He Jiang Wei Ding
Moonis Ali Xindong Wu (Eds.)

Advanced Research in Applied Artificial Intelligence

LNAI 7345

25th International Conference
on Industrial Engineering and Other Applications
of Applied Intelligent Systems, IEA/AIE 2012
Dalian, China, June 2012, Proceedings

Springer

Lecture Notes in Artificial Intelligence 7345

Subseries of Lecture Notes in Computer Science

He Jiang Wei Ding Moonis Ali
Xindong Wu (Eds.)

Advanced Research in Applied Artificial Intelligence

25th International Conference
on Industrial Engineering and Other Applications
of Applied Intelligent Systems, IEA/AIE 2012
Dalian, China, June 9-12, 2012
Proceedings

Springer

Volume Editors

He Jiang
Dalian University of Technology
Dalian, China
E-mail: jianghe@dlut.edu.cn

Wei Ding
University of Massachusetts Boston
Boston, MA, USA
E-mail: ding@cs.umb.edu

Moonis Ali
Texas State University–San Marcos
San Marcos, TX, USA
E-mail: ma04@txstate.edu

Xindong Wu
University of Vermont
Burlington, VT, USA
E-mail: xwu@cs.uvm.edu

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-31086-7 e-ISBN 978-3-642-31087-4
DOI 10.1007/978-3-642-31087-4
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012939335

CR Subject Classification (1998): I.2, H.3-4, F.1-2, C.2, I.4-5, H.2.8

LNCS Sublibrary: SL 7 – Artificial Intelligence

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The last few years have witnessed great advances in applied artificial intelligence to solve real-life problems. To achieve this accomplishment, researchers have devoted significant efforts in various fields, from theoretical analysis to industrial practice.

The International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE), sponsored by the International Society of Applied Intelligence (ISAI), concentrates on artificial intelligence, as well as its applications to complex problem solving. The conference is held annually, and has become an important international event in the field of applied intelligence, where researchers and industrial communities communicate with each other and promote the development of advanced research in artificial intelligence. The 25th IEA/AIE conference was held in Dalian, China, in June 2012. Following the conference tradition, IEA/AIE-2012 provided an international scientific forum for researchers in various fields of applied intelligence.

We received 198 papers from all over the world and selected 82 papers for publication in this volume of LNCS proceedings. Each paper was reviewed by at least two anonymous referees to assure quality. We would like to take this opportunity to thank the Program Committee members as well as all the reviewers for their efforts, which guarantee the high quality of the papers in the proceedings. The selected papers cover theoretical approaches and applications of intelligent systems from various fields of artificial intelligence.

The papers were presented at the IEA/AIE-2012 conference in several regular as well as 11 special sessions. The topics of the papers in the proceedings include data mining, combinatorial optimization and evolutionary algorithms, AI methods and applications, machine learning, agent-based systems, pattern recognition, cognitive computing, decision-making techniques, mission-critical applications, intelligent systems in healthcare, sentiment analysis for Asian languages, social network and its applications, aspects on cognitive computing and intelligent interaction, knowledge-based systems, data mining and computational intelligence for digital forensics and information assurance, mission-critical applications and case studies of intelligent systems, natural language processing and its applications, spatio-temporal data mining, structured learning and their applications, modeling and support of cognitive and affective human processes, and cyber-physical systems for intelligent transportation applications.

The papers in these proceedings highlight the advanced research of applied artificial intelligence and show the potential directions of novel applications. We hope that you will find the papers interesting and obtain inspiration for your research.

The conference also invited three outstanding scholars to give plenary keynote speeches. They were Fei-Yue Wang, from the Chinese Academy of Sciences,

China, Jong Kim, from Pohang University of Science and Technology, South Korea, and Atul Prakash, from the University of Michigan, USA.

In addition to the ISAI and School of Software, Dalian University of Technology, the main sponsors of the conference, our special thanks are due to the Association for the Advancement of Artificial Intelligence (AAAI), Association for Computing Machinery (ACM/SIGART, SIGKDD), Austrian Association for Artificial Intelligence (OeGAI), Catalan Association for Artificial Intelligence (ACIA), Chinese Association of Artificial Intelligence (CAAI), Dalian University of Technology (DUT), European Neural Network Society (ENNS), International Neural Network Society (INNS), Italian Artificial Intelligence Association (AI*IA), Japanese Society for Artificial Intelligence (JSAI), Lithuanian Computer Society–Artificial Intelligence Section (LIKS-AIS), Slovenian Artificial Intelligence Society (SLAIS), Spanish Society for Artificial Intelligence (AEPIA), Taiwanese Association for Artificial Intelligence (TAAI), Taiwanese Association for Consumer Electronics (TACE), and Texas State University-San Marcos.

We wish to thank the members of the Program and Organizing Committees for their hard work, especially those who played essential roles: Mingchu Li (Organizing Chair), Wei Ding (Special Session Chair) and the organizers of the special sessions: Amitava Das, Catholijn M. Jonker, Chengcui Zhang, Chi-Yo Huang, Dan Simovici, Hamido Fujita, Hisham Al-Mubaid, Hong Liu, Honggang Wang, Hyuk Cho, Jan Treur, Jie Liu, Lei Chen, Liangliang Cao, Ling Feng, Mark Hoogendoorn, Mingjian Tang, Ping Chen, Qingzhong Liu, Tomasz F. Stepinski, Wayne Murray, Wei Ding, Yuanchang Xie, and Zhenggang Wang. We would like to thank the keynote speakers for their inspiring and informative talks. We gratefully thank the authors for their valuable contributions and the other participants of the conference. The conference would not have been possible without their support.

April 2012

He Jiang
Wei Ding
Moonis Ali
Xindong Wu

Organization

Chairs

General Chairs

Moonis Ali	Texas State University, San Marcos, USA
Zhongxuan Luo	Dalian University of Technology, China
Xindong Wu	University of Vermont, USA

Program Chairs

Nick Cercone	York University, Canada
He Jiang	Dalian University of Technology, China
Ramamohanarao Kotagiri	University of Melbourne, Australia

Organizing Chair

Mingchu Li	Dalian University of Technology, China

Special Session Chair

Wei Ding	University of Massachusetts Boston, USA

International Organizing Committee

Floriana Esposito, Italy
Fevzi Belli, Germany
Ah-Hwee Tan, Singapore
Shyi-Ming Chen, Taiwan
Chang-Hwan Lee, South Korea
Lakhmi Jain, Australia
Dian-Hui Wang, Australia
Chunsheng Yang, Canada
Salem Benferhat, France
Paul Chung, UK
Hans Guesgen, The Netherlands
Nicolás García-Pedrajas, Spain
Gitchang, Japan
Richard Dapoigny, France
Anwar Althari, Malaysia
Gustavo Arroyo Figueroa, Mexico
Masoud Mohammadian, Australia

Local Organizing Committee

Wenjie Liu, China
Yan Hu, China
Tie Qiu, China
Hong Yu, China
Zhihui Wang, China
Jifeng Xuan, China
Zhilei Ren, China

Special Sessions

Aspects on Cognitive Computing and Intelligent Interaction
Hamido Fujita

Data Mining and Computational Intelligence for Digital Forensics and Information Assurance (DMCIDFIA)
Lei Chen, Hyuk Cho, Qingzhong Liu, and Chengcui Zhang

Data Mining and Intelligent Systems in Healthcare
Mingjian Tang and Wayne Murray

Decision-Making Techniques and Innovative Knowledge Management
Chi-Yo Huang and Hamido Fujita

Mission-Critical Applications and Case Studies of Intelligent Systems
Ling Feng

Natural Language Processing and Its Applications
Hisham Al-Mubaid and Ping Chen

Sentiment Analysis for Asian Languages
Amitava Das

Social Network and Its Applications
Zhenggang Wang and Liangliang Cao

Modeling and Support of Cognitive and Affective Human Processes
Mark Hoogendoorn, Catholijn M. Jonker, and Jan Treur

Spatio-Temporal Data Mining, Structured Learning and Their Applications
Tomasz F. Stepinski, Dan Simovici, Wei Ding, and Jie Liu

Cyber-Physical System for Intelligent Transportation Applications
Honggang Wang, Hong Liu, and Yuanchang Xie

Invited Speakers

Fei-Yue Wang	Institute of Automation, Chinese Academy of Sciences, China
Jong Kim	Pohang University of Science and Technology, South Korea
Atul Prakash	University of Michigan, USA

Program Committee

Ah-Hwee Tan, Singapore
Amruth Kumar, USA
Antonio Bahamonde, Spain
Anwar Althari, Malaysia
Aoying Zhou, China
Bora Kumova, Turkey
C.W. Chan, Hong Kong
Chang-Hwan Lee, South Korea
Chien-Chung Chan, USA
Chih-Cheng Hung, USA
Chunsheng Yang, Canada
Dajun Zeng, China

Dan Tamir, USA
Dianhui Wang, Australia
Djamel F.H. Sadok, Brazil
Don-Lin Yang, Taiwan
Dunwei Gong, China
Enhong Chen, China
Enrique Herrera-Viedma, Spain
Fevzi Belli, Germany
Floriana Esposito, Italy
Gerard Dreyfus, France
Gitchang, Japan
Gregorio Sainz-Palmero, Spain
Gustavo Arroyo, Mexico
Hai Jin, China
Haibin Duan, China
Haibin Yu, China
Hans W. Guesgen, New Zealand
Hasan Selim, Turkey
Hiroshi G. Okuno, Japan
Hongwei Mo, China
Humberto Bustince, Spain
Jyh-Horng Chou, Taiwan
Kaoru Hirota, Japan
Kazuhiko Suzuki, Japan
Ke Tang, China
Kishan Mehrotra, USA
Leszek Borzemski, Poland
Licheng Jiao, China
Lin-Yu Tseng, Taiwan
Maciej Grzenda, Poland
Manton Matthews, USA
Maoguo Gong, China
Marco Valtorta, USA
Mark Hoogendoorn, The Netherlands
Mingchu Li, China
Miquel Sànchez-Marrè, USA
Nicolás García-Pedrajas, Spain
Patrick Brezillon, France
Paul Chung, UK
Prabhat Mahanti, Canada
Richard Dapoigny, France
Ruppa Thulasiram, Canada
Salem Benferhat, France
Shaheen Fatima, UK
Shie-Jue Lee, Taiwan
Shyi-Ming Chen, Taiwan
Srini Ramaswamy, USA
Takayuki Ito, Japan
Tetsuo Kinoshita, Japan
Tibor Bosse, The Netherlands
Tim Hendtlass, Australia
Valery Tereshko, UK
Vic J. Rayward-Smith, UK
Vincenzo Loia, Italy
Walter Potter, USA
Wei Ding, USA
Xingquan Zhu, Australia
Yo-Ping Huang, Taiwan
Youngchul Bae, South Korea
Yu-Bin Yang, China
Zhi-Hua Zhou, China
Zsolt Janos Viharos, Hungary

Special Session Reviewers

Alexandra Balahur, Spain
Alison Sutinen, Australia
Allan Wong, Hong Kong
Bernardete Ribeiro, Portugal
Bipin Indurkhya, India
Björn Gambäck, Norway
Catherine Havasi, USA
Choochart Haruechaiyasak, Thailand
Dawei Wang, USA
Dipankar Das, India
Eduard Hovy, USA
Erik Cambria, Singapore
Eugene Ko, Taiwan
Feng Hong, China
Fiemke Both, The Netherlands
Fuji Ren, Japan
Haiquan Chen, USA
Henry Lo, USA
Hiroya Takamura, Japan
Hung-Yu Kao, Taiwan
Jane You, Hong Kong
Jifeng Xuan, China

Juan Ortega, Spain
Jun Hakura, Japan
Jun Sasaki, Japan
Kun Hua, USA
Lee Nung Kion, Malaysia
Lei Zhang, USA
Luis Soriamorillo, Spain
Manabu Okumura, Japan
Masaki Kurematsu, Japan
Matthijs Pontier, The Netherlands
Mengyu Qiao, USA
Michael Tu, USA
Ming Yang, USA
Minqing Hu, USA
Muaz Niazi, UK
Narasimha Shashidhar, USA
Natalie van der Wal, The Netherlands
Niek Wijngaards, The Netherlands
Noriko Kando, Japan
Paolo Rosso, Spain
Peter-Paul van Maanen, The Netherlands
Ram Basnet, USA
Roberto Revetria, Italy
Rosanne Vetro, USA
Shengli Fu, USA
Sivaji Bandyopadhyay, India
Sumudu Mendis, Australia
Swapna Somasundaran, USA
Tangjian Deng, China
Virginia Dignum, The Netherlands
Xi Li, USA
Xiaodan Zhuang, USA
Xiaojun Ruan, USA
Xin Jin, USA
Xinfa Hu, USA
Yan Chen, China
Yiming Ji, USA
Yin Shan, Australia
Yingsong Hu, Australia
Yintao Yu, USA
Yutaka Watanobe, Japan
Zheng Zeng, USA
Zhijun Yin, USA
Zhilei Ren, China

Additional Reviewers

Arlette van Wissen, The Netherlands
Caihua Liu, China
Charlotte Gerritsen, The Netherlands
Chia-Han Yang, Taiwan
Chia-Li Lin, Taiwan
Claudia d'Amato, Italy
Francisco Rente, Portugal
Gabriele Modena, The Netherlands
Guillermo Alonso, USA
Hueilin Chen, Taiwan
Ike Azogu, USA
Jeroen de Man, The Netherlands
Jing Sun, China
Kui Yu, USA
Miao Xu, China
Michele Nappi, Italy
Nataliya Mogles, The Netherlands
Nicola Di Mauro, Italy
Pascal Cuxac, France
Pingpeng Yuan, China
Ran Zheng, China
Shu-Kung Hu, Taiwan
Song Gao, USA
Wenbin Jiang, China
Xiangjuan Yao, China
Yang Mu, USA
Yao-Huei Huang, Taiwan
Yenchu Chen, Taiwan
Yilin Kang, Singapore
Ying-Hsun Hung, Taiwan
Yixin Lu, The Netherlands
Yong Zhang, China
Zhibin Yu, China
Zhonghang Xia, USA

Table of Contents

Section 1: Machine Learning Methods

Section 2: Cyber-Physical System for Intelligent Transportation Applications

Section 3: AI Applications

Section 4: Evolutionary Algorithms, Combinatorial Optimization

Section 5: Modeling and Support of Cognitive and Affective Human Processes

Section 6: Data Mining and Computational Intelligence for Digital Forensics and Information Assurance

Section 7: Natural Language Processing and Its Applications

Section 8: Social Network and Its Applications

Section 9: Mission-Critical Applications and Case Studies of Intelligent Systems

Section 10: AI Methods

Section 11: Sentiment Analysis for Asian Languages

Section 12: Aspects on Cognitive Computing and Intelligent Interaction

Section 13: Spatio-temporal Data Mining, Structured Learning and Their Applications

Section 14: Decision Making and Knowledge Based Systems

Section 15: Pattern Recognition

Section 16: Agent Based Systems

Section 17: Decision Making Techniques and Innovative Knowledge Management

Section 18: Machine Learning Applications

Automatic Detailed Localization of Facial Features

Qing He[1,*], Ye Duan[2], and Danyang Zhang[3]

[1] Department of Computer Science, University of Missouri, Columbia, MO, USA
qhgb253@gmail.com
[2] Department of Computer Science, University of Missouri, Columbia, MO, USA
duanye@missouri.edu
[3] Department of Mathematics and Computer Science, York College,
The City University of New York, Jamaica, NY, USA
dzhang@york.cuny.edu

Abstract. We propose a complete framework for automatic detailed facial feature localization. Feature points and contours of the eyes, the nose, the mouth and the chin are of interest. Face detection is performed followed by the region detection that locates a rough bounding box of each facial component, and detailed features are then extracted within each bounding box. Since the feature points lie on the shape contours, we start from shape contour extraction, and then detect the feature points from the extracted contours. Experimental results show the robustness and accuracy of our methods. The main application of our work is automatic diagnosis based on facial features.

Keywords: facial feature localization, eyelid, nose boundary, lip contour, generalized Hough transform.

1 Introduction

Automatic localization of facial features is important technique in many applications. Particularly, computer aided diagnosis is becoming an attracting application in recent years. Dalal et al. [1], Loos et al. [2], and Boehringer et al. [3] performed syndrome classification based on the feature points extracted from facial images, but the feature points were manually identified in their works. We aim at developing an automatic method to locate facial features for the purpose of computer based diagnosis, but the approach is applicable to other fields with similar requirement.

Various methods for facial feature detection/localization have been proposed in literatures. Some only focus on feature points localization [4-8]. However, more details of facial features lie in their shape information. While it may not be too tedious to label a few feature points manually, it is almost impractical to delineate shapes of facial features by hand. Therefore, an automatic procedure for facial feature contour extraction is in demand.

Lip contour extraction has been a branch with extensive studies in the past decade. Since lip is rich in color, most previous works utilized the color information to

* This author has joined Center for Neuroscience and Regenerative Medicine, Henry M. Jackson Foundation For the Advancement of Military Medicine since February 2012.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 1–9, 2012.

separate the lip from the background. Wang et al. [9] used three quadratic curves to characterize the lip contour. The lip region was first separated from the background in [10]. Eveno et al. [11] used several cubic curves to model the lip contour. Yokogawa et al. [12] matched a template of four quadratic curves to the lip contour, and the lip was first separated from the background by thresholding the hue component.

Fewer studies exist for the shape extraction of other facial components. Vezhnevets et al. [13] proposed an eye contour extraction method. The points on the upper eyelid were detected by Hough transform. A cubic curve was used to fit the upper eyelid points and a quadratic curve was used to fit the lower eyelid. Zheng et al. [14] used Gabor filter to detect the eye corners and the top and bottom points of the iris, which were then used as control points for a spline curve to fit each eyelid. Previous work on chin contour localization can be grouped into parametric model based methods [15-18] and active contour based methods [19-21]. Ding et al. [15] performed detailed facial feature localization. This is the most complete framework in that it locates the shape of the eyes, nose, mouth, eyebrows and the lower chin. However, the main contribution of [15] is the facial component detection, thus the boundary localization of each facial component is not the focus of their work. Most of their localization methods are oversimplified and are not robust enough for varying subjects and image conditions.

We propose a complete framework for automatic detailed facial feature localization. Features of the eyes, the nose, the mouth, and the chin are of interest. Both feature contours and feature points are located. Feature contours are located first, and most feature points can be found as distinct points on the contours. We believe that locating a feature point in an image is a harder task than locating a set of points on a feature contour, because a group of points on a contour contain much richer information than a single point to distinguish themselves from the rest of the image. That is why feature point detection often involves extensive training on a large sample. Our methods do not require any training, and the feature point localization is reduced to finding extreme-value points on a contour.

The primary application of our work is computer based diagnosis, so we assume the input is a frontal color face image with neutral expression, no occlusion or rotation. Usually these assumptions are easy to satisfy in clinical applications.

This paper is organized as follows; section 2 describes face detection and region detection. Section 3 explains eye feature extraction in detail. Lip contour extraction is elaborated in section 4. Section 5 detects nose and chin boundary. Section 6 shows our experimental results. And the whole paper is concluded in section 7.

2 Face Detection and Region Detection

The details of face detection algorithms are beyond the scope of this research. Here we apply the widely used Viola-Jones face detector [22]. The face image is rescaled to a standard size, e.g., the size of the template face image. The purpose of region detection is to locate a properly sized bounding box of each facial component. The parametric template method proposed by Tanaka et al. [22] is applied here. The detected face and regions are shown in Fig. 1 (b). Note that the chin bounding box is not detected in this step. It will be determined based on the mouth bounding box.

(a) (b)

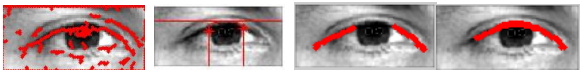
Fig. 1. (a) the input color image; (b) detected face and facial components

3 Eye Feature Extraction

The eye image is first smoothed by a Gaussian filter and normalized in intensity. The reflection in iris is then removed using the method in [7]. Template matching method is then used to locate the iris in the binary image. The top and bottom boundaries are extracted from the detected iris binary image. The top boundary will serve as part of the upper eyelid contour. The entire procedure is shown in Fig. 2.

(a) (b) (c) (d)

Fig. 2. (a) the original eye image; (b) the binary image; (c) the binary image after hole filling; (d) boundaries and key points extracted from the iris region, shown on the normalized eye image after reflection removal



(a) (b) (c) (d)

Fig. 3. (a) local minimum points overlaid on the eye image; (b) search range of the upper eyelid points (the lower left and lower right rectangle regions); (c) detected upper eyelid points by Hough transform; (d) the entire set of upper eyelid points after eye corner detection

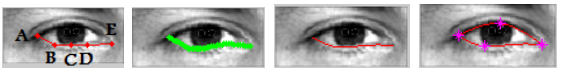
As stated in [13], edge map does not give a robust estimation of the eyelids, and luminance valley points along the horizontal lines were detected as the candidate upper eyelid points. Our method is mostly inspired by [13]. To detect the upper eyelid, a local minimum map is constructed where each local minimum point is the luminance valley point in either horizontal or vertical direction (Fig. 3 (a)). Since the middle part of the eyelid contour is already detected, we only look for the upper eyelid points to the two sides of the iris. We apply Hough transform [24] to detect the two straight lines from the local minimum map. The search space can be restricted to the two sides of the iris (Fig. 3 (b)). The points on the detected lines are the candidate eyelid points (Fig. 3 (c)).

Some false positive points that exceed the bound of the eye may be included in the detected lines. Template matching is applied to locate the eye corners from the candidate eyelid points. Only the eyelid points that are in between the two eye corners are retained. The retained points on the two sides of the upper eyelid together with the

points on the middle eyelid curve constitute the upper eyelid points (Fig. 3 (d)). A Bezier curve is fitted to all these points to obtain the upper eyelid contour (Fig. 3 (d)).

The local minimum map does not capture the lower eyelid very well (Fig. 3 (a)). We approximate the lower eyelid by a piecewise linear curve which consists of four line segments divided by the following five points: two eye corners (A, E), lower left bound of the iris (B), lower right bound of the iris (D), the lowest point in the middle of the iris (C) (Fig. 4 (a)). We start from the line segment AB. By moving B along the vertical direction within a small range we can form several candidate lines. On each line the average gradient magnitude is calculated and the line with the maximum gradient magnitude is selected. The same method is used for the other line segments (Fig. 4 (b)).

A Bezier curve is fitted to all the points on the above line segments to obtain the lower eyelid contour (Fig. 4 (c)). Fig. 4 (d) shows the complete eyelid contour, eye corners and the highest/lowest points on the eyelids.



(a) (b) (c) (d)

Fig. 4. (a) piecewise linear curve of the lower eyelid; (b) detected lower eyelid points; (c) final lower eyelid contour; (d) the entire eyelid boundary and the feature points

4 Lip Contour Extraction

Various color spaces (HSV, Lab, Luv, RGB) have been explored like in [9,11,12]. By examining many color based methods for mouth segmentation, we find that the color transform proposed by [25] is most robust to low mouth/ skin contrast. It transforms an RGB color image (Fig. 5 (a)) to a contrast enhanced grayscale image (Fig. 5 (b)). A binary image can then be easily generated using Otsu's histogram (Fig. 5 (c)). Connected components algorithm is performed to remove small background noise and holes in the mouth region. Then the boundary of the mouth is extracted to get an initial contour (Fig. 5 (d)).

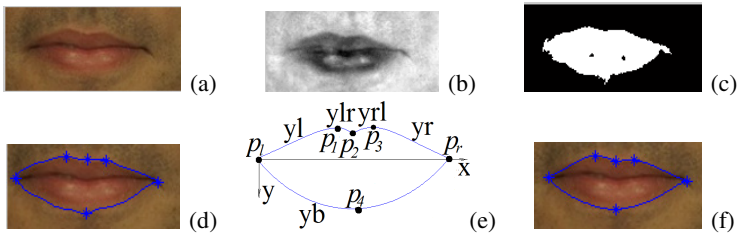


Fig. 5. (a) the original mouth image; (b) contrast enhanced grayscale image; (c) the binary image; (d) the mouth contour extracted from; (e) pl, pr: mouth corner points; p1~p4: four points used to determine the parameters of the parabolas; y*: parabolas for different segments of the lip; (f) the final mouth contour and the feature points

Four parabolic curves (*yl, ylr, yrl, yr*) are used to approximate the upper lip and one (*yb*) to approximate the lower lip (Fig. 5 (e)). Moreover, bending and expansion are applied to the point (x, y) on the CURVE as follows.

$$x' = T(B(x, y), y)$$
$$B(x, y) = x - y^2\delta \tag{1}$$
$$T(x, y) = (x - x_0) / \exp(y \cdot s) + x_0$$

where x' is the new x coordinate of point (x, y). $B(x, y)$ is the bending function proposed in [26] and $T(x, y)$ is the expansion function, x_0 is the x coordinate of the zero-derivative point of the parabola and δ, s are coefficients.

The parameters to be optimized are the coordinates of these six control points and the coefficients for expansion and bending of each curve. The optimization is performed by maximizing an energy function which combines region and gradient information.

$$E = \int_C grad(\vec{p})d\vec{p} + \alpha\int_C region(\vec{p})d\vec{p}$$
$$region(\vec{p}) = \frac{1}{2}\sum_k \frac{(h(k,\Omega_{ip}) - h(k,\Omega_{op}))^2}{(h(k,\Omega_{ip}) + h(k,\Omega_{op}))^2} \tag{2}$$

where $grad(\vec{p})$ is the normalized gradient magnitude at point $\vec{p}$, *h(k,I)* is the kth histogram bin value of image I, Ω_{ip} and Ω_{op} are the regions inside and outside the mouth in a small neighborhood of point $\vec{p}$, and *C* includes the entire lip contour. α is the coefficient to balance the two terms.

The rough positions of the control points can be estimated from the initial mouth contour. The leftmost and rightmost points are identified as mouth corners. The coordinates of $\vec{p}_1 \sim \vec{p}_4$ can be found according to the extrema of y coordinate (Fig. 5 (d)). The optimized parameters of the parabola curves give the coordinates of the feature points as well (Fig. 5 (f)).

5 Nose and Chin Boundary

This section studies nose and chin boundary detection respectively.

Fig. 6. (a) illustration of the nostril detection; (b) detected nostrils and the search range of the nose boundaries (rectangle regions on the left and right sides)

5.1 Nose Boundary Detection

We first locate the rough positions of the nostrils. The average intensities of the rows and the columns of the image are calculated, generating a horizontal profile and a vertical profile of the intensity (Fig. 6 (a)). The x coordinates of the nostrils can be found at the local minima on the horizontal profile, and the y coordinate can be found at the local minimum on the vertical profile.

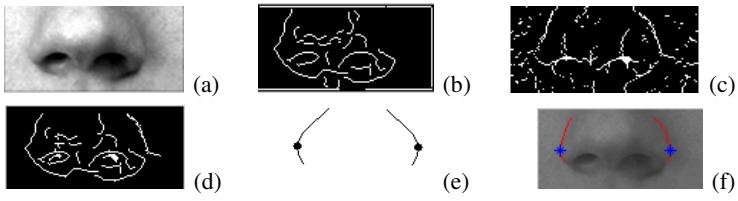


Fig. 7. (a) nose image after histogram equalization; (b) edge map of (a); (c) local minimum map of (a); (d) combined map after small components removal; (e) template nose boundaries and the reference points; (f) detected nose boundaries and feature points shown on the original nose image

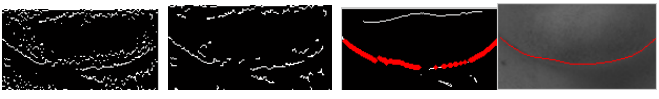
Since the nose boundary is usually weak, histogram equalization is first applied on the grayscale nose image to enhance the contrast (Fig. 7 (a)). An edge map of this enhanced nose image can usually capture most of the nose boundaries. However, it is not always possible to get perfect edges around the nose boundaries (Fig. 7 (b)). Alternatively, local minimum map (Fig. 7 (c)) can be applied here as well which is very close to the edge map around the boundary points, but similar problems may occur as with the edge map. Therefore, we fuse the edge map and local minimum map into a combined map (Fig. 7 (d)).

Generalized Hough transform for arbitrary shapes is then used to detect the boundary contours from the combined map. In our implementation, the template for each nose boundary is a generic shape of the nose boundary, and the reference point is chosen as the leftmost (rightmost) point for the left (right) boundary (Fig. 7 (e)). With the results of the nostril locations, the range of the parameters is restricted to the part of the image shown in Fig. 6 (b). Rotation and scaling parameters can also be included, but the computational cost will be significantly increased. We find that a fixed template shape can serve our purpose well. A Bezier curve is then used to fit the detected local minimum points on each boundary, and the two feature points can then be found on the boundaries (Fig. 7 (f)).

5.2 Chin Boundary Detection

A bounding box immediately below the mouth bounding box is located like in [15]. A combined map is calculated within this bounding box which is the OR of the local minimum map and the local maximum map (Fig. 8 (a)). The reason to use both local minimum and local maximum maps is that the chin points can be intensity valley points or peak points. Small noise regions are removed from the combined map

(Fig. 8 (b)), but there are still some discontinued short line segments in the combined map which are part of the chin (Fig. 8 (b)). Thus, a tensor voting [27] step is added which is a powerful method to identify salient structures in a noisy dataset. The result of tensor voting is shown in Fig. 8 (c), where salient curves are enhanced and the noise is greatly reduced. The chin curve is then detected using generalized Hough transform as described in subsection 5.1.



(a) (b) (c) (d)

Fig. 8. (a) combined local minimum map and local maximum map; (b) combined map after small noise removal; (c) enhanced binary image after tensor voting with the detected chin points overlaid; (d) final chin curve overlaid on the original chin image

The chin curve can be extended to cover a larger portion of the face, and we set the top of the mouth as the upper bound of the chin curve. A larger bounding box containing the face image from the top of the mouth and below is constructed, and generalized Hough transform is performed in a similar way to detect the points on the extended chin. The results are shown in Fig. 9.

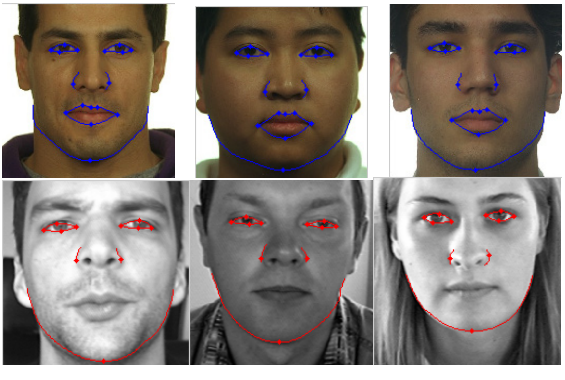


Fig. 9. Results on FERET (top) and BioID (bottom)

6 Results and Validation

The proposed framework is tested on FERET database [28,29] and BioID database [30]. From each database, the images that obviously violate our assumptions are removed. Then 200 images are randomly selected from FERET database and 100 images from BioID database. Mouth features are not applicable to BioID database since our mouth feature localization technique requires color images.

Fig. 9 shows several results on the detected face images, from which we can see the extracted feature points and boundary contours are faithful to the image data. In

order to quantitatively evaluate the performance of our method, the feature points are manually labeled on each image by three different operators. The ground truth position is taken as the average of the manually labeled positions from the three operators. For each feature point, the accuracy of our detection result is calculated as

$$a = 1 - \| \vec{p}_g - \vec{p}_t \| / d_w \tag{3}$$

where $\vec{p}_g$ is the ground truth position of this feature point, $\vec{p}_t$ is the position obtained by our method, $\|\cdot\|$ is the Euclidean distance, and d_w is the width of the corresponding facial component. The value of a is truncated to [0,1]. The average accuracy of all feature points is 0.942 for FERET database, and 0.931 for BioID database.

7 Conclusion

This paper proposes a framework for automatic facial feature localization. Feature points and contours of the eyes, the nose, the mouth and the chin are accurately located. The primary application of this work is the computer aided diagnosis, so only frontal images are considered with some other assumptions. Visual inspection and the quantitative validation of the experimental results demonstrate the accuracy of the feature point localization.

References

1. Dalal, A.B., Phadk, S.R.: Morphometric analysis of face in dysmorphology. Computer Methods and Programs in Biomedicine 85(2), 165–172 (2007)
2. Loos, H.S., Wieczorek, D., Würtz, R.P., Malsburg, C., Horsthemke, B.: Computer-based recognition of dysmorphic faces. Eur. J. Hum. Genet. 11(8), 555–560 (2003)
3. Boehringer, S., Vollmar, T., Tasse, C., Wurtz, R.P., Gillessen-Kaesbach, G., Horsthemke, B., Wieczorek, D.: Syndrome identification based on 2D analysis software. Eur. J. Hum. Genet. 14(10), 1082–1089 (2006)
4. Feris, R.S., Gemmell, J., Toyama, K., Krüger, V.: Hierarchical Wavelet Networks for Facial Feature Localization. In: ICCV 2001 Workshop (2001)
5. Gourier, N., Hall, D., Crowley, J.L.: Facial features detection robust to pose, illumination and identity. In: International Conference on Systems Man and Cybernetics, pp. 617–622 (2004)
6. Cristinacce, D., Cootes, T., Scott, I.: A Multi-Stage Approach to Facial Feature Detection. In: BMVC 2004, pp. 231–240 (2004)
7. Asteriadis, S., Nikolaidis, N., Pitas, I.: Facial feature detection using distance vector fields. Pattern Recognition 42, 1388–1398 (2009)
8. Kozakaya, T., Shibata, T., Yuasa, M., Yamaguchi, O.: Facial feature localization using weighted vector concentration approach. Image and Vision Computing 28, 772–780 (2010)
9. Wang, S., Laua, W.H., Leung, S.H.: Automatic lip contour extraction from color images. Pattern Recognition 37, 2375–2387 (2004)

10. Wang, S.L., Leung, S.H., Lau, W.H.: Lip segmentation by fuzzy clustering incorporating with shape function. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp. 1077–1080 (2002)
11. Eveno, N., Caplier, A., Coulon, P.Y.: Accurate and quasi-automatic lip tracking. IEEE Transactions on Circuits and Systems for Video Technology 14(5), 706–715 (2004)
12. Yokogawa, Y., Funabiki, N., Higashino, T., Oda, M., Mori, Y.: A Proposal of Improved Lip Contour Extraction Method Using Deformable Template Matching and Its Application to Dental Treatment. Systems and Computers in Japan 38(5) (2007)
13. Vezhnevets, V., Degtiareva, A.: Robust and Accurate Eye Contour Extraction. In: Proc. Graphicon 2003, pp. 81–84 (2003)
14. Zheng, Z., Yang, J., Yang, L.: A robust method for eye features extraction on color image. Pattern Recognition Letters 26, 2252–2261 (2005)
15. Ding, L., Martinez, A.: Precise detailed detection of faces and facial features. In: CVPR (2008)
16. Kampmann, M.: MAP estimation of chin and cheek contours in video sequences. EURASIP J. Appl. Signal Process. 2004(6), 913–922 (2004)
17. Wang, J., Su, G.: The research of chin contour in fronto-parallel images. In: Proceedings of the International Conference on Machine Learning and Cybernetics, pp. 2814–2819 (2003)
18. Chen, Q., Cham, W., Lee, K.: Extracting eyebrow contour and chin contour for face recognition. Pattern Recognition 40(8), 2292–2300 (2007)
19. Lam, K.M., Yan, H.: An Analytic-to-Holistic Approach for Face Recognition Based on a Single Frontal View. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(7), 673–686 (1998)
20. Huang, F.Z., Su, J.: Face contour detection using geometric active contours. In: Proceedings of the Fourth World Congress on Intelligent Control and Automation, pp. 2090–2093 (2002)
21. Sun, D., Wu, L.: Face boundary extraction by statistical constraint active contour model. In: Proceedings of the International Conference on Systems, Man and Cybernetics, vol. 6, pp. 14–17 (2002)
22. Viola, P., Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: CVPR, pp. I. 511– I. 518 (2001)
23. Tanaka, K., Sano, M., Ohara, S., Okudaira, M.: A parametric template method and its application to robust matching. In: CVPR, vol. 1, pp. 620–627 (2000)
24. Duda, R., Hart, P.: Use of the hough transform to detect lines and curves in pictures. Communication of the Association of Computer Machinery 15(1), 11–15 (1972)
25. Canzlerm, U., Dziurzyk, T.: Extraction of Non Manual Features for Video based Sign Language Recognition. In: Proceedings of IAPR Workshop, pp. 318–321 (2002)
26. Wörz, S., Rohr, K.: Localization of anatomical point landmarks in 3D medical images by fitting 3D parametric intensity models. Medical Image Analysis 10, 41–58 (2006)
27. Tang, C.K., Medioni, G., Lee, M.S.: Tensor Voting. In: Boyer, K., Sarkar, S. (eds.) Perceptual Organization for Artificial Vision Systems. Kluwer Academic Publishers, Boston (2000)
28. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.: The FERET database and evaluation procedure for face recognition algorithms. Image and Vision Computing J. 16(5), 295–306 (1998)
29. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.: The FERET Evaluation Methodology for Face Recognition Algorithms. IEEE Trans. Pattern Analysis and Machine Intelligence 22, 1090–1104 (2000)
30. http://www.bioid.com/downloads/facedb/facedatabase.html

Confidence-Based Incremental Classification for Objects with Limited Attributes in Vertical Search

Ozer Ozdikis[1], Pinar Senkul[2], and Siyamed Sinir[3]

[1,2] Middle East Technical University, Computer Engineering Department, 06531, ODTU, Ankara/Turkey
[3] Karniyarik Ltd. Kosgeb Teknoloji Gelistirme Merkezi, No:302, 06531, ODTU, Ankara/Turkey
{ozer.ozdikis,senkul}@ceng.metu.edu.tr, siyamed@karniyarik.com

Abstract. With vertical search engines, it is possible to search the web pages on a specific domain such as products, restaurants or academic papers and present the users only the interested information. Gathering and integrating such objects from multiple web pages into a single system provides a useful facility for users. Placing the extracted objects from multiple data sources into a single hierarchical structure is a challenging classification problem, especially if there are limited object attributes. In this work, we propose a confidence-based incremental Naïve Bayesian approach for categorization, focusing on the product domain. Incremental approach is based on extending the training set and retraining the classifier as new objects are assigned to a category with high confidence. The ordering of product data is taken into account as well. The proposed approach is applied on a vertical search engine that collects product data from several online stores.

Keywords: Product Categorization, Information Integration, Vertical Search, Web Crawling, Naïve Bayes Classification.

1 Introduction

Vertical search engines are web search engines that gather information on a specific domain of interest from several web pages. With such a focused search on a specific domain, it is possible to tune the processes of crawling, indexing and searching in order to collect more structured information. Extraction of structured data using vertical search engines is also called object-level vertical search [1][2]. Products from online stores, restaurants from yellow pages, or publications with their title and authors are well-known examples for such objects.

Such a searching facility helps users to find what they are looking for without browsing several web sites. However, while providing this utility to the users, classification of objects into a single hierarchical structure becomes a major problem. Keeping the objects in a hierarchy is necessary for organization and maintenance purposes. Conforming to a category structure facilitates the search and insertion operations. Moreover, people usually expect a tree-like categorization while looking for

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 10–19, 2012.

a product in an online store or while looking for a restaurant in yellow pages. For the sake of usability and user satisfaction, the results of the search engine should similarly be presented in a hierarchy. The classification problem becomes more challenging when the object data extracted from web pages are very limited. Most importantly, it is not always possible to have the category information from all sites [3].

In this paper we present an approach where product data collected by a vertical search engine from multiple online stores are categorized into a single product catalog. For this problem, we developed three methods based on Naïve Bayes classifiers and compared the results on a large dataset. We tested these methods on a vertical search engine named Karniyarik[1] where more than 380 online stores are used for data extraction, and around 3 million products are acquired of more than 7500 brands.

We aimed to implement a classifier that does not require any human intervention. In our approach, the category structure of a comprehensive online store, called *master store*, is used as the target product catalog. Products from other stores, called *source stores*, are aimed to be categorized with respect to this catalog. The first method that we developed is a simple Naïve Bayes classifier that uses the product name and price as classification features. The second method is an improvement over the first one by incrementally expanding the training set using the classified products with high confidence. In this method, we use "confidence" as a measure of the reliability of the category selected by the Naïve Bayes. The third method aims to exploit the product ordering. Considering that the products to be categorized are given in a list, it is more likely to observe the same category for the consecutive products in the list. This *neighborhood* concept is explained later in detail. Our contribution is mainly in the second and third methods, where Naïve Bayes is applied in multiple passes on the product data to be classified.

We would like to note that if the categories of objects were always available, the problem could be considered as a schema/ontology integration problem with different approaches for solution [4][5]. However, being limited to only the product name and price, we implemented a classifier to determine the categories of products based on these attributes.

The rest of the paper is organized as follows: We first present the related work in Section 2. In Sections 3 and 4, we present the crawling technique and proposed classification algorithms respectively. Section 5 gives the results of our experiments. Finally the concluding remarks and future work are presented in Section 6.

2 Related Work

A similar work for crawling and extracting product data from online stores is presented in [1]. Although the classification is considered at the page level, i.e. classifying a page as product or not-product, the idea of aggregating product data into

[1] `http://www.karniyarik.com` (last accessed, 17/08/2011).

a single system is similar. A vertical search engine developed for searching online course materials is introduced in [6]. In this work, homepages of universities are crawled using focused crawlers. Focused crawlers discover the interested pages and filter out the irrelevant ones.

The concept of Named Object (NO) is introduced in [7] as part of a web page that can be named in a user-understandable way. Product data with a name, price and image is given as an example domain dependent NO. The paper discusses the difficulties of NO extraction and proposes a method based on HTML processing (e.g. table, div, etc.) and using web design patterns.

Naïve Bayes is a widely used classification method for text categorization based on word occurrences in documents [8][9]. In this method, Bayes' rule is used for estimating the probability of a document d to be classified in category c, where d is composed of k words, w_1, w_2, …w_k. The probability values are calculated for all categories and the one with the highest probability is assigned as the category of the document. There are several attempts to improve the accuracy of the Bayesian Classifiers. The work in [10] is based on normalizing the document length and giving weights to the terms in the documents. The confidence issue of the Naïve Bayes classifier is addressed in [11]. In order to increase the confidence, an iterative Naïve Bayes is applied, which updates the contingency tables by iteratively cycling through all the training examples.

The idea of combining classified and unclassified documents has been studied in [12]. The method is based on a co-training algorithm, where the documents are partitioned in two views (based on different features) and separate classifiers are trained for these views. A subset of unlabeled documents is classified iteratively using these classifiers, and resulting self-labeled examples are added to training set for the next iterations.

[13] and [14] apply incremental learning for the scenarios that include introduction of new training data, for example coming from a data stream. The main focus is on, how these newly observed data instances are integrated with the old learned Bayesian network and how the network structure can be changed.

Another improvement choice is to include human supervision during the training of the classifiers. In the initial phases of the classification, the results can be presented to the user to be able to make corrections, thus correcting the model and improving the results [15].

The specific case of classifying products to a product catalog has been studied in [16]. An improvement on Naïve Bayes for integrating documents from a source catalog to a master catalog is discussed. The idea is that, if two documents belong to the same category in the source then, because they must be similar, they may belong to the same category in the master catalog with a higher probability. The authors design a method to tune a coefficient value that represents the weight of this parameter in the classification calculations. Some assumptions in this approach are the availability of catalog information for all documents, homogeneity and a significant overlap between the source and master catalogs.

3 Crawling the Products

There are several algorithms for crawling, which differ mostly in the visiting strategy of new web pages. Focused crawlers are written for a specific purpose on a specific domain, therefore prioritizing the content to be extracted. In [17], further improvements on how to improve their performance and quality are discussed. In this work, we implemented a simpler and well-known algorithm for crawling, namely the Breadth-First (BF) algorithm [3]. Given the homepage of an online store, links are visited recursively in a breadth-first manner. The product information in a crawled page is identified by running XML queries written in XQuery[2] language on the HTML content. These queries are written specific to each store in a template-dependent manner, which tries to match product descriptions in HTML.

In addition to its simplicity, the BF approach is especially advantageous in our proposed solution. The reason is that we aim to feature the crawling order while making the classification. It is a common pattern for online stores that products of the same type are presented together. Therefore, with BF crawling, it is expected to retrieve objects in the same category in chunks. How this valuable intuition is used is explained in the following section.

4 Product Classification

In this work, there are two key features used for classification, namely product names and their prices. A product name can be considered as a bag (or possibly a set) of terms [8]. Before the classification process, preprocessing is applied on these names: terms in the product names are extracted; word stemming is applied; stop words and punctuations are removed. Therefore, every product is represented with a list of terms extracted from their names. As mentioned earlier, product data of the master store is used as the training set. Thus the categories are populated with the terms and prices of the products in the master store. As a result of this process, each category "*c*" includes a price list of the products belonging to that category, and a list of term-frequency pairs "<*t, f*>" for all distinct terms extracted from the product names in category *c*. In this mapping, *f* represents the number of products in *c* that contain the term *t* in their names. The mean and standard deviation of the prices are found prior to the classification process. These statistics about the terms and prices are used in order to calculate the probability of an unclassified product to belong to a category.

4.1 Method I: Classification with Naïve Bayes

The first method uses the price "p_i" and terms "$terms_i$" extracted from the name of the product "d_i" as features in the Naïve Bayes classification. Therefore, the category of the product d_i will be the category c with highest posterior probability, which is calculated with the well-known formula given in equation (1).

[2] `http://www.w3.org/xml/query`

$$c = \arg\max_{c \in C} P(c) \cdot P(p_i \mid c) \cdot P(terms_i \mid c) \quad (1)$$

The prior probability of a product to be in category c, represented as $P(c)$, is the ratio of the number of preclassified products in c to the total number of preclassified products in the training set. This means, the most crowded category has the highest chance, independent of the product price or name. Price is handled as a continuous feature with normal distribution. While finding the probability of product d_i being in category c, its price p_i is given to a Gaussian equation [18]. While finding the probability using product name feature, each term t in the product name is searched in the term-frequency mappings of c, and corresponding f values are used for calculating the probability $P(terms_i|c)$, as given in equation (2).

$$P(terms_i \mid c) = P(t_1, t_2, ...t_k \mid c) = \prod_{j=1}^{k} \frac{f_j}{n_d \in c} \quad (2)$$

where ($n_d \in c$) represents the number of products in category c. In order to avoid the zero-count problem in the formula, a variation of the no-match approach is used [9].

After normalization, these probability values are finally used to calculate the posterior probabilities of all categories for a product. The category with the highest probability is assigned to that product.

4.2 Method II: Incremental Extension of Training Set

In this method, the probability calculations used in Method I are used again, but this time with the additional criteria for confidence. Traditionally, Naïve Bayes assigns the category with the highest posterior probability to the products in one pass. However the confidence of this decision depends on the posterior probabilities of other categories. For example, if the posterior probability value found by Bayes for the best category c_1 is 51% and the probability for the second category c_2 is 49%, it would be questionable to assign the product to the first category. It would be more confident if $P(c_1 \mid d) \gg P(c_2 \mid d)$ rather than just $P(c_1 \mid d) > P(c_2 \mid d)$. Confidence values are found by taking the ratio of the best category probability to the second best category probability, i.e. $P(c_1 \mid d) / P(c_2 \mid d)$. The intuition behind Method II is to proceed incrementally and accept only a group of most confident classifications at each step. These classifications may provide useful information for the following iterations. Therefore, the products classified at each step are added to the training set. At the end of each step, term frequencies and price statistics are updated for all categories. As a result, unclassified product data from an unknown store is integrated into the training set, gaining more information about new prices and partially learning the vocabulary of the source store. In our experiments, 20% of the most confident categorizations are taken at each step and added into the training set.

4.3 Method III: Including Product Ordering

Method III adds onto the Method II by taking the neighborhood information, i.e. the product order, into account. As a result of the crawling technique we used, similar type of products in a source store are extracted in groups as explained before.

We say "product d_1 is in the *n-neighborhood* of product d_2" if the number of products between d_1 and d_2 is less than n in the ordered list. While deciding the right category for a product, categories of other products in the neighborhood can be utilized in our incremental approach. Table 1 is an example scenario for using the classifications in the neighborhood. In this scenario, assume that we are looking for the category of d_i and assume some of the products in its 3-neighborhood are assigned to a category in the previous iterations. By looking at this table, one can expect d_i to be assigned to the category "Laptop" (with probability 4/6). There is also a very minor chance for the category "LCD TV" (with probability 1/6). This is actually how we use this guidance together with probabilities found with Naïve Bayes. In other words, posterior probabilities produced by Naïve Bayes are compared with the probabilities generated by looking at the neighborhood.

Table 1. Example categories in 3-neighborhood

Product	Assigned Category	Comment
d_{i-3}	Laptop	Products in
d_{i-2}	LCD TV	3-neigborhood
d_{i-1}	Laptop	of di
d_i	?	Product to classify
d_{i+1}	Laptop	Products in
d_{i+2}	?	3-neigborhood
d_{i+3}	Laptop	of di

There are 3 phases in Method III. The first phase is the assignment of a group of categories with the highest confidence. As in Method II, this first group is the top 20% of the products in the source store. The second phase extends Naïve Bayes with the neighborhood information. That means, in the following iterations, in addition to the confidence that is based on posterior probabilities, the agreement of the neighborhood is also required. The best categories proposed by Naïve Bayes and the most probable category found by checking the neighborhood must overlap. For the example case in Table 1, if Naïve Bayes finds "Laptop" as the category with the highest probability, only then d_i will be assigned to a category. As a result, the training set is extended with product classifications with much stronger confidence. The second phase ends when no more products can be assigned to a category, i.e. Naïve Bayes and neighborhood statistics contradict each other. Then the final phase starts. In the final phase, if the category with the highest probability in the neighborhood is among the top-3 categories suggested by Naïve Bayes, it is accepted. Otherwise, the best suggestion of Naïve Bayes is taken. For example, if "Laptop" is among the best three categories found by Naïve Bayes, d_i is assigned to it. Otherwise, whatever is suggested as the best category by Naïve Bayes is taken.

5 Experimental Results

The heterogeneity of catalogs and products introduces some difficulties in analysis. A store may have a single category for all kinds of "Computers", while another one may have deeper levels like "Laptops", "Desktop PCs" and "Netbooks". Similarly several categories may be suitable for a product. For example a "Memory Card" may be classified under "Computer Peripherals" or "Camera Storage". We applied our analysis on 6 source stores selected among the online stores currently in use. Due to the problems addressed above, instead of manually setting the expected categories for each individual product, we selected an appropriate subset of categories in the source catalogs, and identified the expected categories for them in the master catalog. Three methods which were explained in the previous section are applied on these datasets. To measure the accuracy, we use the precision, recall, success rate (ratio of correctly classified products over all products) and F-score. The formula for F-score is given in (3).

$$FScore = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{3}$$

First we present two tables for a detailed analysis on the category level for the first two source stores. For the first source store, we selected 713 products from 24 categories and identified their expected categories in the master catalog. Then we applied our classification methods on these products, and compared the results with our expectations. Table 2 shows the results for the first source store.

Table 2. Results for the first source store

	Method I		Method II		Method III	
	precision	recall	precision	Recall	precision	recall
Kitchen Sets	0.970	0.804	0.953	1.0	0.953	1.0
Ovens	0.940	0.979	1.0	0.979	1.0	0.937
Laptops	0.988	1.0	0.988	1.0	0.988	1.0
Desktop PCs	1.0	0.980	1.0	0.980	1.0	1.0
Netbooks	1.0	0.956	1.0	0.956	1.0	1.0
LCD TVs	0.986	0.935	0.985	0.857	0.987	0.987
Average	0.983	0.977	0.986	0.981	0.991	0.982
F-Score	0.979		0.983		0.986	
Success Rate	0.969		0.973		0.985	

In this table we give the precision/recall values for some of the categories. The row labeled with "Average" shows the average precision/recall values for the 24 categories covering all products selected from this source store. Success rates and F-scores which are calculated by using average precision/recall values are in the last two rows of the table. The increase in the success rate and better precision/recall values indicate a slight improvement from Method I through Method III.

There is a slight deterioration in the coverage of Method II for LCD TVs. When we analyzed the output, we realized that some of the LCD TVs are categorized as LED TVs, which is incorrect. The reason is the similarity of product names in these categories. Using the second method, LED TVs may have higher confidence in the initial iterations, thus may be categorized before the LCD TVs. Considering the common terms of LCD and LED TVs, this may change the balance in favor of the LED TVs in the following iterations. However, this problem is no longer observed using Method III, where the categories of the products in the neighborhood are taken into account.

As the second case study, 225 sample products from 4 categories are selected from the second source store and their expected categories are determined. The results are shown in Table 3. The improvement observed for this store is more obvious.

Table 3. Results for the second source store

	Method I		Method II		Method III	
	precision	recall	precision	recall	precision	recall
Laptops	0.923	0.843	0.924	0.852	0.964	0.939
Desktop PCs	0.821	0.754	0.859	0.901	0.921	0.967
Netbooks	0.758	0.785	0.733	0.785	0.923	0.857
LCD TVs	1.0	0.619	1.0	1.0	1.0	1.0
Average	0.875	0.750	0.879	0.884	0.952	0.940
F-Score	0.807		0.881		0.945	
Success Rate	0.791		0.871		0.942	

In addition to the detailed accuracy analysis for two stores given above, F-scores and success rates for all 6 source stores are presented in Fig. 1 and Fig. 2 respectively. The test data sizes for these 6 source stores range from a few hundred to one thousand.

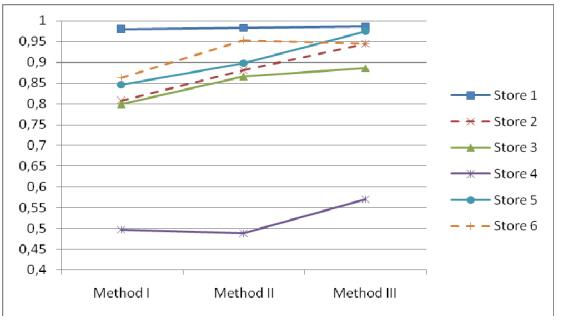


Fig. 1. F-measures of our methods on 6 source stores

It has been observed that the product names in the source stores have a strong effect on the results. Some stores do not give much information about the product in their product names. For example Store-4, the store with the worst accuracy values,

provides just the brand and a code word for the product name. If the master store does not know about this code word, it is almost impossible to find the correct category. Low accuracy values are the result of this problem. Such product descriptions are not much explanatory even for the human users. This can be fixed only through more comprehensible product names.

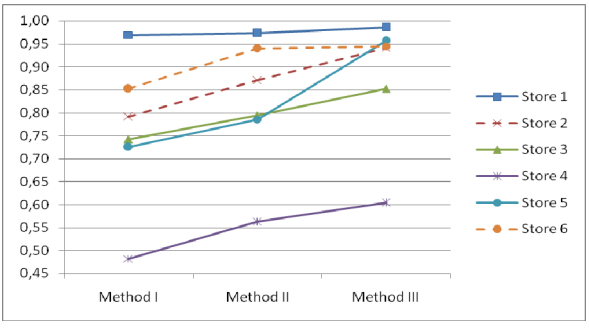


Fig. 2. Success rates of our methods on 6 source stores

6 Conclusion

In this work we introduce a solution to aggregate objects into a single hierarchy extracted from multiple data sources. The solution is based on an improved Naïve Bayes classifier where there are few attributes to decide on the category of the object, and where there is no category information of the objects to apply schema matching techniques. Our implementation is tested on a product search engine currently in the market. It collects the product data extracted from multiple online stores in a master product catalog. In Method I, simple Naïve Bayes classifiers are used in order to select the category of a product. In order to increase the accuracy, in Method II, Naïve Bayes is run in multiple passes, and at each pass, a subset of the products are assigned to the identified categories. This subset is found by using the confidence of the category selection. This method is further extended in Method III by including the order of the products in the product file extracted from the source store. The assumption is that the neighboring products have a higher chance to belong to the same category. We tested out implementation on the products extracted from 6 source stores. Remarkable improvements on F-scores and success rates are observed. This approach can further be improved in several directions. One major problem, as addressed in the previous section, is the differences of the terms used in product names. If the terms in the product name are completely different than the ones in the master store, it is mostly the price which determines the category for that product. A possible solution is to use Web Search Engines in such a way that the unknown product name is given to a Web Search Engine together with the label of the possible categories. The category with the highest hit count could be accepted as the correct category.

References

1. Nie, Z., Wen, J.-R., Ma, W.-Y.: Object-Level Vertical Search. In: 3rd Biennial Conference on Innovative Data Systems Research (CIDR), pp. 235–246 (2007)
2. Olston, C., Najork, M.: Web Crawling. Foundations and Trends in Information Retrieval 4(3), 175–246 (2010)
3. Ding, Y., Korotkiy, M., Omelayenko, B., Kartseva, B., Zykov, V., Klein, M., Schulten, E., Fensel, D.: GoldenBullet: Automated Classification of Product Data in E-commerce. In: Proceedings of Business Information Systems, Poland (2002)
4. Rahm, E.: Towards Large-Scale Schema and Ontology Matching. Schema Matching and Mapping, part:1, 3–27 (2011)
5. Wick, M.L., Rohanimanesh, K., Schultz, K., McCallum, A.: A Unified Approach for Schema Matching, Coreference and Canonicalization. In: 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA (2008)
6. Xiong, Y., Luo, P., Zhao, Y., Lin, F., Feng, S., Zhou, B., Zheng, L.: OfCourse: Web Content Discovery, Classification and Information Extraction for Online Course Materials. In: 18th ACM Conference on Information and Knowledge Management (CIKM), Hong Kong, China (2009)
7. Snasel, V., Kudelka, M.: Web Content Mining Focused on Web Objects. In: 1st International Conference on Intelligent Human Computer Interaction, India, pp. 37–58 (2009)
8. Lewis, D.D.: Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)
9. Kohavi, R., Becker, B., Sommerfield, D.: Improving Simple Bayes. In: van Someren, M., Widmer, G. (eds.) ECML 1997. LNCS, vol. 1224, pp. 78–87. Springer, Heidelberg (1997)
10. Kolcz, A., Yih, W.: Raising the Baseline for High-precision Text Classifiers. In: 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, pp. 400–409 (2007)
11. Gama, J.: Iterative Bayes. Intelligent Data Analysis 4(6), 475–488 (2000)
12. Blum, A., Mitchell, T.: Combining Labeled and Unlabeled Data with Co-training. In: 11th Conference on Computational Learning Theory (COLT), New York, USA, pp. 92–100 (1998)
13. Klawonn, F., Angelov, P.: Evolving Extended Naive Bayes Classifier. In: 6th IEEE International Conference on Data Mining, Los Alamitos, pp. 643–647 (2006)
14. Friedman, N., Goldszmidt, M.: Sequential update of Bayesian network structure. In: 13th Conference on Uncertainty in Artificial Intelligence, Rhode Island, USA, pp. 165–174 (1997)
15. Agrawal, R., Bayardo, R., Srikant, R.: Athena: Mining-Based Interactive Management of Text Databases. In: Zaniolo, C., Grust, T., Scholl, M.H., Lockemann, P.C. (eds.) EDBT 2000. LNCS, vol. 1777, pp. 365–379. Springer, Heidelberg (2000)
16. Agrawal, R., Srikant, R.: On Integrating Catalogs. In: 10th International Conference on World Wide Web, Hong Kong, pp. 603–612 (2001)
17. Batsakis, S., Petrakis, E.G.M., Milios, E.: Improving the Performance of Focused Web Crawlers. Data and Knowledge Engineering 68(10), 1001–1013 (2009)
18. Mitchell, T.: Machine Learning. McGraw Hill (1997)

Algorithms for Detecting Outliers via Clustering and Ranks

Huaming Huang, Kishan Mehrotra, and Chilukuri K. Mohan

Department of EECS, Syracuse University

Abstract. Rank-based algorithms provide a promising approach for outlier detection, but currently used rank-based measures of outlier detection suffer from two deficiencies: first they assign a large value to an object near a cluster whose density is high even through the object may not be an outlier and second the distance between the object and its nearest cluster plays a mild role though its rank with respect to its neighbor. To correct for these deficiencies we introduce the concept of modified-rank and propose new algorithms for outlier detection based on this concept. Our method performs better than several density-based methods, on some synthetic data sets as well as on some real data sets.

Keywords: Outlier detection, ranking, neighborhood sets, clustering.

1 Introduction

Outlier detection is an important task for data mining applications. Several effective algorithms have been successfully applied in many real-world applications. Density-based algorithms such as "local outlier factor" (LOF) and connectivity-based outlier factor (COF) were proposed by [1] and [8] respectively. Jin *et al.* [7] proposed another modification, called INFLO, which is based on a symmetric neighborhood relationship. Outliers detection based on clustering has been proposed in the literature, see Chandola et al. [3], where an object is declared as an outlier if it does not belong to any cluster. This in turn, requires a new clustering philosophy in which all objects of a given data set are not required to be in at least one cluster. Tao and Pi [9] have proposed a density-based clustering and outlier detection (DBCOD) algorithm, which belongs to this category. In this paper we use clustering to eliminate the objects that are not suspected outliers and evaluate outlierness of the remaining objects only.

Another rank based detection algorithm (RBDA) was recently proposed by Huang et al.[5]. It was observed that RBDA demonstrates superior performance than LOF, COF, and INFLO. However, RBDA is found to assign a large outlierness value to an object in the vicinity of a large cluster, although the object may not be an outlier. In this paper we present few approaches to rectify this deficiency of RBDA — first is a simple modification to RBDA whereas in the second and third approaches the size of the cluster is explicitly addressed; in all cases clustering acts as a preprocessing step.

The paper is organized as follows. In Section 2, after introducing key notations and definitions, we briefly describe RBDA and DBCOD. In Section 3, first we

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 20–29, 2012.

illustrate the above described weakness of RBDA followed by suggested measures of outlier detection. These new measures are compared with RBDA and DBCOD using one synthetic and three real data sets. Brief descriptions of data sets and a summary of our findings are presented in Section 4, followed by the conclusions and future work.

2 Notation and Definitions

The following notations and concepts are used throughout the paper.

2.1 Notation

- D denotes the given dataset of all observations.
- $d(p, q)$ denotes the distance between two points $p, q \in D$. This distance measure could be any appropriate distance but for concreteness we use the Euclidean distance.
- $d_k(p)$ = the distance between p and its kth nearest neighbor, where $k > 0$ is a positive integer.
- $\mathcal{N}_k(p) = \{q \in D - \{p\} : d(p, q) \leq d_k(p)\}$ denotes the set of k nearest neighbors of p.
- $r_q(p)$ denotes the rank of p among neighbors of $q \in \mathcal{N}_k(p)$; i.e., $r_q(p)$ is the rank of $d(q, p)$ in $\{d(p, o) : O \in D - \{q\}\}$.
- $\mathcal{RN}_k(p) = \{q : q \in D \text{ and } p \in \mathcal{N}_k(q)\}$ denotes the set of reverse k nearest neighbors of p.

2.2 Definitions

The following definitions are used in the proposed clustering algorithm; all definitions are relative with respect to a positive integer ℓ. In other words, for example, D-reachable defined below should be viewed as D-reachable given ℓ.

- *D-reachable* – An object p is directly reachable (D-reachable) from q, if $p \in \mathcal{N}_\ell(q)$.
- *Reachable* – An object p is reachable from q, if there is a chain of objects $p \equiv p_1, \ldots, p_n \equiv q$, such that p_i is D-reachable from p_{i+1} for all values of i.
- *C*onnected – If p is reachable from q, and q is reachable from p, then p and q are connected.
- *Neighborhood Clustering*(NC-clustering) – We use the breadth-first search on a graph whose node-set is D and where an edge exists between $p, q \in D$ if $p \in \mathcal{N}_k(q)$ and $q \in \mathcal{N}_k(p)$. A connected component C of the graph is a cluster if the following three conditions are satisfied:
  1. For any two objects p and q in C, $p \neq q$, p and q are connected.
  2. For $p \in C$, p is D-reachable from at least two other objects in C.
  3. $|C| \geq m^*$, where m^* is the minimum number of objects in a cluster, it is pre-defined by users (domain experts).

If any connected component C does not satisfy these conditions, it is broken up into isolated points and all such objects are declared potential outliers. Condition 3 above is used to avoid treating a small number of outliers as a cluster. We denote the clustering method as NC-clustering; more formally as NC-clustering($\ell, m*$). For instance, NC-clustering(6,5) means that a cluster contains connected objects for $\ell = 6$ and a cluster must contain at least 5 objects.

The values of ℓ and m^* are mainly decided based on domain knowledge. If ℓ is small NC-clustering method will find small and tightly connected clusters and large value of ℓ will find large and loose clusters. If the clusters are small and tight, we expect to find more objects that don't belong to any cluster whereas in the latter case, only a few objects will be declared as outliers. In real world applications (such as credit card fraud detection) most of the transactions are normal and only 0.01% or less of the transactions are fraudulent. In this case, a small value of ℓ is more suitable than a large ℓ. The value of m^* has a similar effect: if m^* is too small, then the cluster size may also be too small, and a small collection of outliers may be considered as a cluster, which is not what we want. In our experiments, m^* is set to a fixed value of 6.

RBDA is a rank-based outlier detection approach that identifies outliers based on mutual closeness of a data point and its neighbors. For $p, q \in D$, if $q \in \mathcal{N}_k(p)$ and $p \in \mathcal{N}_k(q)$, then p and q are "close" to each other. To capture this concept we define a measure of "outlierness" of p, as follows:

$$O_k(p) = \frac{\sum_{q \in \mathcal{N}_k(p)} r_q(p)}{|\mathcal{N}_k(p)|}.$$

If $O_k(p)$ is 'large' then p is considered an outlier.

Density-Based Clustering and Outlier Detection Algorithm (DBCOD). For $p \in D$, Tao and Pi [9] define the local density, the neighborhood-based density factor, and neighborhood-based local density factor of p, respectively, as:

$$\mathrm{LD}_k(p) = \frac{\sum_{q \in \mathcal{N}_k(p)} \frac{1}{d(p,q)}}{|\mathcal{N}_k(p)|}, \ \mathrm{NDF}_k(p) = \frac{|\mathcal{RN}_k(p)|}{|\mathcal{N}_k(p)|}, \text{ and } \mathrm{NLDF}_k(p) = \mathrm{LD}_k(p) \times \mathrm{NDF}_k(p).$$

The threshold of NLDF, denoted as τ_{NLDF}, is defined as:

$$\tau_{\mathrm{NLDF}} = \begin{cases} \min_k(\mathrm{NLDF}_k(p)) & \text{if for all objects } p \in D, \mathrm{NDF}_k(p) = 1 \\ \max_k(\mathrm{NLDF}_k(p)) & \text{otherwise} \end{cases}$$

Using the above definitions, Tao and Pi's [9] find the clusters based on the definitions in section 2.2, *except their definition of D-reachability is as follows: p and q are in each other's k-neighborhood and* $NLDF_k(q) < \tau_{NLDF}$. Points outside the clusters are declared as outliers.

3 Weighted RBDA and Other Improvements

In general RBDA performs better than density-based algorithms such as LOF, COF and INFLO (see [6]). These density based measures do not assign appropriate

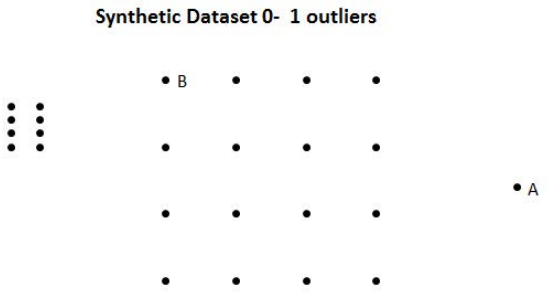


Fig. 1. Synthetic dataset-0 with one outlier, but LOF, COF and INFLO identify B as the most significant outlier

measures of outlierness to one or two objects that are clearly far away from a cluster whereas RBDA is mostly successful. A simple example illustrates this observation. Consider the synthetic dataset in Figure 1. This dataset contains two clusters of different densities and an 'outliers' A. For $k = 5, 6, 7$ or 8, the density-based algorithms such as LOF, COF and INFLO do not identify A as the most significant outlier. Instead, B is their top choice, which is wrong. Why B gets a higher outlier value? The reason is that some of B's k-neighbors are from a high density cluster while the others are from a low density cluster and due to mix density of neighborhoods density-based algorithms fail to identify object A as an outlier. RBDA identifies the object A as the most significant outlier.

However, behavior of RBDA is also inconsistent with expectation when an object is near a dense cluster, which we identify as the 'cluster density effect'. Consider the data in Figure 2 where two points are of special interest; A in the neighborhood of a cluster with low density (25 objects) and B in the neighborhood of a cluster with high density (491 objects).

By visual inspection, it can be argued that the object 'A' is an outlier whereas object 'B' is a possible but not definite outlier. For k=20, $O_{20}(A)$=25 because rank of 'A' is 25 from all of its neighbors. On the other hand, the ranks of ''B' with respect to its neighbors are: 2, 8,. . . , 132, 205, 227; so that $O_{20}(B)$ is 93.1. RBDA concludes that 'B' is more likely outlier than 'A'. It is clearly an artifact due to large and dense cluster in the neighborhood of 'B', i.e., a point closer to a dense cluster is likely to be misidentified as an outlier, even though it may not be. Such behavior of RBDA, due to cluster density, is observed for some values of k.

By visual inspection, we generally conclude that a point is an outlier if it is 'far away' from the cluster. This implies that the distance of the object (from the cluster) plays an important role; but accounted for in RBDA only through ranks. Perhaps this deficiency in RBDA can be fixed by incorporating distance

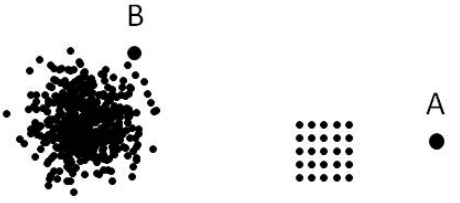


Fig. 2. An example to illustrate 'Cluster Density Effect' on RBDA; RBDA assigns larger outlierness measure to B

in RBDA. The distance can be measured in many ways; either collectively for objects in $\mathcal{N}_k(p)$ or by accounting for the distance of each $q \in \mathcal{N}_k(p)$ separately. These different ways of accounting for distance lead to potentially many possible measures of outlierness. We have explored some of them but in the next subsection we present only one that performed better than others.

3.1 Weighted RBDA (RADA)

Rank-based approach ignores useful information contained in the distance of the object from other neighboring objects. To overcome this weakness of RBDA due to "cluster density effect", we propose to adjust the value of RBDA by the average distance of p from its $k-$neighbors. Step by step description of this rank and distance based detection algorithm is given below:

1. Choose three positive integers k, ℓ, m^*.
2. Find the clusters in D by $\mathcal{NC}(\ell, m^*)$ method.
3. Declare an object o a potential-outlier if it is does not belong to any cluster.
4. Calculate a measure of outlierness: $W_k(p) = O_k(p) \times \frac{\sum_{q \in \mathcal{N}_k(p)} d(q,p)}{|\mathcal{N}_k(p)|}$.
5. If p is a potential-outlier and $W_k(p)$ is large, declare p is an outlier.

For the dataset in Figure 2 we observe that $W_{20}(\mathrm{A}) = 484.82$ and $W_{20}(\mathrm{B}) = 396.19$ implying that A is more likely outlier than B, illustrating that RADA is capable of fixing the discrepancy observed in RBDA.

3.2 Outlier Detection Using Modified-Ranks (ODMR)

In this section we propose an alternative procedure to overcome the cluster density effect. We have observed that the size of neighboring cluster plays an important role when calculating the object's outlierness via RBDA. To modify this effect, all clusters of all sizes are assigned equal weights (including isolated points viewed as a cluster of size 1) and all $|C|$ observations of the cluster are assigned equal weights $= 1/|C|$.[1] The rank $r_q(p)$ of an observation p is equal to the number of points within a circle of radius $d(q,p)$ centered at q. In RBDA

[1] We have experimented with another weight assignment to points within a cluster, equal to $1/\sqrt{|C|}$, but the results are not as good as when weights are $1/|C|$.

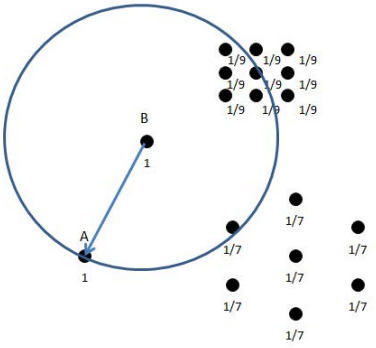


Fig. 3. Assignment of weights in different clusters and modified-rank (modified-rank of A, with respect to B, is $1 + 1 + 5 \times \frac{1}{9} + \frac{1}{7}$.)

we sum $r_q(p)$ for all values of $q \in \mathcal{N}_k(p)$. In the proposed version, we calculate "modified-rank" of p, which is defined as the sum of weights associated with all observations within the circle of radius $d(q,p)$ centered at q; that is

$$\text{modified-rank of } p \text{ from } q = mr_q(p) = \sum_{s \in \{d(q,s) \le d(q,p)\}} \text{weight}(s),$$

and sum the "modified-ranks" in $q \in \mathcal{N}_k(p)$.

Figure 3 illustrates how modified-rank is calculated. Step by step description of the proposed method is as follows:

1. Choose three positive integers k, ℓ, m^*.
2. Find clusters in D by $\mathcal{NC}(\ell, m^*)$. All objects not belonging to any cluster are declared as potential-outliers.
3. If C is a cluster and $p \in C$, then the weight of p is $b(p) = \frac{1}{|C|}$.
4. For $p \in D$ and $q \in N_k(p)$, Q denotes the set of points within a circle of radius $d(q,p)$, i.e., $Q = \{s \in D \mid d(q,s) \le d(q,p)\}$. Then the modified-rank of p with respect to q, denoted as $mr_q(p)$, is computed as $mr_q(p) = \sum_{s \in Q} b(s)$.
5. For a potential outlier p, its ODMR-outlierness, denoted as $\text{ODMR}_k(p)$, is defined as: $\text{ODMR}_k(p) = \sum_{q \in \mathcal{N}_k(p)} mr_q(p)$
6. If p is a potential outlier and $\text{ODMR}_k(p)$ is large, we declare p is an outlier.

3.3 Outlier Detection Using Modified-Ranks with Distance (ODMRD)

Influenced by the distance consideration of section 3.1, in this section we present yet another algorithm that combines ODMA and distance. $\text{ODMRD}_k(p)$ is obtained by implementing all steps as before except Step 5 of the previous algorithm is modified as follows:

(5*) For a potential outlier p, its ODMRD-outlierness, denoted as $\text{ODMRD}_k(p)$, is defined as: $\text{ODMRD}_k(p) = \sum_{q \in \mathcal{N}_k(p)} mr_q(p) \times d(q,p)$

4 Experiments

4.1 Datasets

We use one synthetic and three real datasets to compare the performance of RBDA with RADA, ODMR, ODMRD and Tao and Pi's DBCOD.

Real Datasets. Real datasets consist of iris, ionosphere, and Wisconsin breast cancer datasets obtained from UCI repository. The real datasets were used in two different ways, following the criterion used in [4],[8], and [2]:

1. By making a rare set out of one the class. (1) In the Iris dataset, which is a three-class problem and contains 150 observations equally divided in three classes, 45 observations were removed randomly from the iris-setosa class. (2) In the ionosphere dataset, which is a two-class problem, out of 126 'bad' instances, 116 were randomly removed, leaving 10 'outliers'. (3) Finally, in the Wisconsin dataset, which is also a two-class problem and consists of 236 observations of benign and 236 observations of malignant cancer, after removing duplicates and observations with missing features, 226 malignant observations were removed, leaving 10 'outliers'.
2. By planting new observations in the existing datasets. These planted observations are such that one or more features are assigned the extreme values. (1) In the Iris dataset three observations were planted, (2) in the ionosphere dataset three outliers were planted and (3) in the Wisconsin dataset two outliers were planted.

Synthetic Datasets. The synthetic datasets are two dimensional so that it is easy to see and interpret the results. Synthetic dataset consists of 515 instances including six planted outliers; has one large normally-distributed cluster and two small uniform clusters. This datasets is intended to test the algorithms' ability to overcome the problem of "cluster density effect". This dataset and clusters obtained by an application of $\mathcal{NC}(6, 6)$, are depicted in Figure 5.

4.2 Performance Measures

Three metrics, m_t, recall, and RankPower [5], are selected to measure the performance of proposed algorithms; briefly defined below. We list m most suspicious objects in the dataset D, by a given outlier detection algorithm, which contains exactly d_t true outliers. Let the algorithm produces m_t (true) outliers out of m. Suppose that the algorithm assigns the rank R_i to the ith outlier among m, where $R_i = 1$ represents most suspicious outlier and a larger value of R_i means that the algorithm considers that the ith outlier is less suspicions. Based on these values the performance measures we consider are:

$$\text{Recall} = \frac{|m_t|}{|d_t|}, \quad \text{RankPower} = \frac{n(n+1)}{2\sum_{i=1}^{n} R_i}.$$

RankPower summarizes the overall performance of an algorithm but an object by object assignment of ranks is naturally more illuminating.

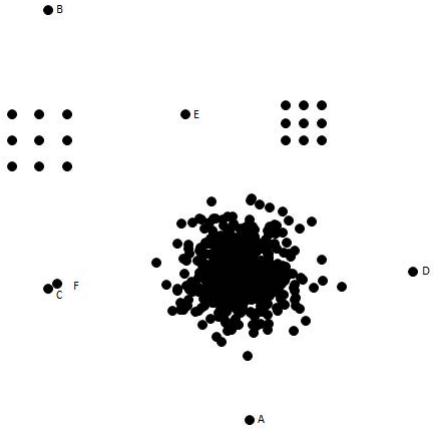


Fig. 4. Synthetic dataset

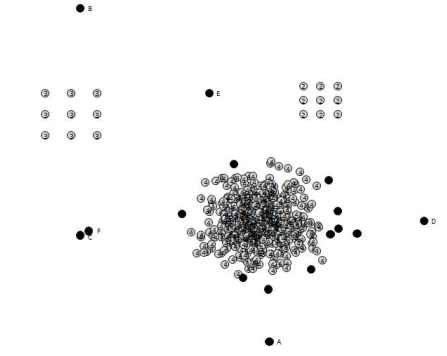


Fig. 5. Synthetic dataset with clusters found by $\mathcal{NC}(6,6)$; black object represents the outliers

4.3 Results

In this section we present a sample of results, extensive tables for all datasets for various values of m and k are available in the Appendix of the technical report [6]. In the first experiment, with planted outliers in the synthetic data set, we observed that RBDA, ODMR, ODMRD, and RADA were able to detect all six outliers. Algorithm DBCOD detected only three of them, namely, $\{A, B, D\}$.

In Table 1 we compare RBDA, ODMR, ODMRD, RADA, and DBCOD for ionosphere dataset with rare class. It is observed that with respect to metrics m_t and Re algorithms RBDA, ODMR, ODMRD, and RADA perform equally well whereas DBCOD does not perform well for $k \leq 30$. RankPower is more discriminatory metric. Using this metric, behavior of algorithms can be summarized as:

$$\text{ODMRD} \geq \text{RADA} \geq \text{RBDA} \geq \text{ODMR} \geq \text{DBCOD} \tag{1}$$

where by $\geq$ we indicate a better performance.

In Table 2 we compare the algorithms for the Wisconsin with rare class. For this dataset no algorithm dominates, however, in general, our algorithms do better than DBCOD.

To determine if any algorithm does better than others in most of the cases, in Table 3 we summarize the performance ranks of the algorithms. These performance ranks were obtained as follows. For a fixed k,and m_t equal to maximum number of outliers in the data set we calculate the RankPower of each algorithm and assign it a value as in equation 1, 1 if it has the best performance, and 5 when the algorithm has the worst performance. Values of k were chosen between 5% to 10% of the size of datasets. The entries in the body of table 3 are averages over all values of k. It can be seen that RADA has the best overall performance.

Table 1. Performance measures of RBDA, ODMR , ODMRD, RADA , and DBCOD for ionosphere dataset

m	RBDA			ODMR			ODMRD			RADA			DBCOD		
	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP
5	5	0.5	1	5	0.5	1	5	0.5	1	5	0.5	1	0	0	0
15	8	0.8	0.783	8	0.8	0.783	8	0.8	0.818	8	0.8	0.818	0	0	0
30	9	0.9	0.703	9	0.9	0.682	9	0.9	0.726	9	0.9	0.726	0	0	0
60	9	0.9	0.703	9	0.9	0.682	9	0.9	0.726	9	0.9	0.726	9	0.9	0.091
85	10	1	0.369	10	1	0.364	10	1	0.390	10	1	0.387	10	1	0.098

Table 2. Comparison of RBDA, ODMR, ODMRD, RADA and DBCOD for k= 7 for the Wisconsin dataset with rare class. Maximum values are marked as red.

m	RBDA			ODMR			ODMRD			RADA			DBCOD		
	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP	m_t	Re	RP
15	9	0.9	0.714	7	0.7	0.8	8	0.8	0.8	8	0.8	0.8	9	0.9	0.662
25	10	1	0.64	10	1	0.611	10	1	0.618	10	1	0.64	9	0.9	0.662
40	10	1	0.64	10	1	0.611	10	1	0.618	10	1	0.64	10	1	0.545

Table 3. Summary of RBDA, ODMR, ODMRD, RADA and DBCOD for all experiments

Dataset	RBDA	ODMR	ODMRD	RADA	DBCOD
Synthetic	3.00	1.00	1.00	1.00	5.00
Iris with rare class	2.67	2.00	2.00	2.33	5.00
Ionosphere with rare class	3.80	3.20	1.20	1.80	5.00
Wisconsin with rare class	3.33	3.00	3.67	1.67	2.67
Iris with outliers	1.00	1.00	1.00	1.00	1.00
Ionosphere with outliers	3.00	3.00	1.50	1.00	5.00
Wisconsin with outliers	1.00	1.00	1.00	1.00	5.00
Summary	2.54	2.03	1.62	**1.40**	4.10

Numbers in the table represent the average performance rank of the algorithms; a small value implies better performance.

5 Conclusion

We observe that rank based approach is highly influenced by the density of neighboring cluster. Furthermore, by definition, ranks use the relative distances and ignore the 'true' distances between the observations. An outlier detection algorithm benefits from 'true' distance as well. Thus we introduce distance in RBDA and observe that the overall performance of RADA is much better than the original RBDA. That the 'true' distance plays an important role is further confirmed

by the performance of the alternative algorithms ODMR and ODMRD; it is observed that in general ODMRD performs better than ODMR. We plan to further investigate the proposed algorithms for robustness and consistency.

References

1. Breunig, M.M., Kriegel, H.-P., Ng, R.T., Sander, J.: Lof: Identifying density-based local outliers. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 93–104. ACM Press (2000)
2. Cao, H., Si, G., Zhang, Y., Jia, L.: Enhancing effectiveness of density-based outlier mining scheme with density-similarity-neighbor-based outlier factor. Expert Systems with Applications: An International Journal 37(12) (December 2010)
3. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys 41(3) Article 15 (July 2009)
4. Feng, J., Sui, Y., Cao, C.: Some issues about outlier detection in rough set theory. Expert Systems with Applications 36(3), 4680–4687 (2009)
5. Huang, H., Mehrotra, K., Mohan, C.K.: Rank-based outlier detection. Journal of Statistical Computation and Simulation 82(11), 1–14 (2011)
6. Huang, H., Mehrotra, K., Mohan, C.K.: Outlier detection using modified-ranks and other variations. Technical Report number SYR-EECS-2011-12, Department of EECS, Syracuse University, Syracuse University, Syracuse, NY, USA (November 2011)
7. Jin, W., Tung, A.K.H., Han, J., Wang, W.: Ranking Outliers Using Symmetric Neighborhood Relationship. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 577–593. Springer, Heidelberg (2006)
8. Tang, J., Chen, Z., Fu, A.W.-C., Cheung, D.W.: Enhancing Effectiveness of Outlier Detections for Low Density Patterns. In: Chen, M.-S., Yu, P.S., Liu, B. (eds.) PAKDD 2002. LNCS (LNAI), vol. 2336, pp. 535–548. Springer, Heidelberg (2002)
9. Tao, Y., Pi, D.: Unifying density-based clustering and outlier detection. In: Proceedings of the Second International Workshop on Knowledge Discovery and Data Mining, WKDD 2009, Moscow, Russia, January 23-25, pp. 644–647. IEEE Computer Society (2009) ISBN 978-0-7695-3543-2

Grouping Co-occurrence Filtering Based on Bayesian Filtering

Takuya Yoshimura[1], Yutaro Fujii[2], and Takayuki Ito[3]

[1] Department of Computer Science and Engineering, Nagoya Institute of Technology
yoshimura.takuya@itolab.nitech.ac.jp
[2] Master course of Techno-Business Administration (MTBA),
Nagoya Institute of Technology
fujii@itolab.nitech.ac.jp
[3] Master course of Techno-Business Administration (MTBA),
Department of Computer Science and Engineering,
Graduate School of Engineering, Nagoya Institute of Technology
ito.takayuki@nitech.ac.jp

Abstract. Recently, many people are using communication tools on the Web, but some send harmful information to others. Most operators manually deal with harmful information, which is expensive. In this paper, we implement two-word co-occurrence filtering by applying the Bayesian filtering method as a spam filter. We propose grouping co-occurrence filtering based on Bayesian filtering and experimentally verify our approach. Grouping co-occurence filtering detect harmful or safe documents at low cost. Our result suggests that grouping co-occurrence filtering is more stable and has a higher accuracy than co-occurrence filtering baesd on Bayesian filtering.

Keywords: Co-occurrence, Bayesian Filtering, Gray Robinson.

1 Introduction

Such communication tools as social network services, bulletin board systems, and blogs continue to increase on the Web. But this information can contain such harmful adult content. The Japanese Ministry of Internal Affairs and Communications (MIC) requested that cellular phone companies address this problem [1]. Most web sites identify harmful content after receiving it manually. A sufficient strategy for dealing with harmful documents doesn't exist. Manual processing does not satisfy demand because it is too expensive to divide information into harmful or safe categories. Therefore, many studies automatically detect harmful documents.

Three conventional filtering methods are currently being used. One is called the Whitelist Method, which lists safe websites. A website can be accessed only when it is included on the list. Another is called the Black List Method, which lists websites that can't be accessed. Finally, the last method stores words called

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 30–39, 2012.

stop words and denies access to websites that include any stop words. But stop words are subjectively defined.

The Blacklist and Whitelist Methods may restrict the entire site. Since all websites must be checked manually, it is difficult to deal with words whose meanings gradually change on the Web. The Stop Word Method is also expensive because we have to select criteria for the stop words. Moreover, the selection criteria will not cover all the documents, which might include slang or secret words. These filtering methods are not practical.

In this paper, several methods are used to extract the feature values of each document and to automatically detect harmful documents. We suppose a method to group co-occurrences by applying existing filtering and consider the availability of the filter based on comparative experiments. We focus on Paul Graham [2][3] and *Gray Robinson* [4][5] filtering to implement a Bayesian filter that resembles a spam filter.

2 Related Works

The implementations of filtering methods are based on Bayesian filtering, which uses a word's frequency or its co-occurrence. Bayesian filtering effectively filters spam. Its applications include *Gray Robinson* and *Paul Graham*. This paper implements filtering methods using these two methods.

2.1 Paul Graham Filtering

Paul Graham Filtering retrieves a documents feature values using the feature values of 15 words. If the document feature value exceeds a set threshold, the document is detected as harmful:

$$p(w_i) = \frac{\frac{b_i}{N_{bad}}}{a \times \frac{g_i}{N_{good}} + \frac{b_i}{N_{bad}}} \tag{1}$$

Formula (1) shows a word's spam probability. w_i is a word. g_i is the number of its occurrences in safe documents. b_i is the number of its occurrences in harmful documents. N_{good} is the sum of all occurrences in safe documents contained in a training dataset. N_{bad} is the sum of all occurrences in harmful documents. By applying (1) to Formula (2), we obtain the following expression. a is a bias value that reduces the false detection rate of safe documents.

$$p(D) = \frac{\prod_{i=1}^{n} p(w_i)}{\prod_{i=1}^{n} p(w_i) + \prod_{i=1}^{n} (1 - p(w_i))} \tag{2}$$

$p(D)$ is a feature value of document D, and we get the value using the spam probabilities of words. When the feature value exceeds a set threshold, document D is identified as harmful.

2.2 Gray Robinson Filtering

Gray Robinson Filtering, which is an improved form of Paul Graham filtering, can express the spam probability of words that only appear in harmful documents. There is no threshold for detection, so neither safe nor harmful documents are detected:

$$f(w_i) = \frac{s \cdot x + n \cdot p(w_i)}{n + s} \tag{3}$$

$f(w_i)$ expresses the only appearance of a word in a harmful document, and we get the value of accordance with statistical learning. x is the prior probability of a word that has never appeared. s shows the strength given to x. n is the number of occurrences of word w_i. Based on the subsequent formulas, we reach an indicator to detect harmful documents:

$$H(D) = 1 - \{\prod_{i=1}^{n}(1 - f(w_i))\}^{\frac{1}{n}} \tag{4}$$

$$S(D) = 1 - \{\prod_{i=1}^{n} f(w_i)\}^{\frac{1}{n}} \tag{5}$$

$$I_1 = \frac{S - H}{S + H} \tag{6}$$

$$I_2 = \frac{I_1 + 1}{2} \tag{7}$$

$H(D)$ is the hamminess value of document D, and $S(D)$ is its spamminess value. I_1 and I_2 are indicators. We get I_1 with $S(D)$ and $H(D)$. Formula (7) shows the indicator that limits the scope's value. By applying I_1 to (7), we limited the document feature value to a scope of 0 to 1. If I_2 is closer to 0.5, the document is detected as neither harmful nor safe.

2.3 Co-occurrence Filtering

In Bayesian filtering, documents are detected using the spam probability of words. In co-occurrence filtering, co-occurrences are used instead of words. Word w_i of subsections 3.1 and 3.2 is changed to co-occurrence c_i:

$$p(c_i) = \frac{\frac{b_i}{N_{bad}}}{a \times \frac{g_i}{N_{good}} + \frac{b_i}{N_{bad}}} \tag{8}$$

3 Databases

Datasets are gathered for creating databases, which store frequency information to retrieve feature values. This section describes the processes of gathering datasets and creating databases.

3.1 Gathering Datasets

To detect harmful documents, a computer needs to be trained on datasets of harmful and safe examples. We gathered a dataset from 2ch (Japan's largest bulletin board site) using a web crawler and created a 66.3-MB filtering database, and one average document size is 3.5 KB. The gathered dataset was divided into safe and harmful documents. Fig. 1 shows how we gathered datasets and divided them into harmful and safe. Some documents were detected as harmful using the Stop Word Method, and the others were checked manually.

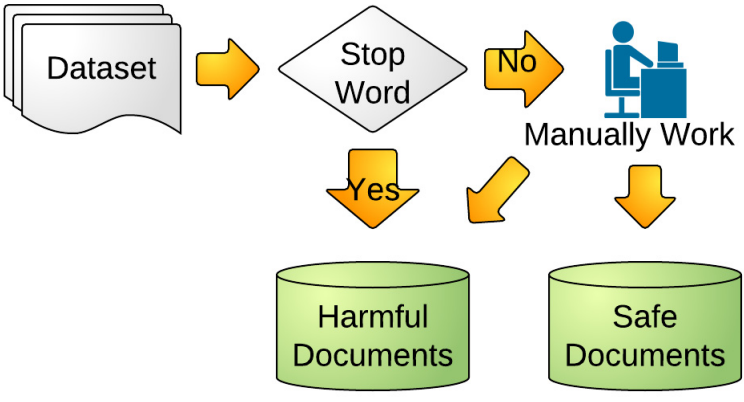


Fig. 1. Gathering datasets

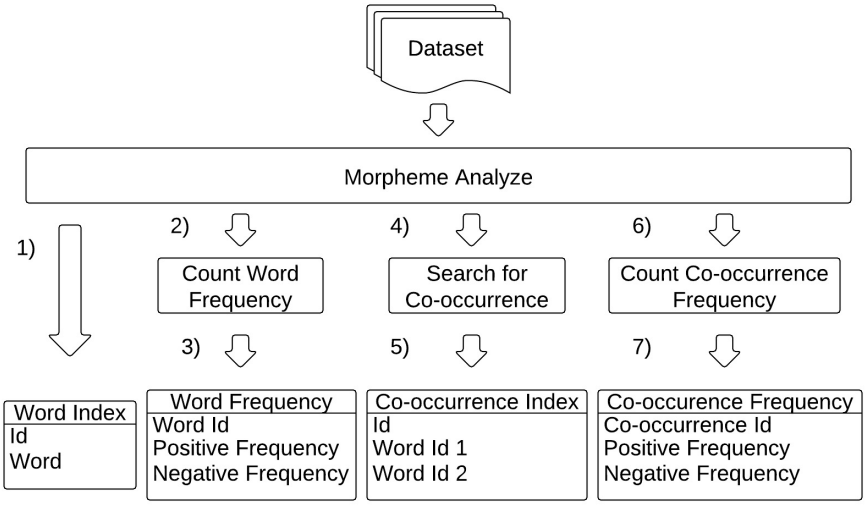


Fig. 2. Processing of creating database

3.2 Creating Databases

We created two kinds of databases for Bayesian and co-occurrence filtering. In the former, the database was created by word frequency. In the latter, it was created by a pair or more of words. Figure 2 shows the flow of creating a database. First, the morphemes of the datasets are analyzed. This paper used MeCab[16], which is a Japanese morphological analyzer. Step 1 creates a database that has two fields which are a word and an id. This database is called the word index. Step 2 separately counts the frequency of each word in harmful or safe documents. In Step 3, a word frequency database is created. A word Id, which is the field of the word frequency database, is associated with the word index. Step 4 retrieves all co-occurrences composed of morpheme pairs. Step 5 creates a database that has words associated with the word Id. Step 6 separately counts the co-occurrence frequency and determines whether a document is harmful or safe. Step 7 creates a co-occurrence frequency database that has a field that is co-occurrence Id of the co-occurrence index.

4 Grouping Co-occurrence Filtering

We imagine a new Grouping Co-occurrence filtering method that retrieves the feature values of the co-occurrence group, which means a set of co-occurrences that contain one word. Fig. 3 provides three co-occurences groups, $Time$ group, is group and $money$ group, at the example sentence of "Time is money". The sentence has three co-occurrence groups. First, a $Time$ group has two co-occurrences that contain $Time$ $\{Time, is\}$ and $\{Time, money\}$. Another group contains two co-occurrences $\{is, Time\}$ and $\{is, money\}$. Finally, a co-occurrence contains $money$ as $\{money, Time\}$ and $\{money, is\}$:

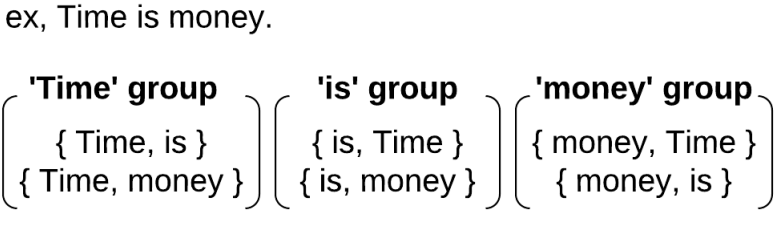


Fig. 3. Co-occurrence group

Grouping co-occurrence filtering retrives the co-occurrence group feature value by using average of each co-occurence spam probability. Formula (9) averages $p(co_i)$ over the n is the number of groups, taking example three. The filtering need to get the average of occurance count of each co-occurrence, Formula (10) averages the co-occurrences. Formula (11) provides the co-occurence group

feature value. We get the value by applying p_A and n_A. p_A is given in Formula (9), and n_A is given in Formula (10). By applying (11) to Formulas (4) and (5) at the *Gray Robinson*'s method, we get the feature values of documents.

$$p_A = \frac{\sum_{i=1}^{n} p(co_i)}{n} \tag{9}$$

$$n_A = \frac{\sum_{i=1}^{n} (g_i + b_i)}{n} \tag{10}$$

$$f_A = \frac{s \cdot x + p_A \cdot n_A}{s + n} \tag{11}$$

The co-occurrence filtering based on *Gray Robinson*'s method must set parameters to exclude noise of co-occurrence, and these parameters are set by a person. Noise of a co-occurence is a ambiguous co-occurence that a filter cannot find out whether its harmful or safe. The filter does not consider noise, but the filtering function is not lost. Conversely, the filtering performance is improved because the filter regard many documents as ambiguous documents if all noise are considered for filtering. So, it is expensive to set parameters until we find the best parameters for training datasets.

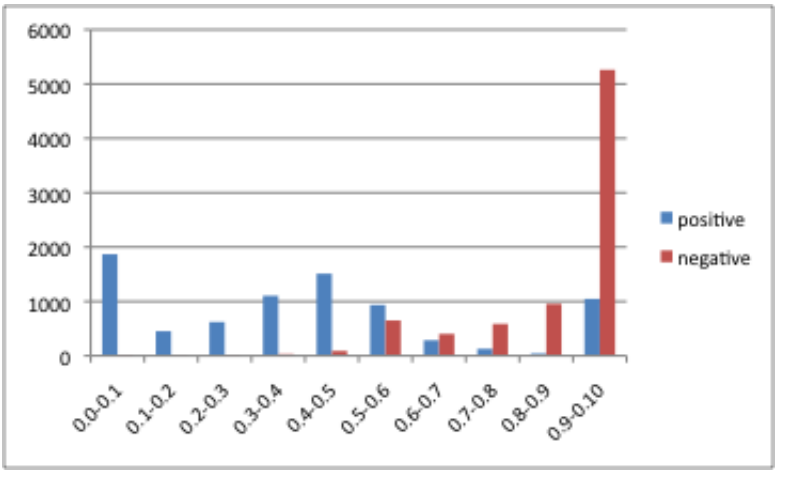
Grouping co-occurrence filtering method does not need setting the parameter, instead it can detect harmful documents by smoothing noise of co-occurrences without setting parameters. We can reduce the work to create filtering because the filter removes the process of setting parameters. So, by using the Grouping Co-occurrence filtering, we can detect harmful or safe documents at low cost.

Meanwhile, accuracy of each filtering must be verified because it is unclear whether grouping co-occurrence filtering works as filtering. We describe each accuracy in the next section of Experiments.

5 Experiments

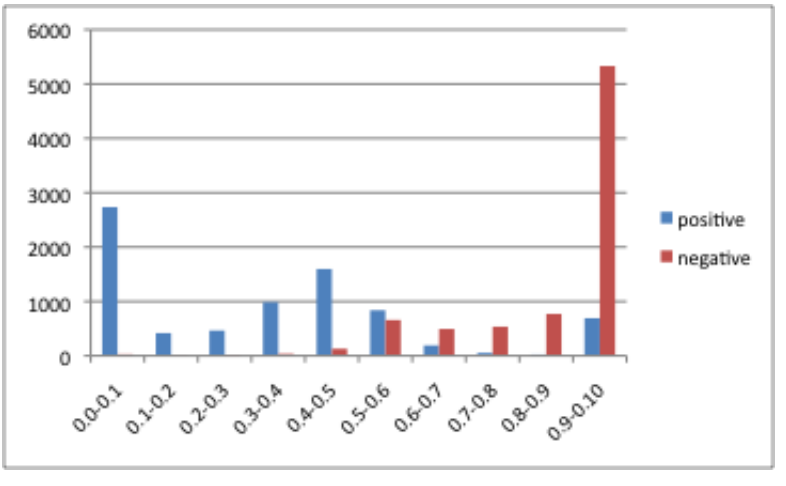
This paper conducted a verification experiment to clarify differences from grouping co-occurrence filtering and co-occurrence filtering. Co-occurrence filtering outperforms Bayesian filtering [10]. This section describes the differences of filtering from the experimental results obtained from a graph that shows the detection results.

We conducted the experiments with 8,000 documents. Fig. 4 is the result of co-occurrece filtering with the parameters set to remove noise co-occurences within a scope of 0.10 to 0.90. Fig. 5 shows the result of co-occurrence filtering set in parameters to remove noise within a scope of 0.07 to 0.953. A both filtering rates of the scope of 0.9 to 1.0 is the highest for harmful documents. Converserly, for safe documents, both rates of scope of 0.0 to 0.1 is the highest.



	positive	negative
safe documents	5864	2136
harmful documents	532	7468

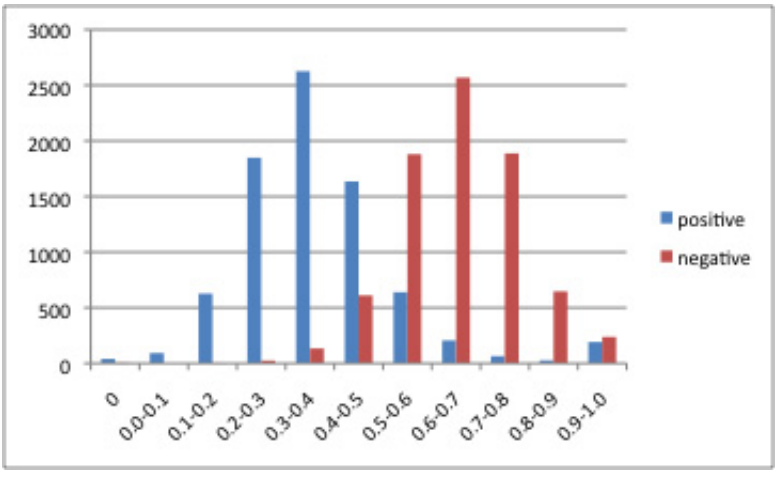
Fig. 4. Co-occurrence filtering with Gray Robinson



	positive	negative
safe documents	6548	1452
harmful documnets	468	7532

Fig. 5. Co-occurrence filtering with Gray Robinson

Fig. 5 is a result of altering Fig. 4's parameter to improve accuracy. Fig. 4 and Fig. 5 find that we must search for the best parameters for the filtering because accuracy changed according to parameters. Fig. 6 shows the detection result of the grouping co-occurrence filtering. Safe documents have a mountain distribution. The different results in the method show how to dispose of noise. In the *Gray Robinson* method, the scope of a feature value is disposed of as a noise word, but grouping co-occurrence filtering disposes of noise by smoothing co-occurence feature values. Table 1 shows the recall, precision, and f-score of the grouping co-occurrence and co-occurrence filtering for detecting safe documents. Grouping co-occurrence filtering was applied, so the f-score was higher than the others. Table 2 shows also three rates for detecting harmful documents. Co-occurrence 2's F-score is as same as grouping co-occurrence's F-score, but co-occurence 2 has a larger difference between recall and precision than grouping co-occurence. So, grouping co-occurence is stable as filtering because of balanced result.



	positive	negative
safe documents	6870	1130
harmful documents	778	7222

Fig. 6. Grouping co-occurrence filtering with Gray Robinson

Table 1. Co-occurrence and grouping co-occurrence filterings of recall, precision and F-score for safe documents

	recall	precision	f-score
Co-occurrence 1	77.3%	91.7%	0.815
Co-occurrence 2	81.9%	93.3%	0.872
Grouping co-occurrence	85.9%	89.8%	0.878

Table 2. Co-occurrence and grouping co-occurrence filterings of recall, precision and F-score for harmful documents

	recall	precision	f-score
Co-occurrence 1	93.3%	77.7%	0.848
Co-occurrence 2	93.3%	83.3%	0.883
Grouping co-occurrence	90.3%	86.4%	0.883

6 Conclusion

We gathered datasets to implement the two Filtering, and implemented the Grouping filtering and Co-occurrence filtering. Since two filtering differ to deal with noisy co-occurrence, we verify each feature and accuracy as a filtering method. Our experimental results found that grouping co-occurrence filtering has higher accuracy than *Gray Robinson* filtering. Grouping filtering does not dispose of all noise; it smoothes noise. Grouping filtering can get several feature values and expresses more ambiguity and detecte harmful or safe documents without setting parameters. Grouping filtering method provides a fine setting and is a practical classifier. In future work, we will verify co-occurrence filtering by using more data sets. When the training data are insufficient, we must improve the accuracy.

References

1. Cellular phone companies were requested to promote harmful site access restriction services (filtering service) on mobile phones and PHSs, `http://www.soumu.go.jp/menu_news/s-news/2007/071210_4.html`
2. Graham, P.: A plan for spam. In: Graham, P. (ed.) Hackers and Painters. OReilly (2004)
3. Graham, P.: Better bayesian filtering. In: Proceedings of the 2003 Spam Conference (2003)
4. Gray, R.: A statistical approach to the spam problem, `http://www.linuxjournal.com/article/6467/`
5. Gray, R.: Spam detection (2002), `http://radio.weblogs.com/0101454/stories/2002/09/16/spamDetection.html`
6. Mera, K., Ichimura, T., Aizawa, T., Yamashita, T.: Invoking emotions in a dialog system based on word-impressions. Trans. Japanese Society for Artificial Intelligence 173, 186–195 (2002)
7. Nagata, M., Taira, H.: Text classification, Trade fair of learning theories. IPSJ Magazine 42(1), 32–37 (2001)
8. Tsuda, Y.: Text Categorization Using Native Bayes Model Based on Co-Occurrence words. In: Symposium on Information Theory and its Applications, SITA (2006)
9. Manning, C.D., Schutze, H.: Foundations of statistical natural language perspectives. Oxford Univ. Press, New York (1999); Guage processing. MIT Press, Cambridge (1999)
10. Ando, S., Fujii, Y., Ito, T.: Filtering Harmful Sentences based on Multiple Word Co-occurrence. In: 2010 IEEE/ACIS 9th International Conference on Computer and Information Science (2010)

11. Matsuo, Y., Ishizuka, M.: Keyword Extraction from a Document using Word Co-occurrence Statistical Information. Transactions of the Japanese Society for Artificial Intelligence 17(3), 217–223
12. Cortes, C., Vapnik, V.: Support Vector Networks. Machine Learning 20, 273–297 (1995)
13. Kobayashi, D., Ishizuka, M.: Classification of Spam Posts on Knowledge Searching Website. In: The 21st Annual Conference of the JSAI (2007)
14. Kumamoto, T., Tanaka, K.: Proposal of Impression Mining from News Articles. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005, Part I. LNCS (LNAI), vol. 3681, pp. 901–910. Springer, Heidelberg (2005)
15. Turney, P.D.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proc. Conference on ACL (2002)
16. Mecab, `http://mecab.sourceforge.net/`

Model Fusion-Based Batch Learning with Application to Oil Spills Detection

Chunsheng Yang[1], Yubin Yang[2], and Jie Liu[3]

[1] Institute for Information Technology, National Research Council Canada
Chunsheng.Yang@nrc.gc.ca
[2] State Key Laboratory for Novel Software Technology, Nanjing University
yangyubin@nju.edu.cn
[3] Dept.of Mechanical and Aerospace Engineering, Carleton University, Canada
jliu@mae.carleton.ca

Abstract. Data split into batches is very common in real-world applications. In speech recognition and handwriting identification, the batches are different people. In areas like oil spill detection and train wheel failure prediction, the batches are the particular circumstances when the readings were recorded. The recent research has proved that it is important to respect the batch structure when learning models for batched data. We believe that such a batch structure is also an opportunity that can be exploited in the learning process. In this paper, we investigated the novel method for dealing with the batched data. We applied the developed batch learning techniques to detect oil spills using radar images collected from satellite stations. This paper reports some progress on the proposed batch learning method and the preliminary results obtained from oil spills detection.

Keywords: Batch Data, Batch Learning, Transfer Learning, Content-based Learning, Model Fusion, Oil Spill Detection.

1 Introduction

In real-world applications, many data are split into batches, and each batch may have its feature space and distribution. For instance, in speech recognition and handwriting identification, the batches are different people. In the area of oil spill detection [1] the batches are the radar images collected from different satellite stations. In the area like train wheel failure predictions [2], the batches are the particular circumstance from different trains. We believe that it is necessary and important to respect batched structure and its characteristics when learning and evaluating algorithms for batched data in real-world applications. Unfortunately, most existing machine learning or data mining techniques/algorithms assume that the data used in training and in the testing has the same feature space and the same distribution. Such an assumption in many real-world applications hardly holds. Recently, many efforts have been put to transfer learning [7]. Basically, transfer learning resolves the issue that training date and future deployment environment (or so-called testing data) may have different distributions or feature space. Transfer learning does not deal with issues in batched

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 40–47, 2012.

data directly. In this work, we attempt to address this issue and investigate an effective method for batch learning by respecting batched data structure and its characteristics represented by features instead of attributes. We are motivated by Kubat et al.'s work on oil spills detection using radar imaged collected from satellite stations. We proposed a model fusion-based method for batch learning and applied it to oil spills detection. In this short paper, we report the on-going progress along with preliminary results and mainly discus the future directions on batch learning.

The rest of this paper is organized as follows. Section 2 presents a problem formulation on batch learning by reviewing related work such as transfer learning; Section 3 introduces the proposed method; Section 4 outlines the experiments conducted in oil spill detection along with some preliminary results; Section 5 discusses the limitation and provides the future direction; the final section concludes the paper.

2 Batch Learning and Related Work

Over the past decades, many researchers have paid attention to transfer learning [7] which is closely related to batch learning. Transfer learning also has different names in machine learning community, including learning to learning, meta-learning, multitask learning [8], and context-sensitive learning [9], robust learning etc. All of these work tried to address the learning problems from one domain to another domain. In other words, these techniques learn a model in a source domain and apply it to a target domain which may have different feature space or distinct distribution. The ultimate goal is to maximize generalization ability of the learn models. In term of the definition given in [7], transfer learning is defined as follows. Given two different domains: source domain (D_s) and target domain (D_T) with learning task T_s and T_T, respectively , transfer learning aims at improving the ability of learning algorithms in D_T using the knowledge in D_T and T_s where $D_s \neq D_T$ and $T_s \neq T_T$. In terms of the definition, there are three different settings which lead to three main transfer learning techniques: inductive transfer learning, transductive transfer learning, and unsupervised transfer learning. Inductive transfer learning deals with the learning issues where the learning tasks are different no matter what domains are the same or not; transductive transfer learning focuses on the problems where domains are different but the target tasks are the same; and unsupervised transfer learning are similar with inductive learning, but the learning tasks in target domain is related to ones in the source domain.

However, batch learning is different from transfer learning. Batch learning is defined as follows. Given that a domain is decomposed into multiple sub-domains expressed with batched data, batched learning is to find the best learning algorithm for domain problem. If we note the batched data as D_1, D_2, D_i… D_n (n is the batch number) and each batch data has its own feature space(note as F_i) and probability distribution (fd_i), batch learning aims to find the best learning algorithm (noted as $f(\cdot)$), $f(\cdot) = \mathrm{F}(F_1, F_2, F_i \ldots F_n)$, where $F_i \neq F_j$ or $fd_i \neq fd_j$. In batch learning, the learning task is always the same. For example, in oil spills detection, the learning task

is always to classify the images into positives or negatives. To respect the batched data structure in developing learning algorithms or techniques, most research work published focused on feature transformation or weighted features in learning process and to build a global classifier to resolve the batched problem. For example, Vural et al. [6] developed a batch classifier with different algorithms: probabilistic analysis and mathematical programming, and applied the developed batch classifier to medical domain: to classify the medical images such as CT scan and MRI image. Similarly, some other research also focused on developing a global model for batch learning [4, 5]. In this work we propose a model fusion-based method by respecting the nature of batched data and developing model for each individual batch. In the following section, we detail the model fusion-based method for batch learning.

3 Model Fusion-Based Framework

We assume that the features extracted from each batch may be different or distribution may be different even through the attribute space in each batch may be identical. We also assume that the batched data covers sufficient samples for the problem space given a domain. Building on techniques from ensemble classifier or model fusion, we developed a model fusion-based method for batch learning. The idea is to build a model (noted as m_i) or classifier for each batch by using the extracted features and then build a batch identifier (noted as m^b) for batch identification using all dataset. The developed method is described in Table 1 using the notation defined above. The method consists of three main steps: feature extraction($F^x(Di)$), batch model building (*BuildModel()*), and batch identifier building (*BuildBatchIdentifier()*).

Table 1. The Model fusion-based method for batch learning

Input: *A batched dataset* $DS = \{D_1, D_2, D_3 ... D_i ... D_n\}$

Output : *models and batch identifier (* m_i *and* m^b *)*

```
Process: {
  For  each Di in  DS  {
       Fi = F^x(Di)                                   /* extracting features for each batch  */
           For  adopted  algorithm j {
                mij = BuildModel( Fi,  algorithm j )     /* building model for each batch */
           }
       mi= ModelSelection(mi1, mi2,... mij);
     }
     For specified classifier algorithm i {
       LabelAllbatcheddata(DS)            /* label batched data as a N-class classification task */
       m^b = BuildBatchIdentifier(algorithm i, DS )     /* building batch identifier*/
     }
    OutputModels(mi, m^b )
}
```

First step is to extract the features which represent the batch characteristics for each batch data. We developed the automated approach that could help further improve the data representation in a batch by creating new powerful features that combine the initial ones. For each batch the created new features may be different, which make it possible to respect the batched data structure in batch learning.

The second step is to independently build model for each batch using the extracted features. To do so, many learning algorithms are available including Instance-Based learning (IB*k*), TFIDF classifier, Naïve Bayes, Support Vector Machine (SVM), Decision Trees, and Neural Networks. In this work, we tend to prefer simple algorithms such as Decision Trees and Naïve Bayes over more complex ones because they are quick and produce models that we can easily explain for each batch. For the built models we evaluate their performance with traditional matrix such as accuracy and select the best one for each batch.

Third step is to build a high-performance classifier as batch identifier. This is N-class classifier. We first label all data in the batched dataset. In other words, we label the data in batch D_i as the class '***i***' , and we will have N-class dataset finally. Using this dataset, we can build a N-class classifier as the batch identifier. We just need to repeat the model building process in the second step and find one best classifier for given batched data.

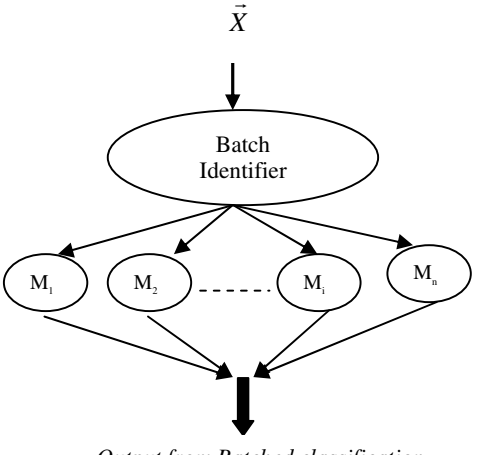


Fig. 1. The structure of Batched Classification Systems Data Mining-Based Fault Identification Framework

After developing the model for each batch and building a batch identifier, we can build a model fusion-based classification system for real-world applications. The Figure 1 shows such a classification system for batched applications. Given an observation ($\vec{X}$), it will be identified first as which batch it belongs to. Then corresponding batch model will be selected to classify the given observation and final result will be generated from the batched classification systems.

4 Experiments and Results

In this section, we demonstrate how the developed batch learning technique works by applying it to oil spills detection. We start from the brief description of the oil spills detection application, then present the experiment conducted along with some preliminary results.

4.1 The Application Domain

Oil spill is an important environment problem. It is interesting the majority of oil spills is caused by ships which depose of oil residues in their tanks during navigation [1]. Radar images collected from satellites stations provide a vital resource to detect oil spills. Such a radar image contains 8,000X8,000 pixels with each pixel representing a square of 30X30m. Oil slicks will appear dark in the image and its size and shape change in time depending on weather and sea conditions. In terms of the nature of image data, such image data is batched. Those images can be used to detect oil spill. Kubat et al. [1] applied machine learning techniques to detect oil spills, pushing oil spills detection toward advanced step from manual detection. In the early satellite image service such as the Tromso Satellite Station (TSS) service in Norway, the oil spills are distinguished by well-trained human expert who can distinct the images between non-spills and spills. Kubat et al. built one global model (classifier) for nine batched data. Even though the developed classifier could help to detect oil spills with much better accuracy than human does, Kubat et al still pointed out the new methods for batch learning are demanded. In this work, we applied our proposed batch learning technique to build a model fusion-based classification system for oil spill detection.

4.2 Experiments and Results

We have obtained the dataset for oil spill detection. This dataset consists of 9 batches. The Table 2 shows the composition of each batch: the number of positive and negative instances for each batch. Each batch data contains 54 attributes covering three groups: target measurements, relative measurements, and non-imagery information.

Table 2. The numbers of positive and negative instances in the batched image data

Batched Images	1	2	3	4	5	6	7	8	9	All
Positives	8	4	2	6	2	4	3	5	7	41
Negatives	3	180	101	129	60	70	76	80	197	896
Total	11	184	103	135	62	74	79	85	204	937

We first extracted the correlated features from each batch for building models for individual batch. In this work, we employed decision tree (J48) and Naïve Bayes as the learning algorithms for developing model for each batch.

In order to evaluate the effectiveness of the model fusion-based approach for batch learning, we conducted two different experiments: developing model fusion-based classification systems and building one global model for all batches. To compare the performance of two different systems, we performed evaluation using cross valuation method. Because of small size of the data, we only used 5 folds cross validation. Table 2 shows the results. We calculated the accuracy for each batch and a total for all

batches. From the Table2, it is obvious that model fusion-based approach outperforms one global model approach for both selected learning algorithms.

Table 3. The accuracy of learning algorithms, J48 and Naïve Bayes

Batches	J48 (Decision Tree)		Naïve Bayes	
	Model Fusion	One Model	Model Fusion	One Model
1	0.91	0.27	0.91	0.27
2	0.995	0.17	0.96	0.08
3	1.0	0.09	1.0	0.78
4	0.99	0.14	0.92	0.04
5	1.0	0.95	1.0	0.71
6	0.98	0.89	0.74	0.93
7	0.98	0.95	0.93	0.61
8	0.99	0.98	0.93	0.87
9	0.99	0.90	0.96	0.95
Total	0.99	0.77	0.94	0.62

5 Discussions and Future Work

The results from oil spills detection demonstrated the effectiveness of the proposed model fusion-based method for batch learning. By respecting the batched data structure in the real-world application, batch learning is a useful technique for batched data. Even though we emphases the importance of batch learning though oil spills detection application, the issues facing us still is a generic one in machine learning. This paper just reported some progress. Many tasks are on-going and will continuously be our future work as well. These issues are as follows.

(1) When the number of batches becomes large the proposed method will be more difficult to deal with the batched issue. In the proposed method, the performance of batch learning systems depends largely on the performance of a batch identifier. The larger the number of batches, the bigger the number of classes for N-class classifier. This will increase the difficulty to build a high-performance classifier. In oil spill detection, we developed a 9-class classifier to identify an observation into one of nine batches. The results shown in Table 2 rely on the 98% accuracy of 9-class batch identifier in this application. However, it is not easy to build such a high-performance classifier with the increasing of the number of batches. To address this issue, we can combine the transfer learning technique and model fusion-based method. We can reduce the batch numbers by combining or grouping the number of batches into a new batch with the help of transfer learning technique. In other words, after examining the feature space and their distribution of the batches, we can transform the feature space or distribution to get a unified feature space or identical distribution for some batches, such that the number of batches will be reduced.

(2) In this work, we developed the model for each batch using the same learning algorithm either J48 or Naïve Bayes. In fact, we can evaluate different learning algorithms for each individual batch to find out the best matching model. This will be the next experiments to look at the model diversity issue. To this end, we need to use more sophisticated model evaluation method such as ROC curve and cost-based curve and integrate the batched feature into evaluation as well.
(3) When we investigated batch leaning techniques for batched data, we assume that the number of batches (the number of sub-problems) is known from real-world applications. For example, the oil spills detection contains nine batches. This assumption may hardly hold because new batch may appear anytime. In such a case, the proposed method has difficult to deal with new batch. We have to rebuild model for new batch and rebuild batch identifier as well. To address this issue, robust learning technique is demanded. Therefore, we are looking into the method for integrating robust leaning with batch learning.

6 Conclusions

In this paper, we emphasized that it is necessary to deal with batched data by respecting the batched data structure. Building the techniques of classifier ensemble and machine learning, we developed a model fusion-based method for batch learning. We also applied the proposed method to a real-world application: to detect oil spills using the batched radar images data. The preliminary results demonstrated the feasibility and usefulness of the proposed method for batch learning. As we mentioned, we only reported some progress for our on-going research on batch learning. We also discussed several critical issues in dealing with batch learning and these issues will be our future work.

Acknowledgment. The authors would like to thank Dr. Robert C. Holte for providing data from the oil spills detection and allow us to explore it more deeply. We also like to thank Dr. Chris Drummond for his valuable discussion and comments on experiments and future direction. This work is supported by the Natural Science Foundation of China (Grant Nos. 61035003, 61021062, 60875011), the International Science and Technology Cooperation Program of China (Grant No. 2010DFA11030), and the Natural Science Foundation of Jiangsu, China (Grant No. BK2011005, BK2010054).

References

[1] Kubat, M., Holte, R., Matwin, S.: Machine Learning for the Detection of Oil Spills in Satellite Radar Images. Machine Learning 30, 195–215 (1998)
[2] Yang, C., Létourneau, S.: Learning to predict train wheel failures. In: The 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 516–525 (2005)

[3] Bermejo, S., Cabestany, J.: A Batch Learning Vector Quantization Algorithm for Nearest Neighbor Classification. Neural Processing Letters 11, 173–184 (2000)
[4] Kinouchi, M., Takada, N., Kudo, Y., Ikemura, T.: Quick Learning for Batch-learning Self-Organizing Map. Genome Informatics 13, 266–267 (2002)
[5] Wong, K.Y.M., Lou, P., Li, F.: Dynamics of Gradient-based Learning and Application to Hyperparameter Estimation. In: Liu, J., Cheung, Y.-m., Yin, H. (eds.) IDEAL 2003. LNCS, vol. 2690, pp. 369–376. Springer, Heidelberg (2003)
[6] Vural, V., Fung, G., Krishnapuram, B., Dy, J., Rao, B.: Batch Classification with Applications in Computer Aided Diagnosis. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) ECML 2006. LNCS (LNAI), vol. 4212, pp. 449–460. Springer, Heidelberg (2006)
[7] Pan, S.J., Yang, Q.: A Survey on Transfer Learning. IEEE Transaction on Knowledge and Data Engineering 22(10), 1345–1359 (2010)
[8] Caruana, R.: Multitask Learning. Machine Learning 28, 41–75 (1997)
[9] Cohen, W.W., Singer, Y.: Context-Sensitive Learning Methods for Text Categorization. ACM Transactions on Information Systems 17(2), 141–173 (1999)

Efficient Determination of Binary Non-negative Vector Neighbors with Regard to Cosine Similarity

Marzena Kryszkiewicz

Institute of Computer Science, Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw, Poland
mkr@ii.pw.edu.pl

Abstract. The cosine and Tanimoto similarity measures are often and successfully applied in classification, clustering and ranking in chemistry, biology, information retrieval, and text mining. A basic operation in such tasks is identification of neighbors. This operation becomes critical for large high dimensional data. The usage of the triangle inequality property was recently offered to alleviate this problem in the case of applying a distance metric. The triangle inequality holds for the Tanimoto dissimilarity, which functionally determines the Tanimoto similarity, provided the underlying data have a form of vectors with binary non-negative values of attributes. Unfortunately, the triangle inequality holds neither for the cosine similarity measure nor for its corresponding dissimilarity measure. However, in this paper, we propose how to use the triangle inequality property and/or bounds on lengths of neighbor vectors to efficiently determine non-negative binary vectors that are similar with regard to the cosine similarity measure.

Keywords: nearest neighbors, ε-neighborhoods, the cosine similarity, the Tanimoto similarity, data clustering, text clustering.

1 Introduction

The cosine and Tanimoto similarity measures are often and successfully applied in classification, clustering and ranking in chemistry, biology, information retrieval, and text mining. A basic operation in such tasks is identification of neighbors. This operation becomes critical for large high dimensional data. The usage of the triangle inequality property was recently offered to alleviate this problem in the case of applying a distance metric [2-6, 8-9]. The triangle inequality holds for the Tanimoto dissimilarity, which functionally determines the Tanimoto similarity, provided the underlying data have a form of vectors with binary non-negative values of attributes [7]. Unfortunately, the triangle inequality holds neither for the cosine similarity measure nor for its corresponding dissimilarity measure. However, in this paper, we propose how to use the triangle inequality property and/or bounds on lengths of neighbor vectors to efficiently determine binary non-negative vectors that are similar with regard to the cosine similarity measure. More specifically, in this paper, we will consider vectors such that a domain of each of their attributes (or dimensions) is binary and may take

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 48–57, 2012.

either value 0 or a positive real value. When the domains of their all dimensions include only 0 and 1, 0 might denote absence of an attribute, while1 its presence. If a positive value different from 1 is allowed for an attribute, it might reflect the importance (weight) of the occurrence of the attribute. The larger positive attribute value, the more important attribute. In the paper, we will call such vectors *binary non-negative vectors*.

Our paper has the following layout. Section 2 provides basic notions and properties used in the paper. In Section 3, we recall the method as offered in [3-4], which applies the triangle inequality property to efficiently calculate neighborhoods using a distance metric also in the case of large high dimensional datasets. In Section 4, we investigate the relationship between the cosine similarity measure and Tanimoto (dis)similarity. In Section 5, we formulate and prove the bounds on the lengths of cosine similar binary non-negative vectors. In Section 6, we investigate the combined usage of the Tanimoto dissimilarity, the triangle inequality and the found bounds on the length of vectors for determining cosine similarity neighborhoods of binary non-negative vectors. Section 7 concludes our work.

2 Basic Notions and Properties

In the chapter, we will consider vectors of the same dimensionality, say n. A vector u will be sometimes denoted as $[u_1, \ldots, u_n]$, where u_i is the value of the i-th dimension of u, $i = 1..n$.

In the sequel, dissimilarity between two vectors p and q will be denoted by $dis(p, q)$. A vector q is considered as *less dissimilar* from vector p than vector r if $dis(q, p) < dis(r, p)$. In order to compare vectors, one may use a variety of dissimilarity measures among which an important class are *distance metrics*.

A *distance metric* (or shortly, *distance*) in a set of vectors D is defined as a dissimilarity measure $dis : D \times D \rightarrow [0,+\infty)$ that satisfies the following three conditions for all vectors p, q, and r in D:

1) $dis(p, q) = 0$ iff $p = q$;
2) $dis(p, q) = dis(q, p)$;
3) $dis(p, r) \leq dis(p, q) + dis(q, r)$.

The third condition is known as the *triangle inequality property*. Often, an alternative form of this property is used; namely, $dis(p, q) \geq dis(p, r) - dis(q, r)$.

In order to compare vectors, one may alternatively use *similarity measures*. In the following, the similarity between two vectors p and q will be denoted by $sim(p, q)$. A vector q is considered as *more similar* to vector p than vector r if $sim(q, p) > sim(r, p)$. Please note that, for example, $-sim(q, p)$ or $1 - sim(q, p)$ could be interpreted as a measure of dissimilarity between q and p.

Among most popular similarity measures is *cosine similarity* and *Tanimoto similarity*. The *cosine similarity* between vectors u and v is denoted by $cosSim(u, v)$ and is defined as the cosine of the angle between them; that is,

$$cosSim(u,v) = \frac{u \cdot v}{|u||v|}, \text{ where:}$$

- $u \cdot v$ is a *standard vector dot product of vectors u and v* and equals $\sum_{i=1..n} u_i v_i$;
- $|u|$ is *the length of vector u* and equals $\sqrt{u \cdot u}$.

Property 1. Let u and v be non-zero vectors. Then, $cosSim(u, v) \in [-1, 1]$.

The *Tanimoto similarity* between vectors u and v is denoted by $T(u, v)$ and is defined as follows,

$$T(u, v) = \frac{u \cdot v}{u \cdot u + v \cdot v - u \cdot v}.$$

In the case of binary vectors with attribute domains restricted to {0, 1}, the Tanimoto similarity between two vectors determines the ratio of the number of attributes ("1s") shared by both vectors to the number of attributes ("1s") occurring in either vector.

Property 2 [10]. Let u and v be non-zero vectors. Then, $T(u, v) \in \left[-\frac{1}{3}, 1\right]$.

Both the cosine similarity and the Tanimoto similarity do not preserve the triangle inequality property. Also, $1 - cosSim(u, v)$ does not preserve this property. However, it was proved in [7] that the measure $1 - T(u, v)$, known as the *Tanimoto dissimilarity*, preserves the triangle inequality property for binary non-negative vectors.

Below, we provide definitions of neighborhoods in a vector set D with regard to a dissimilarity measure *dis* and, respectively, with regard to a similarity measure *sim*.

ε-neighborhood of a vector p in D w.r.t dissimilarity measure dis is denoted by $\varepsilon\text{-}NB_{dis}^{D}(p)$ and is defined as the set of all vectors in dataset $D \setminus \{p\}$ that are dissimilar from p by no more than ε; that is,

$$\varepsilon\text{-}NB_{dis}^{D}(p) = \{q \in D \setminus \{p\} \mid dis(p, q) \leq \varepsilon\}.$$

ε-similarity neighborhood of a vector p in D w.r.t similarity measure sim is denoted by $\varepsilon\text{-}SNB_{sim}^{D}(p)$ and is defined as the set of all vectors in dataset $D \setminus \{p\}$ that are similar to p by no less than ε; that is,

$$\varepsilon\text{-}SNB_{sim}^{D}(p) = \{q \in D \setminus \{p\} \mid sim(p, q) \geq \varepsilon\}.$$

Instead of looking for an ε-neighborhood (an ε-similarity neighborhood), one may be interested in determining *k-nearest neighbors* (*k-similarity nearest neighbors*). Let D' be a set containing k vectors from $D \setminus \{p\}$ and $\varepsilon = \max\{dis(p, q) \mid q \in D'\}$. Then, k-nearest neighbors are guaranteed to be found within ε distance from vector p; that is, are contained in $\varepsilon\text{-}NB_{dis}^{D}(p)$. In practice, the value of ε within which k-nearest neighbors of p are guaranteed to be found is re-estimated (is possibly narrowed) when

calculating the distance between p and next vectors from $D \setminus \{p\}$ [5-6]. In an analogous way, k-similarity nearest neighbors could be determined. In the following, we will focus on determining ε-(similarity) neighborhoods.

3 Triangle Inequality as a Means for Efficient Determining of Neighborhoods Based on Distance Metrics

In this section, we recall the method of determining ε-neighborhoods based on distance metrics efficiently, as proposed in [3-4].

Lemma 1 [3-4]. Let *dis* be a distance metric and D be a set of vectors. For any vectors u, v$\in D$ and any vector r:

$$dis(u, r) - dis(v, r) > \varepsilon \Rightarrow dis(u, v) > \varepsilon \Rightarrow v \notin \varepsilon\text{-}NB_{dis}^{D}(u) \ \wedge u \notin \varepsilon\text{-}NB_{dis}^{D}(v).$$

Now, let us consider vector q such that $dis(q, r) > dis(u, r)$. If $dis(u, r) - dis(v, r) > \varepsilon$, then also $dis(q, r) - dis(v, r) > \varepsilon$, and thus, $v \notin \varepsilon\text{-}NB_{dis}^{D}(q)$ and $q \notin \varepsilon\text{-}NB_{dis}^{D}(v)$ without calculating the real distance between q and v. This observation provides the intuition behind Theorem 1.

Theorem 1 [3-4]. Let r be any vector and D be a set of vectors ordered in a non-decreasing way with regard to their distances to r. Let $u \in D$, f be a vector following vector u in D such that $dis(f, r) - dis(u, r) > \varepsilon$, and p be a vector preceding vector u in D such that $dis(u, r) - dis(p, r) > \varepsilon$. Then:

a) f and all vectors following f in D do not belong to $\varepsilon\text{-}NB_{dis}^{D}(u)$;
b) p and all vectors preceding p in D do not belong to $\varepsilon\text{-}NB_{dis}^{D}(u)$.

As follows from Theorem 1, it makes sense to order all vectors in a given dataset D with regard to a reference vector as this enables simple elimination of a potentially large subset of vectors that certainly do not belong to an ε-neighborhood of an analyzed vector. The experiments reported in [3-4] showed that the determination of ε-neighborhoods by means of Theorem 1 was always faster than the determination using the R-Tree index, and in almost all cases speeded up the clustering process by at least an order of magnitude, also for high dimensional large vector sets consisting of hundreds of dimensions and tens of thousands of vectors. Analogous results were reported when determining k-neighborhoods [5-6].

4 The Cosine Similarity versus the Tanimoto Distance

In this section, we investigate the relation between the cosine similarity and the Tanimoto similarity and dissimilarity measures.

Lemma 2 [1]. Let u and v be non-zero vectors. Then:

$$T(u, v) = \frac{cosSim(u, v)}{A - cosSim(u, v)}, \text{ where } A = \frac{|u|}{|v|} + \frac{|v|}{|u|}.$$

Proof. $T(u, v) = \frac{u \cdot v}{u \cdot u + v \cdot v - u \cdot v} = \frac{cosSim(u,v) \mid u \mid \mid v \mid}{\mid u \mid^2 + \mid v \mid^2 - cosSim(u,v) \mid u \mid \mid v \mid} =$

$\frac{cosSim(u,v)}{\frac{\mid u \mid^2 + \mid v \mid^2}{\mid u \mid \mid v \mid} - cosSim(u,v)} = \frac{cosSim(u,v)}{\frac{\mid u \mid}{\mid v \mid} + \frac{\mid v \mid}{\mid u \mid} - cosSim(u,v)} = \frac{cosSim(u,v)}{A - cosSim(u,v)}$. □

Proposition 1. Let u and v be non-zero vectors. Then:

$$cosSim(u, v) = \frac{T(u,v)A}{1+T(u,v)}, \text{ where } A = \frac{\mid u \mid}{\mid v \mid} + \frac{\mid v \mid}{\mid u \mid}.$$

Proof. Follows from Lemma 2. □

Lemma 3. Let a and b be real numbers such that $ab > 0$. Then $\frac{a}{b} + \frac{b}{a} \geq 2$.

Proof. Follows from $(a-b)^2 \geq 0$. □

Theorem 2. Let u and v be non-zero vectors, $A = \frac{\mid u \mid}{\mid v \mid} + \frac{\mid v \mid}{\mid u \mid}$, $\varepsilon \in [-1,1]$ and $\varepsilon' = \frac{\varepsilon}{A - \varepsilon}$. Then $\varepsilon' \leq \varepsilon$ for $\varepsilon \in (0,1]$, $\varepsilon' \geq \varepsilon$ for $\varepsilon \in [-1,0)$, $\varepsilon' = \varepsilon$ for $\varepsilon = 0$ and

$$cosSim(u, v) \geq \varepsilon \Leftrightarrow T(u, v) \geq \varepsilon' \Leftrightarrow 1 - T(u, v) \leq 1 - \varepsilon'.$$

Proof. By Lemma 3, $A \geq 2$. Hence, $\varepsilon' \leq \varepsilon$ for $\varepsilon \in (0,1]$, $\varepsilon' \geq \varepsilon$ for $\varepsilon \in [-1,0)$ and $\varepsilon' = \varepsilon$ for $\varepsilon = 0$. By Proposition 1, $cosSim(u, v) \geq \varepsilon \Leftrightarrow \frac{T(u,v)A}{1+T(u,v)} \geq \varepsilon$ and by Property 2, $1 + T(u, v) > 0$. Hence, $cosSim(u, v) \geq \varepsilon \Leftrightarrow T(u, v) \geq \frac{\varepsilon}{A - \varepsilon} = \varepsilon'$. □

The corollary beneath follows immediately from Theorem 2.

Corollary 1. Let $D \cup \{u\}$ be a set of non-zero vectors, $\varepsilon \in [-1,1]$ and $\varepsilon'(v, w) = \frac{\varepsilon}{\frac{\mid v \mid}{\mid w \mid} + \frac{\mid w \mid}{\mid v \mid} - \varepsilon}$ for any vectors v, w in $D \cup \{u\}$. Then:

$$\varepsilon\text{-}SNB_{cosSim}{}^{D}(u) = \{v \in D \setminus \{u\} \mid T(u,v) \geq \varepsilon'(u,v)\} =$$

$$\{v \in D \setminus \{u\} \mid 1 - T(u,v) \leq 1 - \varepsilon'(u,v)\}.$$

Corollary 2. Let $D \cup \{u\}$ be a set of binary non-negative vectors and $\varepsilon \in [-1,1]$. Then, $\varepsilon\text{-}SNB_{cosSim}{}^{D}(u)$ can be determined by means of the Tanimoto dissimilarity and supported by the usage of the triangle inequality as specified in Lemma 1.

5 Bounds on Lengths of Binary Non-negative Cosine Similar Vectors

In this section, we consider the possibility of reducing the search space of candidate neighbor vectors by taking into account the lengths of vectors in a given dataset D.

Theorem 3. Let u and v be binary non-negative non-zero vectors such that $cosSim(u, v) \geq \varepsilon$ and $\varepsilon \in (0, 1]$. Then:

a) $|v| \in \left[\varepsilon |u|, \frac{|u|}{\varepsilon}\right]$;

b) $|v|^2 \in \left[\varepsilon^2 |u|^2, \frac{|u|^2}{\varepsilon^2}\right]$.

Proof. Ad a) Since u and v are binary non-negative vectors, then for any dimension i: $u_i v_i$ equals either $u_i u_i$ or $u_i 0$. Hence, $u_i v_i \leq u_i u_i$. Analogously, $u_i v_i \leq v_i v_i$.

Now, $\frac{|u|^2}{|u||v|} = \frac{\sum_{i=1..n} u_i u_i}{|u||v|} \geq \frac{\sum_{i=1..n} u_i v_i}{|u||v|} = \frac{u \cdot v}{|u||v|} = cosSim(u, v) \geq \varepsilon.$

Hence, $\frac{|u|^2}{|u||v|} \geq \varepsilon$. Thus, $|v| \leq \frac{|u|}{\varepsilon}$.

In addition, $\frac{|v|^2}{|u||v|} = \frac{\sum_{i=1..n} v_i v_i}{|u||v|} \geq \frac{\sum_{i=1..n} u_i v_i}{|u||v|} = \frac{u \cdot v}{|u||v|} = cosSim(u, v) \geq \varepsilon.$

Hence, $\frac{|v|^2}{|u||v|} \geq \varepsilon$. Thus, $|v| \geq \varepsilon |u|$.

Ad b) Follows immediately from Theorem 3a. □

Let u be a vector for which we wish to find its ε-cosine similarity neighborhood, where $\varepsilon \in (0, 1]$. Theorem 3 tells us that among vectors shorter than u only those not shorter than $\varepsilon |u|$ may belong to the ε-cosine similarity neighborhood of u, while among vectors longer than u only those not longer than $\frac{|u|}{\varepsilon}$ may belong to this neighborhood.

Corollary 3. Let $D \cup \{u\}$ be a set of binary non-negative non-zero vectors, $\varepsilon \in (0,1]$. Then:

$$\varepsilon\text{-}SNB_{cosSim}{}^{D}(u) = \left\{ v \in D \setminus \{u\} \,\middle|\, |v| \in \left[\varepsilon |u|, \frac{|u|}{\varepsilon}\right] \wedge cosSim(u, v) \geq \varepsilon \right\}.$$

6 The Tanimoto Distance and Lengths of Vectors as a Means to Restrict a Set of Candidates for Members of Cosine Similarity Neighborhoods of Binary Non-negative Vectors

In this section, we investigate the usage of both the Tanimoto dissimilarity, the triangle inequality and the found bounds on the length of vectors for determining cosine similarity neighborhoods of binary non-negative vectors.

Lemma 4. Let $a,c,d > 0$, $c \le d$, $b \in [c, d]$. Let $A = \max\left\{\frac{a}{c}+\frac{c}{a}, \frac{a}{d}+\frac{d}{a}\right\}$ Then:

a) $\frac{a}{b}+\frac{b}{a} \le A$;

b) If $\varepsilon \in [0,1]$, then $1-\frac{\varepsilon}{\frac{a}{b}+\frac{b}{a}-\varepsilon} \le 1-\frac{\varepsilon}{A-\varepsilon}$.

Proof. Ad a) Let $f(b)$: $[c, d] \rightarrow \mathrm{R}^+$ be a function such that $f(b) = \frac{a}{b}+\frac{b}{a}$. Then, $f'(b) = \frac{-a}{b^2}+\frac{1}{a}$.

Case $a \in [c, d]$. $f'(b) = 0$ iff $b = a$; $f'(b) < 0$ iff $b \in [c, a)$; $f'(b) > 0$ iff $b \in (a, d]$. Hence, $f(b)$ has minimal value for $b = a$; $f(b)$ is decreasing in $[c, a)$ and $f(b)$ is increasing in $(a, d]$. Thus, the greatest value of $f(b) = \max\left\{\frac{a}{c}+\frac{c}{a}, \frac{a}{d}+\frac{d}{a}\right\}$.

Case $a < c$. Then, $f'(b) > 0$ iff $b \in [c, d]$. Thus, $f(b)$ is increasing in $[c, d]$. So, the greatest value of $f(b) = f(d) = \frac{a}{d}+\frac{d}{a}$.

Case $a > d$. Then, $f'(b) < 0$ iff $b \in [c, d]$. Thus, $f(b)$ is decreasing in $[c, d]$. So, the greatest value of $f(b) = f(c) = \frac{a}{c}+\frac{c}{a}$.

So, we have proved that in all three cases $\frac{a}{b}+\frac{b}{a} \le \max\left\{\frac{a}{c}+\frac{c}{a}, \frac{a}{d}+\frac{d}{a}\right\} = A$.

Ad b) Follows immediately from Lemma 4a and Lemma 3. □

Lemma 5. Let u and v be non-zero vectors, $\varepsilon \in (0,1]$ and $|v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right]$. Then

$$1-\frac{\varepsilon}{\frac{|u|}{|v|}+\frac{|v|}{|u|}-\varepsilon} \le 1-\varepsilon^2.$$

Proof. Let $a = |u|$, $b = |v|$, $c = \varepsilon|u|$, $d = \frac{|u|}{\varepsilon}$ and $|v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right]$. Then, $b \in [c, d]$. Let $A = \max\left\{\frac{a}{c}+\frac{c}{a}, \frac{a}{d}+\frac{d}{a}\right\}$. Since, $\frac{a}{c}+\frac{c}{a} = \frac{1}{\varepsilon}+\varepsilon$ and $\frac{a}{d}+\frac{d}{a} = \frac{1}{\varepsilon}+\varepsilon$, then $A = \frac{1}{\varepsilon}+\varepsilon$. Hence, and by Lemma 4b, $1-\frac{\varepsilon}{\frac{|u|}{|v|}+\frac{|v|}{|u|}-\varepsilon} = 1-\frac{\varepsilon}{\frac{a}{b}+\frac{b}{a}-\varepsilon} \le 1-\frac{\varepsilon}{A-\varepsilon} = 1-\varepsilon^2$. □

Theorem 4. Let $D \cup \{u\}$ be a set of binary non-negative non-zero vectors, $\varepsilon \in (0,1]$ and $\varepsilon'(v, w) = \frac{\varepsilon}{\frac{|v|}{|w|}+\frac{|w|}{|v|}-\varepsilon}$ for any vectors v, w in $D \cup \{u\}$. Then:

$$\varepsilon\text{-}SNB_{cosSim}{}^{D}(u) = \left\{v \in D \setminus \{u\} \,\middle|\, |v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right] \wedge 1-T(u,v) \le 1-\varepsilon'(u,v)\right\} \subseteq$$

$$\left\{v \in D \setminus \{u\} \,\middle|\, |v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right] \wedge 1-T(u,v) \le 1-\varepsilon^2\right\} \subseteq (1-\varepsilon^2)\text{-}NB_{1\text{-}T}{}^{D}(u).$$

Proof. $\varepsilon\text{-}SNB_{cosSim}{}^{D}(u)$ = (by Corollary 3 and Theorem 2)

$$\left\{v \in D \setminus \{u\} \,\middle|\, |v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right] \wedge 1-T(u,v) \le 1-\varepsilon'(u,v)\right\} \subseteq \text{(by Lemma 5)}$$

$$\left\{v \in D \setminus \{u\} \,\middle|\, |v| \in \left[\varepsilon|u|, \frac{|u|}{\varepsilon}\right] \wedge 1-T(u,v) \le 1-\varepsilon^2\right\} \subseteq$$

$$\left\{v \in D \setminus \{u\} \,\middle|\, 1-T(u,v) \le 1-\varepsilon^2\right\} = (1-\varepsilon^2)\text{-}NB_{1\text{-}T}{}^{D}(u).$$ □

Corollary 4. Let $D \cup \{u\}$ be a set of binary non-negative non-zero vectors and $\varepsilon \in (0,1]$. Then, $\varepsilon\text{-}SNB_{cosSim}{}^{D}(u)$ can be determined by means of the Tanimoto distance and supported by the usage of the triangle inequality as specified in Lemma 1 and Theorem 1.

Theorem 4 suggests a number of ways in which $\varepsilon\text{-}SNB_{cosSim}{}^{D}(u)$, where $\varepsilon \in (0, 1]$, can be determined. First of all, $\varepsilon\text{-}SNB_{cosSim}{}^{D}(u)$ is a subset of $\varepsilon'\text{-}NB_{1-T}{}^{D}(u)$, $\varepsilon' = 1-\varepsilon^2$, which can be determined by means of the triangle inequality (e.g. as proposed in [3-4] and recalled in Section 3), provided dataset D contains only binary non-negative non-zero vectors. In addition, only those vectors v in $\varepsilon'\text{-}NB_{1-T}{}^{D}(u)$ the length of which

belongs to the interval $\left[\varepsilon \mid u \mid, \frac{\mid u \mid}{\varepsilon}\right]$ and that fulfill the condition $1-T(u,v) \leq 1-\varepsilon'(u,v)$, where $\varepsilon'(u, v) = \frac{\varepsilon}{\frac{\mid u \mid}{\mid v \mid}+\frac{\mid v \mid}{\mid u \mid}-\varepsilon}$, have a chance to belong to ε-$SNB_{cosSim}{}^{D}(u)$. In the sequel, the former condition is denoted by LC, the latter by TC and their conjunction by LTC. In Table 1, we present the results of evaluating the selectiveness of these conditions which we carried out on the dataset with 11 binary attributes and 2047 different vectors.

Table 1. Average numbers of evaluated vectors for a vector u, where LC(u) – the percentage of vectors in the dataset fulfilling the condition $\mid v \mid \in \left[\varepsilon \mid u \mid, \frac{\mid u \mid}{\varepsilon}\right]$, r = [10101010101] – used reference vector, TC(u) - the percentage of vectors v that were not eliminated by applying the triangle inequality condition $(1-T(u, r) - (1-T(u, v)) \leq 1-\frac{\varepsilon}{\frac{\mid u \mid}{\mid v \mid}+\frac{\mid v \mid}{\mid u \mid}-\varepsilon}$), LTC($u$) – the percentage of vectors fulfilling both TL(u) and TC(u)

ε	$\frac{\mid \varepsilon \text{ - } SNB_{cosSim}{}^{D}(u) \mid}{\mid D \mid}$	LC(u)	TC(u)	LTC(u)
0.9855	0,05%	16,44%	16,35%	4,85%
0.9200	0,25%	27,24%	53,21%	12,87%

As follows from these results, average selectiveness of LC is similar to that of TC for very high value of ε (0.9855) and greater than TC for lower value of ε (0.9200). In both cases, it is most beneficial to apply both conditions (LTC).

7 Conclusions

In the paper, we have proposed a new solution to determining neighborhoods defined in terms of the cosine similarity measure for binary non-negative vectors. We have proposed and proved that this problem can be transformed to the problem of determining neighborhoods defined in terms of the Tanimoto dissimilarity. This equivalence allows us to apply solutions based on using the triangle inequality that were proposed recently in the literature. In addition, we showed that in the case of binary non-negative vectors, one may restrict the cosine similarity neighbor search area by applying our proposed bounds on the lengths of candidate vectors.

Acknowledgements. This work was supported by the National Centre for Research and Development (NCBiR) under Grant No. SP/I/1/77065/10 devoted to the Strategic scientific research and experimental development program: "Interdisciplinary System for Interactive Scientific and Scientific-Technical Information". I would like to thank an anonymous reviewer for suggestions on how to simplify statements and proofs.

References

1. Leo, E.: New relations between similarity measures for vectors based on vector norms. ASIS&T Journal 60(2), 232–239 (2009)
2. Elkan, C.: Using the triangle inequality to accelerate k-means. In: Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003), Washington, DC, USA, August 21-24, pp. 147–153. AAAI Press (2003)
3. Kryszkiewicz, M., Lasek, P.: TI-DBSCAN: Clustering with DBSCAN by means of the triangle inequality. ICS Research Report 3, Institute of Computer Science. Warsaw University of Technology, Warsaw (2010)
4. Kryszkiewicz, M., Lasek, P.: TI-DBSCAN: Clustering with DBSCAN by Means of the Triangle Inequality. In: Szczuka, M., Kryszkiewicz, M., Ramanna, S., Jensen, R., Hu, Q. (eds.) RSCTC 2010. LNCS, vol. 6086, pp. 60–69. Springer, Heidelberg (2010)
5. Kryszkiewicz, M., Lasek, P.: A Neighborhood-Based Clustering by Means of the Triangle Inequality. In: Fyfe, C., Tino, P., Charles, D., Garcia-Osorio, C., Yin, H. (eds.) IDEAL 2010. LNCS, vol. 6283, pp. 284–291. Springer, Heidelberg (2010)
6. Kryszkiewicz, M., Lasek, P.: A neighborhood-based clustering by means of the triangle inequality and reference points. ICS Research Report 3, Institute of Computer Science. Warsaw University of Technology, Warsaw (2011)
7. Lipkus, A.H.: A proof of the triangle inequality for the Tanimoto dissimilarity. Journal of Mathematical Chemistry 26(1-3), 263–265 (1999)
8. Moore, A.W.: The anchors hierarchy: Using the triangle inequality to survive high dimensional data. In: Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence (UAI 2000), Stanford, California, USA, June 30-July 3, pp. 397–405. Morgan Kaufmann, San Francisco (2000)
9. Patra, B.K., Hubballi, N., Biswas, S., Nandi, S.: Distance Based Fast Hierarchical Clustering Method for Large Datasets. In: Szczuka, M., Kryszkiewicz, M., Ramanna, S., Jensen, R., Hu, Q. (eds.) RSCTC 2010. LNCS, vol. 6086, pp. 50–59. Springer, Heidelberg (2010)
10. Willett, P., Barnard, J.M., Downs, G.M.: Chemical similarity searching. J. Chem. Inf. Comput. Sci. 38(6), 983–996 (1998)

Automatic Chord Recognition Based on Probabilistic Integration of Acoustic Features, Bass Sounds, and Chord Transition

Katsutoshi Itoyama, Tetsuya Ogata, and Hiroshi G. Okuno

Kyoto University

Abstract. We have developed a method that identifies musical chords in polyphonic musical signals. As musical chords mainly represent the harmony of music and are related to other musical elements such as melody and rhythm, we should be able to recognize chords more effectively if this interrelationship is taken into consideration. We use bass pitches as clues for improving chord recognition. The proposed chord recognition system is constructed based on Viterbi-algorithm-based maximum *a posteriori* estimation that uses a posterior probability based on chord features, chord transition patterns, and bass pitch distributions. Experimental results with 150 Beatles songs that has keys and no modulation showed that the recognition rate was 73.7% on average.

1 Introduction

Automatic analysis of musical audio signals has recently become more important. Digital audio players with large memories and music distribution services have become commom place, and music information retrieval (MIR) and recommendation systems are required for the various needs of listeners. For more efficient and effective MIR, content-based information such as musical structure as well as the mood and genre of musical pieces is required in addition to text-based information. Manual annotation requires immense effort, and maintaining a consistent level of quality is not easy. Thus, techniques for extracting musical elements are essential for obtaining content-based information from musical signals.

The ultimate goal of our study is to develop an automatic music analysis system that can recognize various musical elements by considering their relationship to each other. When composing music, composers closely examine the relationships between musical elements such as melody, harmony, rhythm, music structure, timbre, tempo, and so on, so when analyzing music, it makes sense to examine the relationship of musical elements to improve recognition performance.

In this paper, we describe an automatic chord recognition method that uses the relationship between chord and bass pitch for use in the music analysis system mentioned above. To achieve the simultaneous recognition of multiple mucial elements, the use of Bayesian inference which estimates causes on the

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 58–67, 2012.

basis of the likelihood of evidence is promising. The chord sequence is regarded as one element of the harmony and bass pitches are regarded as one element of the melody. The chord sequence consists of a chord symbol sequence and a chord boundary sequence. The chord sequence often represents the mood of music, it can be used to calculate mood similarity between musical pieces which is important in MIR and music recommendation systems. The bass pitch represents a melody in the bass register and thus leads the chord progression.

An approach that has recently been used by many researchers for automated description of chord sequence is the use of hidden Markov models (HMMs). Several methods have been suggested to explore the analogy between speech recognition and chord recognition and to consider the temporal connection of chords [1,2,3]. Sheh *et al.* [1] proposed a method that uses an extension of the pitch class profile (PCP) [4] as a feature vector. They used an HMM that had one state per chord with a large set of classes (147 chord types). However, they were not able to obtain good recognition results. Bello *et al.* [2] also proposed the use of the chroma features and an HMM, and were able to improve recognition accuracy by incorporating musical knowledge into the model. Lee *et al.* [3] built key-specific models for automatic chord transcription. They used a 6-dimensional feature vector called tonal centroid, which is based on Tonnetz [5]. Higher accuracies were obtained by confining recognized chord types to a smaller set.

Yoshioka *et al.* have pointed out that chord symbols affect chord boundary recognition and vice versa [6]. They developed a method that concurrently recognizes chord symbols and boundaries by using a hypothesis-search method to recognize chord sequence and key. Sumi *et al.* [7] reported an improved version of this method that uses Gaussian mixture models (GMMs) to calculate the likelihood of acoustic feature vectors. Mauch *et al.* [8] reported a method using bass pitch classes but the probability of the bass pitch is given a priori.

While most previous studies have considered only the features of chords, we focus on the interrelationship among musical elements and integrate bass pitch information into chord recognition in a probabilistic training-based framework. The framework enables us to deal with multiple musical elements uniformly and facilitates the integration of information obtained from the statistical analysis of real music.

2 Chord Recognition Using Bass Pitch Estimation

We propose a chord recognition method based on the relationship between chord (from harmony) and bass pitch (from melody) and integrate them on the basis of the Bayesian framework.

2.1 Chords and Bass Pitch

We focus on chords and bass pitch specifically because they are closely related to chord sequence. Bass sounds are the most predominant tones in the low frequency region and have the following important properties:

– They compose the bass register of musical chords.
– They lead the chord sequence.

Therefore, we consider bass pitch estimation a promising clue for chord recognition.

2.2 Issues

In the previous method [7], an acoustic feature vector is calculated for a pair consisting of a chord symbol and its segment, and the likelihood of each pair is then evaluated. This previous method lacks a generalization capability of acoustic feature deviation caused by instrumentation, articulation, and arpeggio, thus preventing the use of optimum path-finding methods (such as a Viterbi algorithm) because of the temporal dependency of the acoustic features. Additionally, in the previous method, two 2-gram models represented the transition of chords for each major and minor scale. In western musical theory, a cadence (common unit of chord transition) is written as at least three chords, e.g., tonic→dominant→tonic. This suggests the need for a language model which represent more than two chord sequences.

2.3 Approaches

To address the above issues, we use bass pitch probability distribution and define a probabilistic observation model and a Bayesian chord transition model.

To improve the generalization capability and temporal dependency of the acoustic features, we utilize chord sequences as a Markov process of chord symbols for each temporal unit (eighth-note segment) with Gaussian mixture models (GMMs) for the distribution of the features. We use GMMs because they have a more efficient generalizatoin capability than the single Gaussian model. These models can be trained using actual musical pieces, which we assume will help improve the generalization capability.

We use a hierarchical Pitman-Yor language model (HPYLM) [9] for the transition of chord symbols. HPYLM is a Bayesian extention of the N-gram model with Kneser-Ney smoothing.

3 Chord Recognition System

In this section, we describe our chord recognition system. Inputs are polyphonic musical audio signals such as commercial CD recordings and outputs are estimated chord symbol sequences and a key. Our system calculates the posterior probability of the chord sequence using acoustic features, bass pitch probability distribution, and chord transition. The formalization of the chord recognition, definition of the posterior probability, and calculation method of the probability are described in the following subsections.

3.1 Formalization

In the proposed system, we define eighth-note segments as the smallest time unit of chord sequence estimation. These segments are estimated in advance by the beat tracking method [10]; let $[e_1, \ldots, e_K]$ be the estimated segments. For each segment, acoustic features $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K]$ are calculated and bass pitch probability distributions $B = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_K]$ are estimated.

In the proposed system, a sequence of chord symbols, $\boldsymbol{c} = [c_1, \ldots, c_K]$, for each eighth-note segment is to be recognized, and a key s for is also estimated collatelally. Each chord in the sequence is defined as a probabilistic variable:

$$c_k \in R \times \{\text{Major, Minor, Diminished, Sus4}\}, \tag{1}$$

where R is the set of twelve root tones:

$$R = \{\text{C, C\#, D, D\#, E, F, F\#, G, G\#, A, A\#, B}\}. \tag{2}$$

i.e., we use 48 classes. We use four types of triad chords: major (root, major third, perfect fifth), minor (root, minor third, perfect fifth), diminished (root, minor third, diminished fifth), and suspended4 (root, perfect fourth, perfect fifth) because these four chords are commonly used in popular music. Other chords such as augmented triad, sevenths, and so on are included in the above four chords on the basis of their component notes. For music information retrieval, we believe the broader classes should be sufficient to capture the characteristics and/or moods of musical pieces. The key s is defined as a probabilistic variable:

$$s \in R \times \{\text{Major, Minor}\} \tag{3}$$

and is estimated for the entire input musical piece. We also assume that musical pieces have the following properties:

1. Tempo stays constant.
2. Beat is a common measure (four-four time).
3. Key does not modulate.

Let $p(\boldsymbol{c}, s|X, B)$ be the posterior probability of the chord sequence and the key. We define the chord recognition as a maximum a posteriori (MAP) estimation problem, i.e., $\arg\max_{\boldsymbol{c},s} p(\boldsymbol{c}, s|X, B)$ is the chord sequence and key to be estimated.

3.2 Posterior Probability

The simultaneous probability of the chord sequence, key, acousitc features, and bass pitch probability distributions is defined as

$$p(\boldsymbol{c}, s, X, B) = p(s)\, p(c_1) \prod_{k=1}^{K} p(X_k|c_k)\, p(B_k|c_k) \prod_{k=2}^{K} p(c_k|c_1, \ldots, c_{k-1}, s). \tag{4}$$

A graphical representation of this probabilistic model is shown in Fig. 1. This model has the following properties:

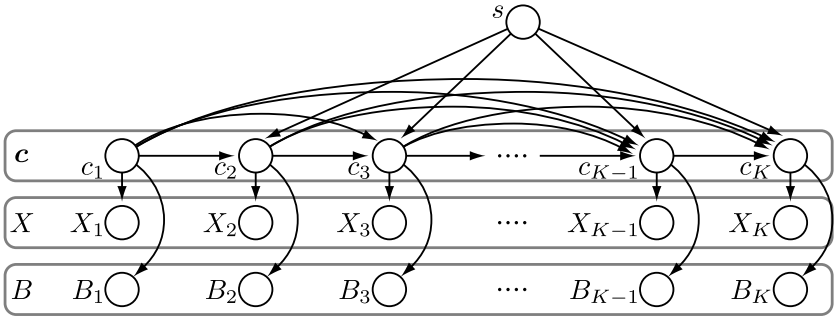


Fig. 1. Graphical model of chord sequence, key, acoustic features, and bass pitch probabilitiy distributions

1. Each chord c_k depends on previous chords $c_1, \ldots, c_{k-1}$ and on the key s.
2. Acoustic features X and bass pitch probability distributions B are conditionally independent over each corresponding chord.

Using Bayes' theorem, the posterior probability of the chord sequence $\boldsymbol{c}$ and the key s satisfies the following equation:

$$p(\boldsymbol{c}, s|X, B) = \frac{p(\boldsymbol{c}, s, X, B)}{p(X, B)}. \tag{5}$$

Since the denominator $p(X, B)$ is unrelated to the MAP estimation, we only have to deal with the numerator $p(\boldsymbol{c}, s, X, B)$. The prior probability of the key $p(s)$ is assumed as a uniform distribution. We obtain the following equation for chord recognition:

$$\begin{aligned} &\arg\max_{\boldsymbol{c},s} p(\boldsymbol{c}, s|X, B) \\ &\quad \equiv \arg\max_{\boldsymbol{c},s} p(c_1) \prod_{k=1}^{K} p(X_k|c_k)\, p(B_k|c_k) \prod_{k=2}^{K} p(c_k|c_1, \ldots, c_{k-1}, s). \end{aligned} \tag{6}$$

The chord sequence and the key wthat maximizes the posterior probability are estimated by using a Viterbi algorithm.

Acoustic Features. We use 12-dimensional chroma vectors [11] as acoustic features that approximately represent the intensities of the 12-semitone pitch classes. Chord symbols are identified by the variety of tones, so this representation is essential for chord recognition. Chroma vectors are calculated from the spectrogram of 55–1000 Hz. Note that chords of the same type that have different root notes, such as C Major and D Major, can be normalized by *cyclic shift*, as shown in Fig. 2. Thus we train only four (the number of chord types) chroma vector models and expand them to 48 (the number of chord symbols) by using cyclically shifted chroma vectors. This normalization increases the amount

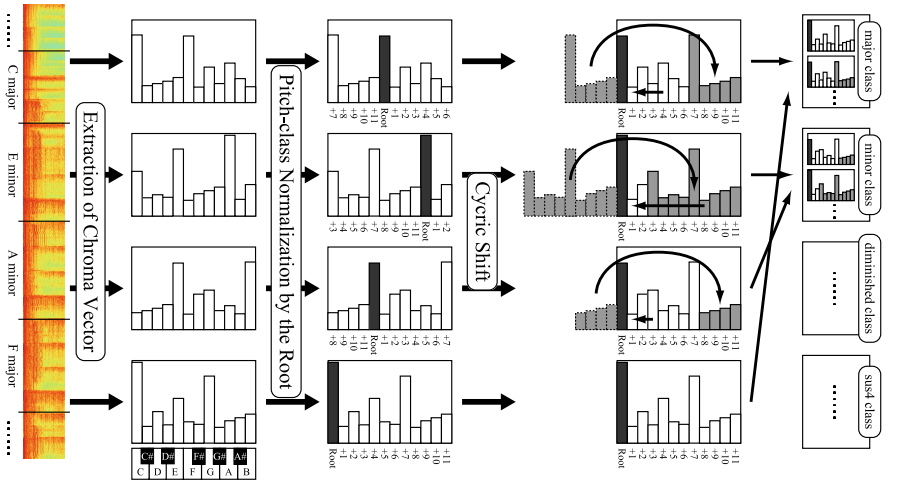


Fig. 2. Cyclic shift of chroma vector. By normalizing the indices of vectors by the root, then shifting and rotating the elements, the vectors for each code kind are obtained.

of training data for each chroma vector model, thus increasing the robustness of the chord recognition.

A Gaussian mixture model (GMM) is used for the probabilistic model of the chroma vectors. We assume that the chroma vectors are statistically independent for each time. The GMM parameters are obtained by using an expectation-maximization (EM) algorithm based on maximum likelihood estimation. Let M be the number of Gaussian mixtures and $\{\alpha_{c,1}, \ldots, \alpha_{c,M}\}$, $\{\boldsymbol{\mu}_{c,1}, \ldots, \boldsymbol{\mu}_{c,M}\}$, and $\{\Sigma_{c,1}, \ldots, \Sigma_{c,M}\}$ be the mixing coefficients, mean vectors, and covariance matrices, respectively. The likelihood of each chord symbol c_k is calculated for each chroma vector $\boldsymbol{x}_k$ as

$$p(\boldsymbol{x}_k|c_k) = \sum_{m=1}^{M} \frac{\alpha_{c_k,m}}{(2\pi)^6 \left|\Sigma_{c_k,m}\right|^{-\frac{1}{2}}} \exp\left(-\frac{(\boldsymbol{y}_k - \boldsymbol{\mu}_{c_k,m})^T \Sigma_{c_k,m}^{-1} (\boldsymbol{y}_k - \boldsymbol{\mu}_{c_k,m})}{2}\right), \tag{7}$$

where $\boldsymbol{y}_k$ is a 12-dimensional vector obtained by the cyclic shift of $\boldsymbol{x}_k$.

Bass Pitch Probability Distribution. The bass sounds lead the chord sequence, so they should be simultaneously analyzed as an element that is important for recognizing the chord sequence.

Bass pitch probability distributions B are estimated as existence degrees of instrument sounds performing bass parts for each pitch by using PreFEst [12]. We assume the bass sounds are performed in the frequency range of f_0 to f_1. Let M_0 and M_1 be the frequency index of f_0 and f_1, respectively. A bass pitch probability distribution $\boldsymbol{b}_k$ is defined as a vector on a $(M_1 - M_0 + 1)$-dimensional simplex:

$$k = 1, \ldots, K : \left[\begin{array}{c} \boldsymbol{b}_k = (b_{k,M_0}, \ldots, b_{k,M_1}) ; \sum_{m=M_0}^{M_1} b_{k,m} = 1; \\ m = M_0, \ldots, M_1 : 0 \leq b_{k,m} \leq 1 \end{array} \right]. \tag{8}$$

Let $\boldsymbol{\beta}_c = (\beta_{c,M_0}, \ldots, \beta_{c,M_1})$ be an averaged bass pitch probability distribution for chord c in the training data. If the bass sound which is usually one of the components of the chord is performed clearly, the chord can easily be estimated from the sound. A clearly performed bass sound results in the sparse bass pitch probability. The likelihood of the chord should be large if the common bass sound is performed.

We define the likelihood function to satisfy the above property as

$$p(\boldsymbol{b}_k | c_k) = (M_1 - M_0 + 1)! \, \langle \boldsymbol{b}_k, \boldsymbol{\beta}_{c_k} \rangle \,, \tag{9}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product of the vectors. The mode of this probabilistic distribution is

$$\begin{aligned} &(b_{k,M_0}, \ldots, b_{k,m}, \ldots, b_{k,M_1}) = (0, \ldots, 1, \ldots, 0) \\ &\text{s.t. } m = \arg \max_{m \in \{M_0, \ldots, M_1\}} \{\beta_{c_k,M_0}, \ldots, \beta_{c_k,M_1}\} \,, \end{aligned} \tag{10}$$

i.e., the likelihood is maximized if the most common pitch of the bass sound m in chord c_k is obviously performed.

Chord Transition. As music theory has indicated, patterns of chord progression (such as I – IV – V) are limited for genres and artists, and artists and transition probability is usually biased. Abstract patterns are translated into concrete chord symbols by using the key. Ambiguity of chord symbols can be resolved by using a model of the chord progression. We model chord progression patterns with a hierarchical Pitman-Yor language model (HPYLM) [9] thus eliminating any ambiguity.

The Hierarchical Pitman-Yor language model is a hierarchical generative model of the N-gram language model, i.e., it approximates the probability $p(c_k | c_1, \ldots, c_{k-1}, s)$ by the (N−1)-th order Markov model $p(c_k | c_{k-n+1}, \ldots, c_{k-1}, s)$ and estimates N-gram probability distribution by using a Chinese restaurant process (CRP) and the Markov-chain Monte Carlo (MCMC) algorithm. The probability of c_k preceded by the context $h = c_{k-n+1}, \ldots, c_{k-1}$ is described as

$$p(c_k | h) = \frac{\mathrm{c}(c_k | h) - d \cdot t_{hc_k}}{\theta + \mathrm{c}(h)} + \frac{\theta + d \cdot t_{h\cdot}}{\theta + \mathrm{c}(h)} p(c_k | h') \,, \tag{11}$$

where $\mathrm{c}(c_k | h)$ means the count of c_k preceded by h, $\mathrm{c}(h) = \sum_{c_k} \mathrm{c}(c_k | h)$ means the sum of them, and $h' = c_{k-n+2}, \ldots, c_{k-1}$ means a context of lower order.

Table 1. Recognition rates [%] for each combination of musical elements

(1) acoustic features, (2) bass pitch, (3) (1) and (2), (4) (1) and chord transition, (5) (2) and chord transition, (6) our proposed method, (7) previous method [7]

Method	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Recognition rate	59.8	41.0	66.6	61.9	52.7	**73.7**	73.4

Table 2. Recognition rates [%] for each number of GMM components

Number of components	1	2	4	8	16
Recognition rate	61.0	67.9	**73.7**	72.6	66.9

4 Experimental Evaluation

We conducted experiments to evaluate the proposed system. 150 songs without modulation were excerpted from 12 Beatles albums for test data. These songs were divided into five random groups for 5-fold cross validation. We set the number of Gaussian mixtures to $M = 4$ and the frequency range of bass sounds $[f_0, f_1]$ to $[29, 261]$ Hz. The 120 songs of the other four groups were used for training chroma vectors, bass sounds, and chord transition models. For correct chord labels, we used ground-truth annotations of the Beatles albums [13]. We evaluated the system by recognition rate defined as

$$\text{Recognition Rate} = \frac{\text{Total Duration of Correct Chord}}{\text{Duration of Song}} \times 100\ \%. \tag{12}$$

We also measured the recognized chord symbols by average and maximum of continuous correct segments with an error tolerance of 50 msec.

To evaluate the effectiveness of our method, we compared the frame-rate accuracies of five methods of computing posterior probability:

1. Using only acoustic features
2. Using only bass pitch probability distribution
3. Using acoustic features and bass pitch probability distribution
4. Using acoustic features and chord transition (3-gram)
5. Using bass pitch probability distribution and chord transition
6. Proposed method (using all three elements)
7. Previous method [7]

The results are shown in Table 1.

To evaluate the effectiveness of using GMMs, we compared the number of GMM components over 1, 2, 4, 8, and 16. Note that 1 GMM component corresponds to the use of single Gaussian distribution model. The results of this evaluation are shown in Table 2.

With our system, the average recognition rate for 150 songs was 73.7% (as shown in Table 1.) Compared with using only acoustic features, the methods using bass pitch and chord transition improved the rate by 6.8 and 2.1 points, respectively, and the proposed method improved the rate by 13.9 points. Our system's accuracy was higher than that of the baseline method because the probabilistic integration enabled us to utilize information about bass sounds as a clue in chord recognition. These results demonstrate both the importance of considering the interrelationship between chord sequence and bass sounds and the effectiveness of the probabilistic integration of these elements.

As shown in Table 2, in all cases involving 2, 4, 8, and 16 Gaussian components, the recognition rates were improved more than the case of a single Gaussian component. We therefore set the number Gaussian components to 4 in subsequent experiments.

We compared the average duration of continuous correct segments with the previous method and our proposed method with 2-gram and 3-gram models. Although the duration was 1.66 sec in the previous method, the durations were 2.88 and 2.97 secs in our method with the 2-gram and 3-gram models, respectively.

5 Conclusion

We presented a chord recognition system that utilizes the interrelationships between various musical elements. Specifically, we focused on acoustic features, bass pitch probability distribution and chord transition and integrated them in a probabilistic framework. The experimental results showed that our system can achieve a chord recognition rate of 73.7% for 150 Beatles songs. The results also showed an increase in accuracy when the three reliabilities are integrated compared with the baseline method or when using only acoustic features. This demonstrates that to recognize musical elements, which consist not only of musical chords but also of other elements, it is important to consider the interrelationship among musical elements and to integrate them probabilistically. Future works includes additional experimental evaluations using more data sets of various artists to eliminate the bias which is caused by using only the Beatles albums. To obtain more information about how to recognize chord sequences more effectively, we will design a method of integrating other musical elements such as rhythm.

References

1. Sheh, A., Ellis, D.P.: Chord segmentation and recognition using EM-trained hidden Markov models. In: ISMIR 2003, pp. 183–189 (2003)
2. Bello, J.P., Pickens, J.: A robust mid-level representation for harmonic content in music signals. In: ISMIR 2005, pp. 304–311 (2005)
3. Lee, K., Slaney, M.: A unified system for chord transcription and key extraction using hidden Markov models. In: ISMIR 2007, 245–250 (2007)
4. Fujishima, T.: Realtime chord recognition of musical sound: A system using common lisp music. In: ICMC 1999, pp. 464–467 (1999)

5. Harte, C., Sandler, M., Gasser, M.: Detecting harmonic change in musical audio. In: AMCMM 2006, pp. 21–26 (2006)
6. Yoshioka, T., Kitahara, T., Komatani, K., Ogata, T., Okuno, H.G.: Automatic chord transcription with concurrent recognition of chord symbols and boundaries. In: ISMIR 2004, pp. 100–105 (2004)
7. Sumi, K., Itoyama, K., Yoshii, K., Komatani, K., Ogata, T., Okuno, H.G.: Automatic chord recognition based on probabilistic integration of chord transition and bass pitch estimation. In: ISMIR 2008, pp. 39–44 (2008)
8. Mauch, M., Dixon, S.: Simultaneous estimation of chords and musical context from audio. IEEE Trans. Audio, Speech and Lang. Process. 18, 1280–1289 (2010)
9. Teh, Y.W.: A hierarchical bayesian language model based on Pitman-Yor processes. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006), ACL-44, pp. 985–992. Association for Computational Linguistics, Stroudsburg (2006)
10. Goto, M.: An audio-based real-time beat tracking system for music with or without drum-sounds. J. New Music Res. 30, 159–171 (2001)
11. Goto, M.: A chorus-section detecting method for musical audio signals. In: ICASSP 2003, pp. V–437–V–440 (2003)
12. Goto, M.: A real-time music-scene-analysis system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. Speech Communication 43, 311–329 (2004)
13. Harte, C., Sandler, M., Abdallah, S., Gómez, E.: Symbolic representation of musical chords: A proposed syntax for text annotations. In: ISMIR 2005, pp. 66–71 (2005)

A Real-Time Transportation Prediction System

Haiguang Li[1], Zhao Li[1], Robert T. White[2], and Xindong Wu[1]

[1] The University of Vermont – Burlington, VT 05405, USA
{hli,zhaoli,xwu}@cems.uvm.edu
[2] State of Vermont AOT – Montpelier, VT 05633, USA
Robert.T.White@state.vt.us

Abstract. In recent years, the use of advanced technologies such as wireless communication and sensors in intelligent transportation systems has made a significant increase in traffic data available. With this data, traffic prediction has the ability to improve traffic conditions and to reduce travel delays by facilitating better utilization of available capacity. This paper presents a real-time transportation prediction system named VTraffic for Vermont Agencies of Transportation by integrating traffic flow theory, advanced sensors, data gathering, data fusion, and data mining and visualization technologies to estimate and visualize the current and future traffic. In our VTraffic system, acoustic sensors were installed to monitor and to collect real-time data. Reliable predictions can be obtained from historical data and be verified and refined by the current and near future real-time data.

Keywords: sensors, intelligent transportation system, traffic prediction, real-time, data fusion, data mining, visualization.

1 Introduction

Traffic congestion is a situation of transportation systems that occurs as the saturation of road network capacity, due to increased traffic volume or interruptions, and is characterized by slower speeds, increased vehicular queuing, and longer trip times. Congestion is one external cost of transport and the reduction of its impact is often one of the primary objectives for transport policy makers. Traffic congestion, continuously one of the major problems in various transportation systems, has many negative effects on travelers, businesses, agencies and cities. One significant aspect is the value of the wasted fuel and additional time. The top 15 urban areas include about 58 percent of the delay estimated for 2010, and the top 20 areas account for over 65 percent of the annual delay. Based on wasted time and fuel, traffic congestion costs about $115 billion in the 439 urban areas in 2010 [1].

The negative impacts of traffic congestion may be alleviated by providing timely and reliable prediction information to system dispatchers and motorists

* This work is supported by Vermont Agencies of Transportation under grant No. 000025425.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 68–77, 2012.

[2]. However, traffic situations vary significantly depending on the weather situation, the season of the year, the day of the week, and even the time of day. In addition, the capacity, which is often mistakenly considered to be constant, may vary because of weather, work zones, or traffic incidents and so on [3]. Furthermore, those conditions are not independent of each other, most of them are interrelated explicitly or implicitly. Therefore, there are a great number of differing and changing circumstances which cause or aggravate congestion. It is a great challenge to estimate under which conditions a "congestion" may occur suddenly.

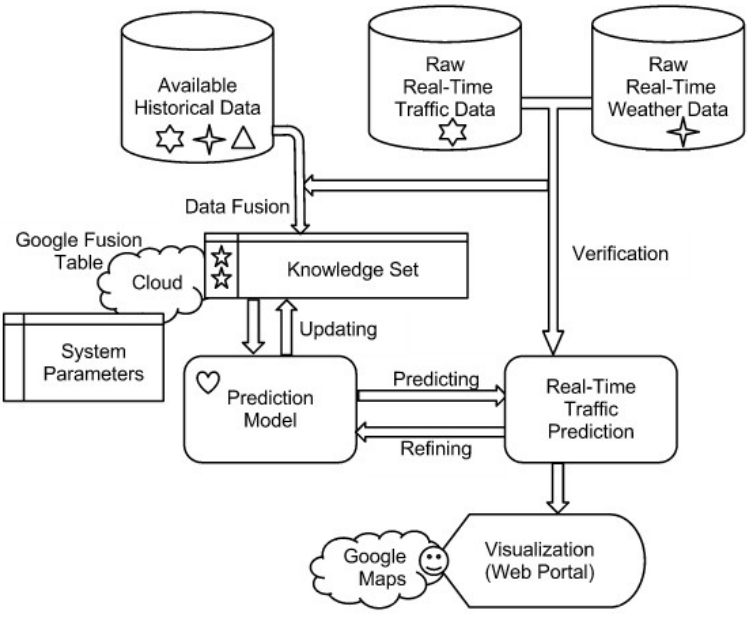


Fig. 1. The Structure of VTraffic System

The structure of our VTraffic system is given in Fig. 1. In our VTraffic system, acoustic sensors were installed on highways, I89 and I91, within the territory of Vermont to monitor and to collect real-time data. Firstly, we design a data fusion strategy to improve the quality of raw data gathered from different sensors and other available historical data to ensure the information can be used for traffic estimation. Secondly, we use data fusion and data mining within the data which comes from heterogeneous data sources to get useful knowledge and store in a Google fusion table in the cloud [4]. Thirdly, a prediction model is built from the knowledge set for real-time traffic predicting, and the real-time data is employed for verifying previous predictions and refining the model. Finally, a web portal is implemented using Google maps for visualization [5].

2 Infrastructure

Typically, interstate highways are the busiest and most important main roads. There are two interstate highways I89 and I91 across the state of Vermont. Therefore, we choose I89 and I91 to deploy our VTraffic system. Dozens of acoustic sensors have already been installed on the two highways*. In the near future, we are planning to add several more sensors into the system.

2.1 Sensor Function

Our system adopts the Vaisala nu-metrics portable traffic analyzer NC-200 shown in Fig. 2 to collect real-time traffic data. The sensor can be installed quickly and easily, which is designed to detect accurate vehicle count, speed, and classification by utilizing vehicle magnetic imaging (VMI) technology. After data has been collected from a sensor, the data can be easily exported to highway data management (HDM) software shown in Fig. 3, where it can be presented in the form of reports, charts and graphs [6].

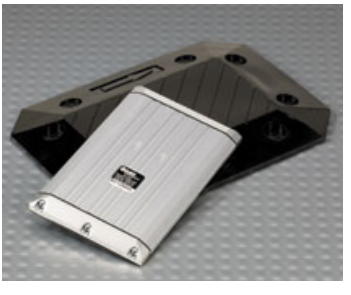


Fig. 2. NC-200

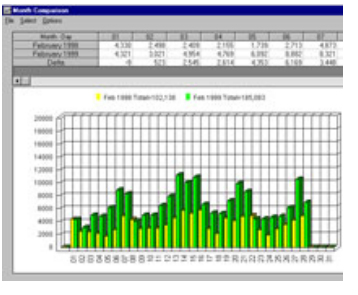


Fig. 3. HDM Software

As we all know, the impact of weather on traffic is very serious. Weather includes visibility, precipitation, wind, and temperature, which affects driver capabilities, vehicle performance, pavement friction, and roadway infrastructure to impact the state of the transportation system [7].

The clarus system was established in 2004 to provide weather information to transportation managers and users to alleviate the effects of adverse weather [8,9]. This system belongs to Federal Highway Administration Research & Innovative Technology Administration, and it is provided as a public service. Our VTraffic system gets historical and real-time weather data from the clarus system.

* There are 56 sensors deployed on I89 and I91, 28 for each. The location details can be found at `http://cs.uvm.edu/~hli/www/traffic/html/sensorlocations.htm`

2.2 Data Collection Strategy

The traffic and weather historical data recorded by transportation managers and sensors are obtained at the very beginning. Considering the data transmission efficiency and computational performance, NC-200 sensors continuously gather real-time traffic data, but only report data once every five minutes. In general, the weather does not occur great changes in a relatively short period of time. Therefore, the real-time weather data are fetched once every five minutes from the clarus system.

3 Data Fusion

Many transportation agencies have developed techniques for collecting the real-time data and storing it historically, while few have been making full use of it. In order to make full use of all available data, we have designed a strategy to integrate data from heterogeneous sources.

3.1 Data Preprocessing

The important attributes of traffic data collected by NC-200 sensors and those of weather data obtained from the clarus system are listed in Table 1. We also have some historical traffic data filled by transportation managers, however, this data only provides general traffic information about some particular sites, we can only use them for verification during preprocessing.

Table 1. Data Formats

Data Source	Data Format								
NC-200	Location	Date	Time	VehicleCount	Volume	Speed	Report time	Occup	...
Clarus	Location	Date&Time		Temperature	Humidity	WindSpeed	WindDirect	Visibility	...
Integration	Location	Date	Time	Confidence	Parents	TrafficAttrs	WeatherAttrs	Children	...

The historical data is preprocessed with the following steps: 1) Remove useless attributes; 2) Guess missing values; 3) Correct wrong data which can be corrected; 4) Remove wrong data which cannot be corrected; 5) Remove redundant data.

3.2 Data Fusion

The traffic data and the weather data are different in source, sensor type, location, time, and data format. Therefore, we have to fuse them first, and the format of the integration data is shown in Table 1.

Data fusion strategy consists of the following steps:

1. As mentioned in Section 2.2, the data reported by NC-200 sensors contain the real-time traffic of the last five minutes. We first split the data into twenty pieces, each interval of fifteen seconds, and the data format remains the same.
2. From Table 1 we can easily observe weather data has the attribute "Date & Time" while there are two attributes "Date" and "Time" in traffic data. We split "Date & Time" into "Date" and "Time".
3. By adding the weather data to each traffic data piece, we fuse those two data sets into one. Since the location and time between the weather data and the traffic data may not be exactly the same, we just choose the weather data with the closest time and location.
4. The integration data generated in Step 3, also called the knowledge set, is stored in a Google fusion table in the cloud [4].

4 Prediction Model

As we discussed in Section 2.2, our sensors collect real-time traffic data only once every five minutes. Therefore, we do not know the real-time traffic between any two consecutive data collection until the next data has arrived. However, five minutes on the highway is a relatively long time, we have to model the real-time traffic. The continuous real-time traffic can be modeled according to the real-time data collected by the sensors and all other available knowledge generated by the data fusion process in Section 3. The model will be used for real-time traffic prediction and travel-time guidance.

4.1 Modeling

In our VTraffic system, there are two interstate highways, I89 and I91, and obviously, each of them has two directions south and north, which means there are four highways: I89N, I89S, I91N, and I91S. Therefore, we have to model the real-time traffic of the four highways. To simplify, we assume that the real-time traffic of the four highways are independent of each other.

The prediction model for a highway is a 4-tuple (Σ, Υ, Ξ, Δ), where

1. Σ is a finite set of sensors, each sensor owning a state represents the real-time traffic of its location. As Fig. 4 shows, the sensors are sequentially arranged from 1 to n.
2. Υ is a finite set of roads, each road owning a state represents the real-time traffic of that road. The ith road is the sub-section between the ith sensor and the $(i+1)$th sensor, therefore $|\Upsilon| = |\Sigma| - 1$.
3. Ξ is a set of knowledge items from the historical data and the real-time data (mentioned in Section 3). Ξ keeps constantly updating because that the real-time data turns into the historical data over time.
4. $\Delta : \Sigma \times \Xi \times \Upsilon \rightarrow (\Sigma' \times \Xi' \times \Upsilon')$ is the predictor.

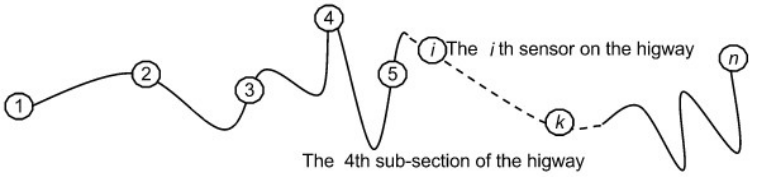


Fig. 4. The Topology of a Highway

Furthermore, we use four separate models to model the four highways respectively, then the four models together form the prediction model of our VTraffic system.

4.2 Real-Time Predicting

We collect the real-time traffic data once every five minutes, and predict the real-time traffic once every "fifteen seconds", where the prediction interval "fifteen seconds" is a system parameter.

Predicting. The main idea of our prediction model is: the real-time traffic at a particular site associates with time, weather, location, and the previous traffic situation, the traffic situation some distance before and so on. Based on this idea, we use the k-nearest neighbor (k-NN) algorithm [10] to get k knowledge items from Ξ. The prediction algorithm is shown in Algorithm 1, it use k-NN twice: Step 2 uses k-NN to find a set of k-nearest knowledge items N_0 at the prediction site S based on the current weather and time, while Step 4 finds N_1 at other sites based on the current weather, time, and previous traffic condition at this site S. Steps 3 and 5 predict the traffic based on N_0 and N_1 respectively. Here, the confidence of the predictions $\Re_0$ and $\Re_1$ comes from the confidence of items in N_0 and N_1 respectively. Step 6 integrates the predictions $\Re_0$ and $\Re_1$ to obtain

Algorithm 1. Predicting

Require:
Set $K = 10$; $\Theta = 0.6$; $S = Site$; $\Re = NULL$;
$\Xi' = \{\hbar | \hbar.location = S \& \hbar \subseteq \Xi\}$; $\overline{\Xi'} = \Xi - \Xi'$;

Ensure:
1: **if** $S \in \Sigma \& \Theta \leq 1 \& \Theta \geq 0$ **then**
2: $N_0 \leftarrow knn(K, \Xi', Weather(S), Time(S))$;
3: $\Re_0 \leftarrow predicting(N_0)$;
4: $N_1 \leftarrow knn(K, \overline{\Xi'}, Weather(S), Time(S), Traffic(S))$;
5: $\Re_1 \leftarrow predicting(N_1)$;
6: $\Re \leftarrow Sum(\Re_0 \times \Theta, \Re_1 \times (1 - \Theta))$
7: $AddRelationship(N_0 \bigcup N_1, \Re)$
8: $\Xi \leftarrow \Xi \cup \{\Re\}$;
9: **end if**
10: **return** $\Re$;

the final prediction $\Re$. Obviously, the items in N_0 and N_1 are the parents of $\Re$, on the other hand, $\Re$ is a new child of the items in N_0 and N_1. Therefore, Step 7 add their relationships in their "parents" and "children" attributes. Finally, Step 8 adds the prediction $\Re$ into the knowledge set Ξ, and this algorithm returns the prediction $\Re$ at Step 10.

The traffic of the ith road is the arithmetic mean of sensor conditions at its both ends shown in Formula (1).

$$\Upsilon[i] = avg(\Sigma[i], \Sigma[i+1]) \tag{1}$$

Verification. Our prediction model updates the traffic of the entire transportation system immediately when sensors collect and report the real-time traffic data. Meanwhile, the real-time data also verifies the previous predictions: if a prediction passed the verification, the knowledge items which generated the prediction will get a bonus; otherwise they will get a penalty. The bonus or penalty will affect the confidence of the corresponding knowledge items.

4.3 Travel-Time Estimation

When a travel-time request occurs, our model uses the real-time traffic state for estimation shown in Formula (2). Firstly, the model identifies the starting and ending sites from the request. Secondly, the two sites are mapped into several adjacent roads on a highway. Thirdly, the travel-time of each road is calculated according to the real-time traffic of that road. Finally, the overall travel-time is the sum of all travel-time on every road.

$$travel\_time(A, B) = \sum_{r \in Roads} \frac{r.distance}{r.state.speed} \tag{2}$$

5 Visualization

Our VTraffic system displays the traffic of the entire transportation system as geographical objects on a map, then changes the color, size, and displays of meaningful markers and curves based on the real-time traffic prediction, to allow users to quickly grasp the traffic of the entire system or some particular location.

5.1 Algorithms

As we mentioned in our prediction model in Section 4.1, a highway is divided into $n-1$ adjacent roads by n sensors. We can get the real-time traffic from the model, and then display on a map in a user-friendly form of visualization. There are two important algorithms for the visualization of our system: Initialization (shown in Algorithm 2) and updating (Algorithm 3).

The initialization algorithm is invoked when a user launches or refreshes the visualization web portal. Its main function is to create and initialize the essential

objects, and then display them on the map. For each sensor in Σ, the algorithm creates a marker to represent the sensor. In Step 2, according to the prediction of the *model*, it chooses a color C to represent the traffic state at that site. In Steps 3 and 4, a new $Marker$ is created, in order to make important information more eye-catching, $Visible = True$ only when the traffic at that site is congested. From Step 5 to 9, a *Road* is created and assigned a color C according to the traffic prediction of the *model* to represent its traffic state.

Algorithm 2. Initialization

Require:
Set $A = \Sigma$; $\xi = Congestion_{Threshold}$; $Markers = \{\emptyset\}$; $Roads = \{\emptyset\}$;
Ensure:
1: **for** $(I \leftarrow 0; I < A.size(); I \leftarrow I + 1)$ **do**
2: $C \leftarrow Model.predict(A[I]) \geq \xi \ ? \ COLOR_{Free} : COLOR_{Congested}$;
3: $Visible \leftarrow Model.predict(A[I]) \geq \xi \ ? \ False : True$;
4: $Markers \leftarrow Markers \cup newMarker(A[I], C, Visible)$;
5: **if** $I > 0$ **then**
6: $S \leftarrow Model.predict(A[I]) + Model.predict(A[I-1])$;
7: $C \leftarrow S/2 \geq \xi \ ? \ COLOR_{Free} : COLOR_{Congested}$
8: $Roads \leftarrow Roads \cup newRoad(A[I-1], A[I], C)$;
9: **end if**
10: **end for**

Algorithm 3. Updating

Require:
Set $\xi = Congestion_{Threshold}$; $M = Markers$; $R = Roads$;
Ensure:
1: **for** $(I \leftarrow 0; I < M.size(); I \leftarrow I + 1)$ **do**
2: $C \leftarrow Model.predict(M[I]) \geq \xi \ ? \ COLOR_{Free} : COLOR_{Congested}$;
3: **if** $C \neq M[I].getColor()$ **then**
4: $M[I].setColor(C)$;
5: $M[I].setVisible(!M[I].getVisible())$;
6: **end if**
7: **if** $I > 0$ **then**
8: $S \leftarrow Model.predict(M[I]) + Model.predict(M[I-1])$;
9: $C \leftarrow S/2 \geq \xi \ ? \ COLOR_{Free} : COLOR_{Congested}$;
10: $R[I-1].setColor(C)$;
11: **end if**
12: **end for**

The updating algorithm is automatically invoked to update the traffic at regular intervals. Therefore, it is very important and effective to optimize this algorithm. This algorithm is very similar to Algorithm 2, and there are only two differences: a) this algorithm does not create any new object, and it just reuses the $Marks$ and *Roads* created by Algorithm 2. b) this algorithm only updates

those markers and roads which need to be updated. With a), it does not consume any additional memory, with b) it tries to do as little as possible. By a) and b), the updating algorithm has been well optimized.

We have also designed a *Zoom_Change* function and a *Click_Event* monitor. The *Zoom_Change* deals with some details when the zoom size of the map is changing, and the *Click_Event* monitors the click events and shows more detailed information about a particular site where the marker is clicked.

5.2 Visualization

Fig. 5 is a run-time screenshot. The real-time traffic prediction is displayed over the highway as color-coded lines. The colors indicate the traffic on the road (green: free, red: congested), and the small pink circles (come from Step 3 of Algorithm 2 in Section 5.1) indicates the traffic at a particular sensor site is congested.

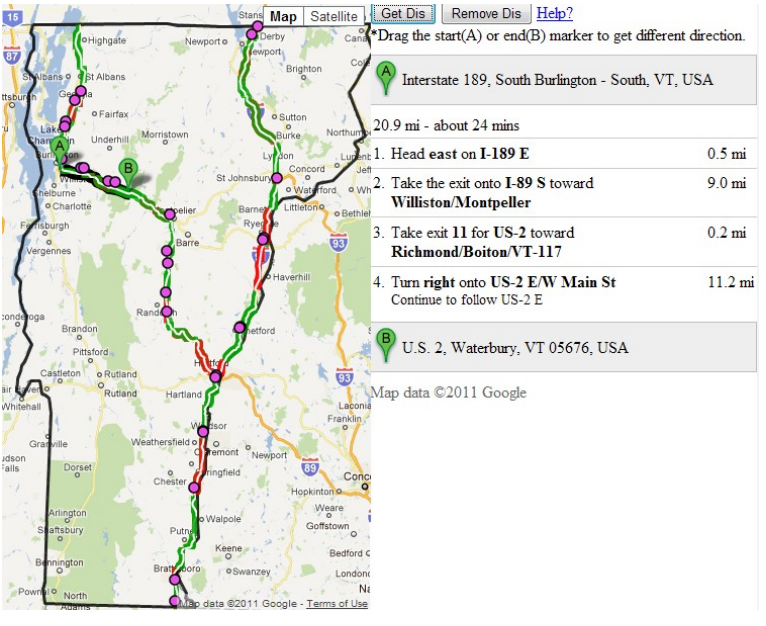


Fig. 5. Visualization*

By simply clicking on a pink circle, users can get an infowindow to show more information in detail. Users can zoom in to have a more specific vision with more detail or zoom out to get a broader perspective with less detail. The list on the right shows the travel information from A to B (also marked in the left map), which is estimated by our model in Section 4.3.

* A demo can be found at `http://cs.uvm.edu/~hli/www/traffic`. Please read the help file first at `http://cs.uvm.edu/~hli/www/traffic/html/help.htm`

6 Conclusions

Our VTraffic system is a dynamic real-time traffic prediction system that provides traffic predictions and travel guidance. In order to make full use of all available data, we built a data fusion strategy to integrate data from heterogeneous data sources. To maximize the quality of the prediction, a dynamic prediction model was implemented. The predictions will be automatically verified by real-time data, and the model will be refined dynamically. A web portal was also built for visualization, using it users can easily observer the traffic and get useful guidance.

References

1. Schrank, D., Lomax, T., Eisele, B.: 2011 urban mobility report. Technical report, Texas Transportation Institute (September 2011)
2. Chien, S.I.J., Liu, X., Ozbay, K.: Predicting travel times for the south jersey real-time motorist information system. Transportation Research Record: Journal of the Transportation Research Board, 1885/2003, 32–40 (January 2007)
3. FHWA. Fhwa operations - operations story (August 2011)
4. Google Fusion Table, `https://www.google.com/fusiontables/`
5. Google Maps, `http://code.google.com/apis/maps/`
6. Vaisala Homepage, `http://www.vaisala.com/`
7. Pisano, P.A., Goodwin, L.C., Rossetti, M.A.: U.S. highway crashes in adverse road weather conditions. In: 88th American Meteorological Society Annual Meeting, New Orleans, Louisiana (January 2008)
8. Pisano, P.: Clarus success stories: Using clarus data to improve operations. Technical Report FHWA-JPO-10-005, U.S. Department of Transportation Road Weather Management (2009)
9. Clarus System, `http://www.clarus-system.com/`
10. Athitsos, V., Alon, J., Sclaroff, S.: Efficient nearest neighbor classification using a cascade of approximate similarity measures. In: IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, pp. 486–493 (April 2005)

Several Remarks on Mining Frequent Trajectories in Graphs

Henry Z. Lo, Dan A. Simovici, and Wei Ding

Univ. of Massachusetts Boston, Dept. of Computer Science, Boston, Massachusetts
{henryzlo,dsim,ding}@cs.umb.edu

Abstract. We apply techniques that originate in the analysis of market basket data sets to the study of frequent trajectories in graphs. Trajectories are defined as simple paths through a directed graph, and we put forth some definitions and observations about the calculation of supports of paths in this context. A simple algorithm for calculating path supports is introduced and analyzed, but we explore an algorithm which takes advantage of traditional frequent item set mining techniques, as well as constraints placed on supports by the graph structure, for optimizing the calculation of relevant supports. To this end, the notion of the path tree is introduced, as well as an algorithm for producing such path trees.

1 Introduction

Determining frequent item sets in market basket data sets is an unsupervised data mining activity that has received a great deal of attention beginning with the seminal paper [2] and continuing with several fundamental references [6,9,5] A monograph dedicated to this task is [1].

Finding frequent item sets is a necessary step in computing association rules. An association rule stipulates that with a certain probability customers who buy an item set K will buy an item set H. Such rules provide actionable information for marketeers who will place items from $K \cup H$ in physical proximity in order to stimulate sales.

The purpose of this paper is to develop a study of frequent trajectories in graphs inspired by the ideas used in the analysis of market basket data sets. The study of trajectory data has been explored intensively in the literature [3,8,4] motivated by the large amount of spatio-temporal data allowed by location acquisition technologies. Our model is simpler than the model used in the previous references, in that, it does not include explicitly the temporal aspect. In exchange, our approach extends ideas that originate in market-basket analysis and allows us to build simple and efficient algorithms that will allow, at a later stage, the integration of the temporal aspect.

We present some preliminary results and formulate a number of open problems that we intend to approach in our future research. In the second section we set forth definitions and preliminary information relevant to our work. Theorems and observations about the implications of the graph for calculating path supports are discussed in the third section. Section 4 proposes a simple algorithm for calculating path supports. Section 5 introduces the notion of path trees, which provide insight into possible trajectories in a graph, and may be used in future work to reduce the amount of computation

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 78–87, 2012.

required for path support. The paper concludes with a discussion of further avenues of research.

2 Trajectories in Directed Graphs

Unless stated otherwise, vectors in $\mathbb{R}^l$ are row vectors, except for vectors of the form $\mathbf{e}_i$ which are column vectors; the components of $\mathbf{e}_i$ are 0 with the exception of the ith component that equals 1, for $1 \leq i \leq l$.

Let $\mathcal{G} = (V, E)$ be a finite directed graph without loops having the set of vertices V and the set of edges $E \subseteq V \times V$. We assume that $|V| = n$ and $|E| = m$. If $e_k = (v_i, v_j) \in E$, we refer to v_i as the *source* of e_k and to v_j as the *target* of e_k. This defined the mappings $\mathsf{source} : E \longrightarrow V$ and $\mathsf{target} : E \longrightarrow V$ given by $\mathsf{source}(e_k) = v_i$ and $\mathsf{target}(e_k) = v_j$ for $1 \leq k \leq |E|$.

The *set of outgoing edges of a vertex* v_i is $\mathsf{out}(v_i) = \{e \in E \mid \mathsf{source}(e) = v_i\}$, while the set of *incoming edges of* v_i is $\mathsf{inc}(v_i) = \{e \in E \mid \mathsf{target}(e) = v_i\}$. The *out-degree of a vertex* v_i is the number $\mathsf{d}_+(v_i) = |\mathsf{out}(v_i)|$; *in-degree of* v_i is the number $\mathsf{d}_-(v_i) = |\mathsf{inc}(v_i)|$.

If D is a set of vertices in $\mathcal{G} = (V, E)$, denote by $\mathcal{G}_D$ the subgraph of $\mathcal{G}$ determined by the set D, $\mathcal{G}_D = (D, (D \times D) \cap E$. The previous notations are extended for D by defining the *set of outgoing edges of* D as

$$\mathsf{out}(D) = \{e \in E \mid \mathsf{source}(e) \in D, \mathsf{target}(e) \notin D\},$$

while the set of *incoming edges of* D is

$$\mathsf{inc}(D) = \{e \in E \mid \mathsf{target}(e) \in D, \mathsf{source} \notin D\}.$$

A *trajectory* in the graph $\mathcal{G}$ is a sequence of edges $(e_1, \ldots, e_p)$ such that $\mathsf{target}(e_i) = \mathsf{source}(e_{i+1})$ for $1 \leq i \leq p-1$ and no vertex occurs twice in the sequence

$$(\mathsf{source}(e_1), \ldots, \mathsf{source}(e_p), \mathsf{target}(e_p)).$$

The directed graph $\mathcal{G}$ is represented by its incidence matrix $C_{\mathcal{G}} \in \{-1, 0, 1\}^{n \times m}$ defined as

$$(C_{\mathcal{G}})_{pk} = \begin{cases} -1 & \text{if } \mathsf{source}(e_k) = v_p, \\ 1 & \text{if } \mathsf{target}(e_k) = v_p, \\ 0 & \text{otherwise.} \end{cases}$$

If the graph is clear from context, the subscript $\mathcal{G}$ will be omitted.

Note that each column corresponds to an edge e_k and contains exactly two non-zero numbers that correspond to the source and the target of e_k. Each row corresponds to a node of the graph and contains a -1 for each edge that exits the node and an 1 for each edge that enters the node.

A trajectory is represented by a sequence $\mathbf{t} = (t_1, \ldots, t_m) \in \{0, 1\}^m$, where $m = |E|$, given by

$$t_k = \begin{cases} 1 & \text{if } e_k \text{ occurs in the trajectory,} \\ 0 & \text{otherwise.} \end{cases}$$

for $1 \leq k \leq m$.

Example 1. Consider the directed graph given in Figure 1 which has seven vertices and 14 edges. The incidence matrix $C \in \mathbb{R}^{7\times 14}$ is:

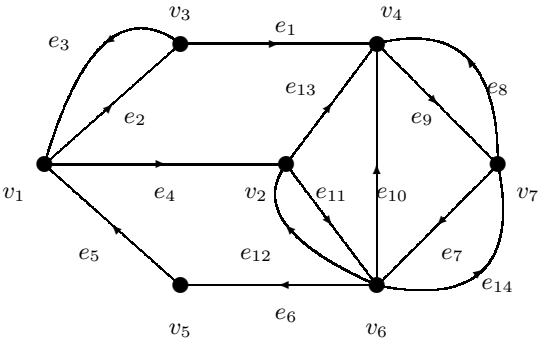


Fig. 1. Directed Graph

$$C = \begin{pmatrix} 0 & -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 0 \\ -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & -1 & 1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Theorem 1. *Let $\mathcal{G} = (V, E)$ be a directed finite graph with $|V| = n$ and $|E| = m$. If $\boldsymbol{t} \in \{0, 1\}^m$ represents a trajectory in the graph $\mathcal{G}$ that departs from the vertex v_i and ends in vertex v_j then*

$$C\boldsymbol{t}' = -\boldsymbol{e}_i + \boldsymbol{e}_j.$$

A trajectory table for a directed graph $\mathcal{G} = (V, E)$ is a table whose attributes are the edges of a directed graph and whose rows are trajectories. For instance, the following matrix T is a table of trajectories in the graph $\mathcal{G}$:

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}
$\mathbf{t}_1$	1	1	0	0	0	0	1	0	1	0	0	0	0	0
$\mathbf{t}_2$	0	0	0	1	1	1	0	1	1	0	0	0	1	0
$\mathbf{t}_3$	0	0	0	1	0	0	0	1	1	1	1	0	0	0
$\mathbf{t}_4$	0	0	0	1	0	0	0	0	1	0	0	0	1	0
$\mathbf{t}_5$	0	0	0	0	0	0	1	0	0	1	0	0	0	0
$\mathbf{t}_6$	0	0	0	0	0	0	1	0	0	0	0	1	1	0

By Theorem 1, the matrix CT' gives the extremities of the paths specified above

$$CT' = \begin{pmatrix} -1 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & -1 \end{pmatrix}$$

The first, third and fourth columns refer to paths that start in v_1 and end in v_6, v_4 and v_7, respectively.

3 Support for Edge Sets

If D is a set of vertices in $\mathcal{G}$ and no trajectory begins or ends in D, then

$$\sum_{e \in \mathsf{inc}(D)} \mathsf{supp}(e) = \sum_{e \in \mathsf{out}(D)} \mathsf{supp}(e).$$

Let $E_{\mathbf{t}}$ be the set of edges that occur on a trajectory $\mathbf{t}$ in a directed graph $\mathcal{G} = (V, E)$. A set of edges K of a directed graph $\mathcal{G} = (V, E)$ occurs on a trajectory $\mathbf{t} = (e_1, \ldots, e_p)$ if $K \subseteq E_{\mathbf{t}}$.

The T-*support* of K is

$$\mathsf{supp}_T(K) = |\{\mathbf{t} \text{ in } T \mid K \subseteq E_{\mathbf{t}}\}|.$$

It is immediate that the support function $\mathsf{supp}_T : \mathcal{P}(E) \longrightarrow \mathbb{N}$ is anti-monotonic, that is, $E_1 \subseteq E_2$ implies $\mathsf{supp}_T(E_2) \leq \mathsf{supp}_T(E_1)$ for $E_1, E_2 \subseteq E$.

Unlike, the similar problem involving market basket, there exists certain interesting connections between the supports of edge sets motivated by the underlying graph structure.

Theorem 2. *Let $\mathcal{G} = (V, E)$ be a directed tree having the root v_0 and the set of leaves $\{u_1, \ldots, u_\ell\}$. The support of any path that joins v_0 to a leaf u_p equals* $\min\{\mathsf{supp}(e) \mid e$ *occurs on the path joining v_0 to u_p*$\}$.

Note that any directed graph has a cover that consists of directed trees because the edges of the graph yield such a cover.

4 A Simple Algorithm for Support Computation

Path supports are recorded by the object `supports` that consists of a hash map h such that $h(\wp) = \mathsf{supp}(\wp)$ for any path $\wp$, and a method `update` which sets the support of the paths.

Let Ξ be a set of pairs of the form $(\wp, s)$, where $\wp$ is a path and $s \in \mathbb{N}$. The call `supports.update`(Ξ) sets the supports of the paths that occur in the first components of the pairs of Ξ to the values specified by the second components of these pairs, respectively. When this method is called as `supports.update`$(\wp, s)$ we assume that $\Xi = \{(\wp, s)\}$. The function `recursive-traversal` takes as arguments a set of paths T, a vertex v, a path $\wp$ that ends in v, a minimal level of support θ and performs the computation shown in Algorithm 1.

The function `recursive-traversal` is used in the function `traverse` which starts with a set of paths T, a vertex v and a minimal support θ and computes the supports of the θ-frequent paths that emerge from v. The pseudocode of this function is shown in Algorithm 2.

Data: $T, v, \wp, \theta$
Result: The `supports` mapping
initialize `supports`;
foreach *edge* $e \in \mathit{out}(v)$ **do**
  $\wp = (\wp, e)$;
  $s = \mathsf{supp}(\wp)$;
  `supports.update`$(\wp, s)$;
  if $s > \theta$ **then**
    $\Xi =$ `recursive-traversal`$(T, \mathsf{target}(e), \wp, \theta)$;
    `supports.update`(Ξ);
  end
  $\wp = \wp - \{e\}$;
end
return `supports`;

Algorithm 1. The recursive function `recursive-traversal` computes the supports of all θ-frequent paths that extend a given path $\wp$

Data: Initial vertex v and minimal support threshold θ
Result: The `supports` mapping
initialize `supports`;
$T' = \emptyset$;
$C = incidence\_matrix$;
foreach *path* $p \in T$ **do**
  if $(Ct)_v == -1$ **then**
    $T' = T' \cup \{p\}$;
  end
end
$\Xi =$ `recursive-traversal`$(T', v, \emptyset, \theta)$;
`supports.update`(Ξ);
return `supports`;

Algorithm 2. The function `traverse` computes the supports of all θ-frequent paths from an initial vertex v

For 1000 trajectories and a minimal support of 0.3 the algorithm applied to trajectories that originate in v_1 generates the following results:

Path	Support
(v_1, v_2)	502
$(v_1, v_3), (v_3, v_4), (v_4, v_7), (v_7, v_6)$	354
$(v_1, v_3), (v_3, v_4), (v_4, v_7)$	498
$(v_1, v_3), (v_3, v_4)$	498
(v_1, v_3)	498

The algorithms in this paper were implemented in Python 2.7 and run on a computer with an Intel $i7 \times 980$ @ 3.33 GHz processor running Ubuntu 11.10. An experiment was run on the traverse algorithm on the graph in Figure 1, with trajectories generated as randomly terminated walks starting at vertex $v1$. Results are shown for 100, 1,000, 10,000, 100,000, and 1,000,000 trajectories. The support threshold for these experiments is 0.2.

Table 1. Average running time of `traverse` vs. number of trajectories for minimum support 0.2

Size of data set	100	1,000	10,000	100,000	1,000,000
Time (ms)	9.645	95.331	972.251	9697.326	106538.961

The dependency of the average time is shown in Figure 2.

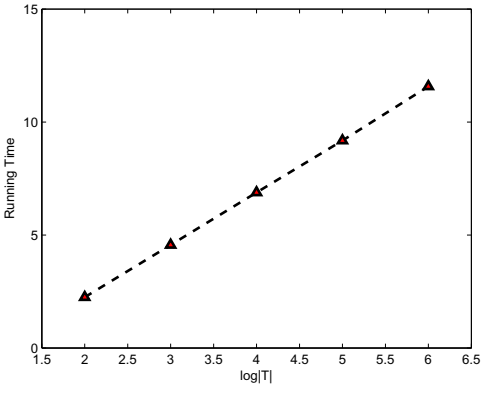


Fig. 2. Dependency of the Average Running Time on the Size of the Set of Trajectories

Table 2. Average running time (ms)/number of maximal frequent paths of `traverse` vs. θ for $|T|$ trajectories

Data size	Minimal Support θ				
	0.01	0.05	0.1	0.2	0.4
1000	102/6	102/6	98/4	95/3	87/2
10000	1026/6	1035/6	990/4	960/2	875/2
100000	10747/6	10853/6	10318/4	9981/3	8975/2

Theorem 3. *Let $\mathcal{G} = (V, E)$ be a directed tree having the root v_0 and the set of leaves $\{u_1, \ldots, u_\ell\}$. The support of any path that joins v_0 to a leaf u_p equals $\min\{\mathsf{supp}(e) \mid e$ occurs on the path joining v_0 to $u_p\}$.*

5 The Path Tree of a Graph

Market basket data studies seek arbitrary frequent item sets. In contrast, we are interested here in supports of sequences of edges that form paths in the traffic graph. Thus, we need to develop an adequate counterpart to Rymon trees that are used in formulating the standard Apriori algorithm [7].

Let P_{v_i} to be the set of all simple paths which originate from vertex v_i. We can visualize P_{v_i} graphically using a tree rooted at v_i. The children of each vertex v_i in this tree are vertices which are direct successors of v_i in the graph and are not ancestors of v_i in the tree.

The *path tree* for paths that start from v_1 in the directed graph given in Figure 1 is shown in Figure 3.

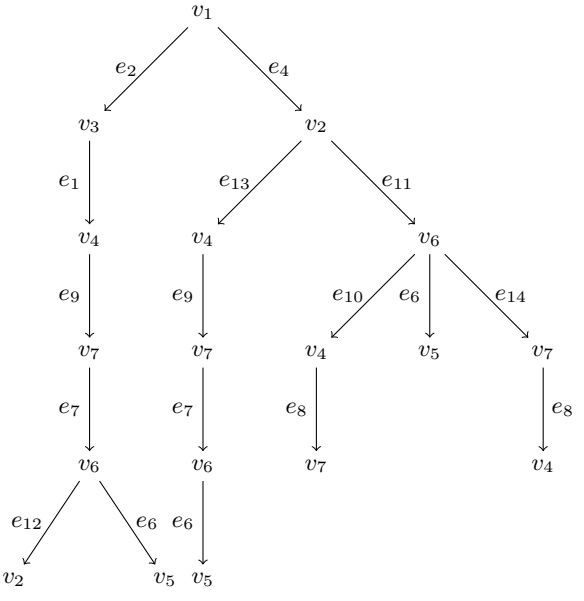


Fig. 3. Path tree for the graph in Figure 1

Note that in the path tree we could have multiple occurrences on an edge. For example, in the tree shown in Figure 3, the edge e_7 occurs twice, on the paths $e_2e_1e_9e_7$ and $e_4e_{13}e_9e_7$.

Theorem 4. *Let $\gamma_{v_i}(e_j) = \{\wp \in P_{v_i} \mid e_j$ is the last edge in $\wp\}$. That is, $\gamma_{v_i}(e_j)$ is the set of all paths which begin at vertex v_i and end with edge e_j. Then, for trajectories beginning at v_i, $\mathit{supp}(e_j) = \sum_{\wp \in \gamma_{v_i}(e_j)} \mathit{supp}(\wp)$.*

Note that when $\mid \gamma_{v_i}(e_j) \mid = 1$, then $\mathsf{supp}(e_j) = \mathsf{supp}(\wp)$ for $\wp \in \gamma_{v_i}(e_j)$. We can use this fact to extrapolate supports for paths which end in unique edges without actually calculating support for such a path.

The following algorithm decreases the number of computations requiring passes through all trajectories, as is required during the computation of support. It does so by using the case in the previous theorem when $\mid \gamma_{v_i}(e_j) \mid = 1$.

Note that using this method requires pre-computation of edge supports, which can be done in one pass.

Using the theorem requires the construction of a path tree rooted at some vertex v_i. However, the path tree can become intractably large. We can limit our attention to the relevant parts of the path tree by halting tree growth before edges which are not θ-frequent are added.

The following algorithm computes the set of maximal paths M_{x_i}.

Data: T, v, θ
Result: M_v
initialize `supports`;
$T' = \emptyset$;
$C = incidence\_matrix$;
foreach *path* $p \in T$ **do**
  if $(Ct)_v == -1$ **then**
    $T' = T' \cup \{p\}$;
  end
end
$s = \sum_{t \in T'} t$;
$\xi = \{(e_i, s_i) \mid s_i \in s\}$;
`supports.update`(ξ);
$M_v = \emptyset$;
foreach *edge* $e \in$ *out*(v) **do**
  if *supp*$(e) > \theta$ **then**
    $M_v = M_v \cup$ `path-traverse`$(\texttt{supports}, e, \theta)$;
  end
end
return M_v;

Algorithm 3. The function `max-path` computes the set of all maximal θ-frequent paths M_v that originate from vertex v

Data: `supports`, $\wp, \theta$
Result: set of paths M
$M = \emptyset$;
foreach *edge* $e \in$ *out*(*target*$(lastedge(\wp))$ **do**
  if *supports*$(e) > \theta$ *and* *target*$(e) \notin \wp$ **then**
    $M = M \cup$ `path-traverse`$(\texttt{supports}, (\wp, e), \theta)$;
  end
end
if $M = \emptyset$ **then**
  return $\{\wp\}$;
else
  return M;
end

Algorithm 4. The recursive function `path-traverse` traverses all θ-frequent paths starting with $\wp$, and returns a set of paths M which can not be extended with θ-frequent, non-repeated edges.

For 10,000 trajectories and a minimal support of 0.1 the `max-path` function returns the following table containing the maximal paths that start from v_1:

Maximal Path
$(v_1, v_2), (v_2, v_4), (v_4, v_7), (v_7, v_6), (v_6, v_5)$
$(v_1, v_2), (v_2, v_6), (v_6, v_5)$
$(v_1, v_3), (v_3, v_4), (v_4, v_7), (v_7, v_6), (v_6, v_2)$
$(v_1, v_3), (v_3, v_4), (v_4, v_7), (v_7, v_6), (v_6, v_5)$

The dependency of the average running time versus the size of the data set for a minimal support level of 0.2 is shown in Table 3.

Table 3. Average running time of `max-path` vs. number of trajectories for minimum support 0.2

Size of data set	100	1,000	10,000	100,000	1,000,000
Time (ms)	0.173378	1.099992	9.958768	100.506067	996.609592

The dependency of the average running time and the number of maximal paths on the size of the data set and the minimal support is presented in Table 4.

Table 4. Average running time (ms)/number of maximal paths of `max-path` vs. θ for $|T|$ trajectories

Data size	Minimal Support θ				
	0.01	0.05	0.1	0.2	0.4
1000	1.14/6	1.14/6	1.11/4.1	1.11/3	1.06/2
10000	10.10/6	10.03/6	10.03/4	10.25/3	10.57/2
100000	102.38/6	104.12/6	107.64/4	102.07/3	102.01/2
1000000	1002.63/6	1002.18/6	1002.06/4	1002.22/3	1002.33/2

6 Conclusion and Further Work

There are many issues left to investigate. Support may be defined for a variety of types of sets of edges and connections between supports for various sets of edges out to be analyzed and used to simplify algorithms for computing supports.

Association rule need to be explored in this context. Connections between the confidence of rules of the form $\wp \to e_1, \ldots, \wp \to e_h$, where $e_1, \ldots, e_h$ are edges that continue the path $\wp$ can be used to simplify the computation of confidence of such rules.

A measure of "attractiveness" can be introduced for path that join two vertices v_1 and v_2. The tradeoff between the length of the path and the support of the path (which shows the number of drivers that take the path) can be used for defining such a measure.

References

1. Adamo, J.-M.: Data Mining for Association Rules and Sequential Patterns. Springer, New York (2001)
2. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Buneman, P., Jajodia, S. (eds.) Proceedings of the 1993 International Conference on Management of Data, Washington, D.C, pp. 207–216. ACM, New York (1993)
3. Giannotti, F.: Mobility, data mining and privacy: Mining human movement patterns from trajectory data. Extraction et gestion des connaissances (EGC 2011), Actes, 25 au 29 janvier 2011, Brest, France. Revue des Nouvelles Technologies de l'Information, vol. RNTI-E-20, pp. 5–6. Hermann-Éditions (2011)
4. Giannotti, F., Nanni, M., Pedreschi, D., Pinelli, F., Renso, C., Rinzivillo, S., Trasarti, R.: Unveiling the complexity of human mobility by querying and mining massive trajectory data. VLDB Journal 20(5), 695–719 (2011)
5. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: Chen, W., Naughton, J.F., Bernstein, P.A. (eds.) Proceedings of the ACM-SIGMOD International Conference on Management of Data, Dallas, TX, pp. 1–12. ACM, New York (2000)
6. Mannila, H., Toivonen, H.: Levelwise search and borders of theories in knowledge discovery. Technical Report C-1997-8, University of Helsinki (1997)
7. Simovici, D., Djeraba, C.: Mathematical Tools for Data Mining. Springer, New York (2008)
8. Trasarti, R., Pinelli, F., Nanni, M., Giannotti, F.: Mining mobility user profiles for car pooling. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, pp. 1190–1198. ACM (2011)
9. Zaki, M.J., Hsiao, C.J.: Efficient algorithms for mining closed itemsets and their lattice structure. IEEE Transactions on Knowledge and Data Engineering 17, 462–478 (2005)

Cyber-Physical Integration to Connect Vehicles for Transformed Transportation Safety and Efficiency*

Daiheng Ni, Hong Liu, Wei Ding, Yuanchang Xie, Honggang Wang, Hossein Pishro-Nik, and Qian Yu

University of Massachusetts, MA, USA
ni@engin.umass.edu, {hliu,hwang1}@umassd.edu, Wei.Ding@umb.edu, Yuanchang_Xie@uml.edu, pishro@ecs.umass.edu, qyu@geo.umass.edu

Abstract. In this paper, we explore a distributed cyber-physical solution using connected vehicle technology (CVtech) to substantially mitigate transportation systems' safety and efficiency problems. Future vehicles, by communicating with other vehicles (V-V), roadside infrastructures (V-R), and personal communication devices (V-P), will adapt to the external regional environment and consequently avoid collisions and congestion. We proposes to seamlessly integrate networked and embedded sensing, computational intelligence, and real-time communication (cyber) into transportation infrastructure including vehicles and roadsides (physical) to facilitate self-organization and system coordination. Specifically, this research addresses two specific themes: Foundations by advancing basic theories in component fields and abstracting the particular knowledge into core principles that integrate cyber and physical processes; and Methods and Tools by designing alternative architectures, modeling a unified online system of cyber and physical elements. The integration of research and education will prepare the future workforce to operate and advance CPS.

Keywords: Cyber-physical system, vehicle communications, data mining.

1 Introduction

On Tuesday, January 17, 2012 at about 4:00 PM, a 23-year-old man from Springfield, MA was speeding in the breakdown lane on Interstate 91 southbound near Exit 17 when his car slammed into the rear of a disabled tractor trailer, according to state police. The man was in a 1995 Nissan Altima with no passenger and he was pronounced dead at the scene. As a result of the accident, the breakdown lane and the far right lane were closed for cleaning up and accident investigation, which caused a major traffic tie-up for about five hours. This tragedy was just one out of 6 million car accidents reported annually in the United States [1], which adds to a total cost of $232 billion with accidents and congestion combined [2].

More than 57% of these car accidents can be directly or indirectly attributed to drivers' inattention, lack of cooperation, and poor decisions[3]. This is so because our

* This work is supported by UMass S&T 2010 grant.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 88–94, 2012.

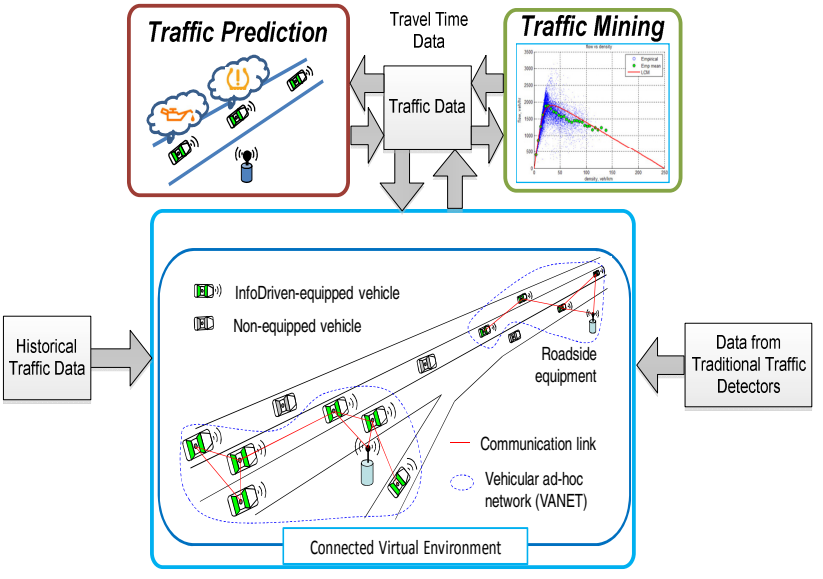


Fig. 1. Traffic Data Mining and Predictions

current transportation systems rely almost exclusively on drivers to monitor their surroundings, decide actions next, execute control maneuvers, and make route choice. Many accidents are resulted from a momentary lapse of attention or a slight misjudgment. To address these human limitations, connected vehicle technology [4] has been proposed and its promising future has been demonstrated by proof-of-concept and pilot studies[8]. However, we have yet to fully exploit its capability by integrating cyber (sensing, computing, and communications) and physical (vehicles and roadside) components. Without this, we cannot proactively warn drivers of an imminent collision and facilitate self-organization among drivers to avoid joining and worsening congestion.

In this paper, we describe a coordinated and distributed transportation cyber-physical system of tomorrow with transformed safety and efficiency. The overall objective of this proposed research is to advance the Science of Cyber-Physical Systems (CPS), the first of NSF CPS Research Target Areas, by providing a unified perspective to capture interacting dynamics of a connected vehicle paradigm, see Figure 1. Our proposed system will enable an ever-vigilant CPS Co-Driver which is able to assist its human driver by proactive safety hazard warning and self-organization for optimal routes. Compared with existing knowledge base, the work in the paper is unique and transformative because we not only integrate knowledge of component fields into each other to advance the state of the art of individual field but also merge these fields into a Science of Cyber-Physical Systems in transportation.

2 Enable Real-Time and Reliable Vehicle Communication

Highway safety applications impose strict requirements of timeliness and reliability on vehicle communication. Although dedicated short-range communication (DSRC) protocol has been developed in IEEE 802.11p, it does not consider the influence of traffic dynamics on data rate and channel access. In addition, the protocol is designed for one-hop applications, while a reliable multi-hop scheme is required to minimize communication delay within traffic. The objective of this aim is two-fold: (a) to quantify the optimal data rate and channel access probability by integrating traffic dynamics into DSRC protocol, and (b) to determine the optimal selection of relay vehicles for reliable multi-hop communications. At the end of this aim, we expect to have an optimized DSRC scheme to ensure real-time and reliable vehicle communication. Our approach (a) is to incorporate vehicle dynamic positions and status into VANET design. Consider a traffic stream in local equilibrium where all vehicles move with the same speed v and inter-vehicle spacing x. Assume that a vehicle (with ID 0) suddenly brakes with deceleration rate b at time $t = 0$. The driver of the following vehicle (with ID 1) sees the braking light and applies brake after some perception-reaction time of τ_{pr} seconds. Without inter-vehicle communications, the collision between the first two vehicles will inevitably cause a chain of collisions among subsequent vehicles. However if vehicles are able to communicate, a trailing vehicle $i, i > 1$ is able to start slowing down only after $\tau_{pr} + \tau_c(i)$ seconds after the braking of vehicle 0 where $\tau_c(i)$ is the incurred delay of communications to inform vehicle i. Hence as it can react well before observing the brake lights of its immediate leader, the probability of a collision is reduced.

Further, system integration with traditional layered approaches often loses the timing efficiency provided by DSRC at lower layer. The team proposes a joint design of MAC, mobility prediction, resource reservation, and congestion control protocols for strict real-time applications. The real-time performance can be guaranteed if control signaling messages are exchanged in a cross-layer manner. A real-time control engine will be developed to achieve real-time communications with four components: mobility prediction, resource reservation, network traffic congestion control, and real-time MAC protocol. The proposed MAC protocol is a time-bounded protocol. The prerequisite of V-V communication for traffic safety applications is timely medium-access. We propose to design a time-bounded medium-access control protocol with traffic control and resource reservation in a cross-layer manner as shown in Fig. 2. Mobility predictions can be done based on the history of vehicle locations and vehicle speeds. The limited bandwidth resources will be preserved through time slot assignments to achieve guaranteed time-bounded message delivery among vehicles. The prerequisite of DSRC for traffic safety applications is timely medium-access. An effective design is time-bounded medium-access control protocol with traffic control and resource reservation in a cross-layer manner. To do so, traffic dynamics is predicted using historical vehicle positions and speeds; the limited bandwidth resources is prereserved through time slot assignments to achieve guaranteed time-bounded message delivery among vehicles; high-level congestion traffic control can be implemented through traffic classification and priority-based delivery.

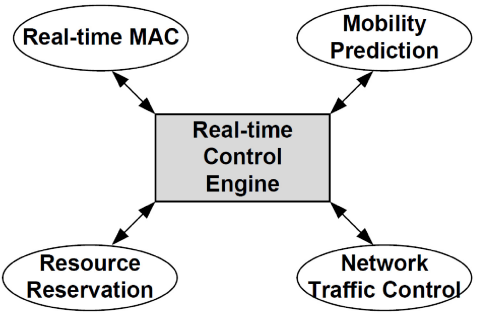


Fig. 2. High-Level Model of Real-Time Communication Control Engine

3 Mine Traffic Data for Dynamic Vehicle Routing

Connected vehicles are able to generate detailed and very accurate traffic data in real time. These data, if properly utilized, can greatly improve transportation efficiency. Unfortunately, the efficiency impact of the connected vehicle technology hasn't been given the deserved attention. Existing travel time prediction and dynamic vehicle routing models are not ready to fully embrace the opportunities made possible through connected vehicles.

As shown in Figure 1, roadside equipment (RSE) units deployed at strategic locations exchange information with OBEs installed on passing by vehicles. Both RSEs and neighboring OBEs are interconnected and share traffic information. Vehicles outside the range of any RSE may still be connected to the rest of the vehicle and infrastructure network via neighboring vehicles. This vehicle and infrastructure network can generate very accurate traffic information (i.e., vehicle trajectories) in great detail, based on which some fundamental traffic problems related to efficiency can be well addressed from a brand new perspective, including: (a) How to accurately infer current and predict future traffic conditions at locations with and without RSE coverage; and (b) How to best utilize the inferred and predicted traffic information for improving traffic operations. Proposed solutions to these questions based on the CVTech are detailed below. Traffic condition at a location can be represented as a variable vector $x = \{xt, x_{t-1}, x_{t-2}, \ldots, x_0,\}$ where time t represents now and t-1 a moment ago. In addition x_t itself is a vector, e.g. $x_t = \{l_t, q_t, k_t, v_t\}$ where l, q, k, v denote traffic location, flow, density, and speed, respectively. The objective here is to predict into the future, e.g. finding $x_{t+1}, x_{t+2}, \ldots$ based on what is known.

Traffic data are collected across time and space. The data analysis must take account of spatial and temporal autocorrelation, which are the characteristics at neighborhood locations that are often positively or negatively correlated. Thus reliable traffic prediction should depend on previous traffic information and vehicles in close vicinity. A traditional model of traffic prediction only on temporal information could yield unstable parameter estimates. Such a method is defined as a function f to calculate an estimate of x at time t+1, using the (d-1)k time steps back from time t. Thus we have $x_{t+1}=f(x_t,x_{t-k},$

…,x_t-(d-1)k) where d is the number of inputs and k is the time delay. In this paper, we proposes a new method to use temporal, spatial, traffic flow, density, and speed information simultaneously using tensor feature regression. Tensors of higher orders have been proved to be effective data structures to model complex science and engineering problem. We extract five-order (traffic location, flow, density, speed, time) tensor features to represent traffic information. The five-order tensor is a multi-dimension matrix to store traffic information uploaded from OBEs to RSEs. It includes time series data in the same location as temporal information, and it includes the surrounding traffic information at the same time as spatial information. In the new tensor regression approach, the predicted traffic information at time t+1 is calculated as $x_{t+1}=f(N(x_t),N(x_{t-k}), \ldots,N(x_{t-(d-1)k}))$ where $N(x)$ is the surrounding vicinity locations of location x, and t varies between d time-steps. A tensor regression method can then be built. In a matrix formulation for such a regression model, we have $Y=xW+e$, where Y is the predicated traffic data, W is the parameter vector to be estimated during the learning stage, e consists of residuals. In the learning stage, all the historical data are available and can be built in d time-steps back. W is estimated to fit the model by the Ordinary Least Squares method, $W=(x^Tx)^{-1}x^TY$. A predicated value is given by $Y'=xW$, and the residual is $e=Y-Y'$. Every time a vehicle with OBE passes by a RSE, it receives updated traffic information from and sends its origin, destination, and vehicle trajectory data to the RSE. From the trajectory data, a lot of useful information can be obtained, including speed, acceleration/deceleration rate, and travel times of upstream segments in the past few minutes. Such information can be correlated with the traffic data obtained from upstream/downstream RSEs and traditional loop detectors. A mathematical relationship can then be established using the tensor regression method. The RSE mines the trajectory and loop detector data and continuously provides estimates and predictions of the traffic states of areas not covered by RSEs.

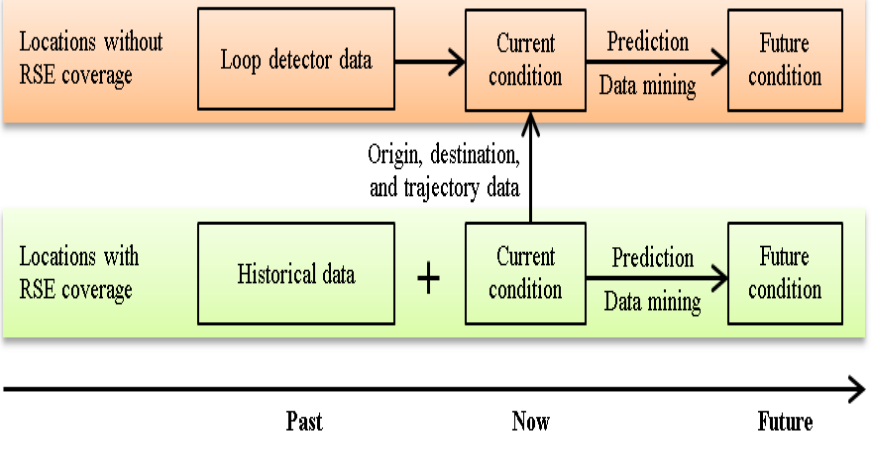


Fig. 3. Traffic Estimation and Prediction for Areas not Covered by RSEs

4 Coordination by Traffic Flow Optimization

Building on the above studies, scenario-based analysis can be conducted to search for an optimized future state if drivers are willing to perform system-wide coordination. More specifically, a few schemes can be identified to allocate traffic to less congested routes and revoke the simulation to update system state. Once an optimal scheme is found, re-routing information and associated benefits are disseminated to drivers for them to make educated choice. Many congestion problems can be effectively addressed by allowing self-organizing and system-wide coordination. System-wide coordination requires optimized traffic assignment among alternative routes and such a problem can be formulated as follows. In order achieve personal goal (e.g., shortest travel time), one needs to minimize objective function $miny(x) = \sum_n \int_0^{x_n} t_n\,(q)dq$, where n denotes a specific route, q denotes flow, $t_n(q)$ denotes travel time given n and q, and x_n assignment of flow on route n. Once optimized traffic assignment is found, incoming drivers are prompted with recommended route choices and associated benefits. To simplify the problem, the optimization is performed at RSEs and concerns only their local networks.

5 Conclusions

In this paper, compared with existing knowledge base, the proposed research is *unique* and *transformative* because we not only integrate knowledge of component fields into each other to advance the state of the art of individual field but also merge these fields into a *Science of Cyber-Physical Systems* in transportation. We studied traffic-responsive real-time and reliable vehicle communication scheme, field theory model built on connected vehicles, networked vehicle routing algorithms. The outcomes of this research are to revolutionize vehicle automation and highway safety. The proposed methods and tools for interdisciplinary problem solving are extensible and transformative. Less highway accidents and reduced congestion resulted from this research amount to billions of dollars savings, let alone improved quality of life and boosted national economy.

References

[1] BTS, National Transportation Statistics 2011, Bureau of Transportation Statistics (2011), http://www.bts.gov/publications/national_transportation_statistics/
[2] Clifford, C.: U.S. car accident cost, CNNMoney.com (2008)
[3] NHTSA, Distracted driving, National Highway Traffic Safety Administration (NHTSA), http://www.distraction.gov/index.html
[4] USDOT, Connected Vehicle Technology, http://www.its.dot.gov/connected_vehicle/connected_vehicle.htm

[5] Ni, D., Pishro-Nik, H., Prasad, R., Kanjee, M.R., Zhu, H., Nguyen, T., Nekoui, M.: Development of a VII-Enabled Prototype Intersection Collision Warning System. International Journal of Internet Protocol Technology 4(3), 1–8 (2009)
[6] RITA, Safety Pilot Program Overview, http://www.its.dot.gov/safety_pilot/index.htm
[7] Schagrin, M., Oliver, M.: Cooperative Intersection Collision Avoidance Initiative Presentation, http://www.its.dot.gov/cicas/
[8] Kiley, D.: An Option on New Cars Now Proven to Save Lives, July 19 (2011), http://autos.aol.com/article/radar-based-collision-avoidance/

Multi-modal Remote Sensing System for Transportation Infrastructure Inspection and Monitoring

Tzu-Yang Yu[1], Christopher Niezrecki[1], and Farhad Ansari[2]

[1] The University of Massachusetts Lowell, Lowell MA 01854, USA
{Tzuyang_Yu,Chritopher_Niezrecki}@uml.edu
http://tyu.eng.uml.edu/
http://sdasl.uml.edu/
[2] The University of Illinois Chicago, Chicago IL 60607, USA
FAnsari@uic.edu
http://www.uic.edu/depts/cme/people/faculty/fansari.html

Abstract. Managing the growing population of deteriorated transportation infrastructure systems (i.e. highway bridges) and being able to accurately inspect them in a timely and cost effective manner is a major societal challenge within the United States today. A multi-modal remote sensing system (MRSS) that will be used as the next generation of rapid, distant, interrogation technology for bridge inspection is proposed. In the proposed MRSS technology, advantages of nondestructive testing (local inspection) and structural health monitoring (global, continuous monitoring) are combined by using continuous wave imaging radar (CWIR), digital image correlation (DIC), and fiber optic sensors (FOS). MRSS represents the next-generation of portable bridge inspection technology for efficient inspection, evaluation and rating of bridges.

Keywords: multi-modal remote sensing, continuous wave imaging radar, digital image correlation, fiber optic sensors.

1 Condition Assessment of Aging Civil Infrastructure

Our transportation infrastructure has long been the largest investment and the backbone for supporting economic and technological developments in the Nation. After experiencing the most intensive construction activities in the past fifty years, many of our critical transportation infrastructure systems, such as highway bridges, have been deteriorated with their remaining capacity at a dangerous and uncertain level. Today, managing the growing population of deteriorated highway bridges and being able to inspect them in a timely fashion is a major societal challenge, especially in a financially difficult time. Traditional nondestructive testing/inspection/evaluation (NDT/I/E) methods for important structures like highway bridges cannot provide timely evaluation to prevent deteriorated bridges from sudden collapse. Automated, low-cost, efficient bridge inspection techniques for interrogating critical bridge components are needed.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 95–103, 2012.

Critical transportation infrastructures like highway bridges can fail not only due to superstructure (girder, pier) failures, but also due to substructure (foundation) failures. Failures of highway bridges can be attributed to various causes including natural hazards (e.g., earthquake, storm, hurricane, flooding), service loads (e.g., dead load, traffic, snow, wind), man-made accidents (e.g., fire), and terrorism (e.g., explosion), in conjunction with material deterioration such as concrete cracking and spalling and steel corrosion. It is realized that bridges can be damaged or fail due to various causes, and each cause can affect more than one bridge element to different extents. Research also has shown that material deterioration changes the deterioration rate of different highway bridges. In a recent study based on the survey conducted by the National Bridge Inventory (NBI), the source of structural deficiency in highway bridges are more due to bridge superstructure and substructure than due to bridge deck [1]. It is found that, in order to effective and efficient management of bridges, superstructure and substructure deserve significant maintenance effort.

2 Nondestructive Evaluation/Testing for Highway Bridges

Existing highway bridge inspection techniques used by State Department of Transportation (DOT) and County DOT include visual inspection, mechanical sounding, rebound hammer, cover meter, electrical potential measurements, and ultrasonics (impact-echo and pulse velocity) [2]. Among them, visual inspection and mechanical sounding are most frequently used. When applied to the condition assessment of highway bridge girders and piers, accessibility of inspected locations is the major challenge. With current technology, visual inspection and mechanical sounding are time consuming, labor intensive, and cost inefficient for the sake of requiring specialized lifts to carry inspectors to reach the locations to be inspected. Safety issue, interference with existing traffic, and subjective evaluation (visual inspection) are additional disadvantages in such inspection. Other NDE/T techniques including ground penetrating radar (GPR), radiography (neutron, X-ray, gamma ray), infrared (IR) thermography, laser vibrometry, magnetic particle, acoustic emission, dye penetrant, and eddy current are only applied for experimental purpose and on a case-by-case basis. Table 1 compares existing NDE/T techniques for their applicability for highway bridge girders and piers. Another approach is structural health monitoring (SHM). Discrete (e.g., strain gauges, accelerometers) and continuous (e.g., fiber optic) sensors have been applied in the past. Wireless strain gauge systems have also been proposed [2]. Fiber optic sensors (intrinsic/extrinsic; Fabry-Perot, Bragg grating, optical time domain reflectometry) can provide accurate measurements on many physical and chemical quantities (serving as intrinsic or extrinsic sensor). In general, wireless sensors are applicable for transportation infrastructure applications, although current technology requires extensive deployment of sensors to achieve sufficient levels of detectability for local defects. Meanwhile, power supply and long-term maintenance are key issues for wireless sensors.

Table 1. Comparison of existing NDE/T techniques for highway bridge girders/piers

Technique	Applicable mater.	Material prop.
Visual inspection	Concrete, steel	Optical
Mechanical sounding	Concrete, steel	Mechanical
Rebound hammer	Concrete	Mechanical
Cover meter	Concrete	Electrical
Half-cell potential sensor	Concrete, steel	Electrical
Ultrasound	Concrete, steel	Mechanical
Ground penetrating radar	Concrete	Electromagnetic
Radioactive	Concrete, steel	Nuclear
Thermography	Concrete, steel	Thermal
Laser vibrometry	Concrete, steel	Optical
Magnetic particles	Steel	Magnetic
Acoustic emission	Concrete, steel	Magnetic
Dye penetrant	Concrete, steel	Geometric
Eddy current	Steel	Electrical

In view of the needs and the importance of critical transportation infrastructure like highway bridges, the objective of this project is to develop a multi-modal remote sensing system (MRSS) as the next generation of rapid, distant, nondestructive inspection technology for bridge inspection. The proposed MRSS combines advantages of NDT (localized inspection) and SHM (global, continuous monitoring) and represents a next-generation handheld inspection technology for efficient inspection, evaluation and rating of bridges. Overview of MRSS is described in the following section.

3 Technical Approach of MRSS

MRSS consists of i) an integrated sensor component with continuous wave (CW) imaging radar, high-resolution optical camera, global positioning system (GPS), and a laser ranger for in-situ inspection, and ii) a fiber optics sensor (FOS) component for continuous monitoring. The FOS component provides continuous data at low cost for long-term monitoring, while the integrated sensor component provides discrete data for short-term inspection. In the design of MRSS, the FOS component triggers short-term inspection using the integrated sensor when irregular loading events are observed, based on the continuously-collected

structural response. In this scheme, the FOS component complements the integrated sensor component. In the following, principles of these components in MRSS are provided.

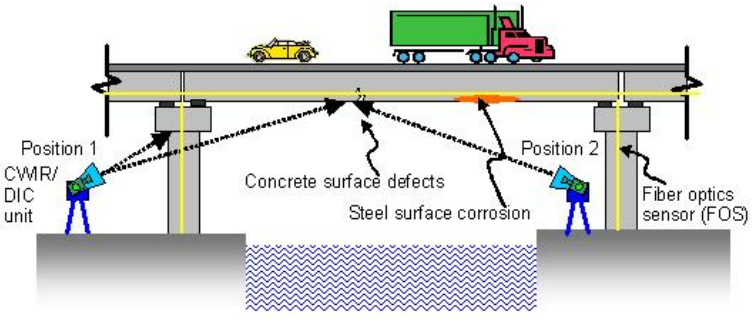


Fig. 1. Inspection and monitoring scheme of MRSS

3.1 Continuous Wave Imaging Radar

In the design of a continuous wave imaging radar (CWIR) sensor, with the reflection measurements at different frequencies, relative elevations, inspection angles, and signal polarizations, subsurface imaging of inspected structures are performed using synthetic aperture radar (SAR) algorithms. Radar image resolutions are improved with the visual images provided by the high-resolution digital image correlation sensor. Depending on the information the radar needs to learn from the returned signal, waveform types, frequency, and bandwidth can be designed to achieve the goal. In the hardware design of CWIR, (a) high frequency (Ku-band, 12 18 GHz) and (b) wide bandwidth waveform are chosen for their sensitivity to surface backscattering changes in radar signals. Phased array antenna is applied in CWIR to the radar scanning function, aided by a mechanically controlled system to improve the focusing capability of CWIR. In the design of CWIR, a laser ranger is included and collocated with the radar system to provide exact range gating control at each scanning point regardless of the alignment between the radar and the structure. The laser ranger helps the receiver ranging gating and rejecting unwanted returned signals from unknown sources.

In addition to the use of raw returned signal for damage detection, further processing of radar signals using synthetic aperture radar (SAR) imaging and backprojection algorithms are applied. SAR imaging and backprojection algorithms have been demonstrated as a promising technique in the subsurface sensing of construction materials [3]. It is found that post-processing technique can be used to reveal the presence and location of near-surface defects in a multilayer dielectric structure like GFRP (glass fiber reinforced polymer)-concrete [4]. Primary steps in SAR imaging are described in the following.

Define the time-independent SAR point response $S(x, y\sin\theta_i)$ collected by CWIR sensor as

$$S(x, y\sin\theta_i) = \operatorname{sinc}\left(\frac{\pi y\sin\theta_i}{\rho_r}\right)\operatorname{sinc}\left(\frac{\pi x}{\rho_{xr}}\right) \tag{1}$$

It is equivalent to express $S(x, y\sin\theta_i)$ in $S(\bar{r}_{s,j})$ in which $\bar{r}_{s,j}$ is the distance of the point scatterer in the slant plane. ρ_r and ρ_{xr} are the range and cross-range resolutions of the radar, respectively. θ_i is the incident angle with respect to the vertical axis (z axis).

$$S(\bar{r}_{s,j}) = \operatorname{sinc}\left(\frac{\pi R_{s,j}}{\rho}\right) \tag{2}$$

where $R_{s,j} = |\bar{r}_{s,j}|$. $\rho = \sqrt{\rho_r^2 + \rho_{xr}^2}$. The time-dependent $S(\bar{r}_{s,j})$ can be written by

$$S(\bar{r}_{s,j}, t) = \frac{1}{R_{s,j}^2}\int_{\omega_c - \pi B}^{\omega_c + \pi B} d\omega \cdot \exp\left[i\omega t\right] \tag{3}$$

The range compression on $S(\bar{r}_{s,j}, t)$ is conducted by shifting the time t to $\hat{t} = t - \dfrac{2R_{s,j}}{c}$. c is the speed of light and ω the radian frequency variable. ω_c is the carrier frequency (in radian). B is the frequency bandwidth. This leads to

$$S(\bar{r}_{s,j}, t - \frac{2R_{s,j}}{c}) = S(\bar{r}_{s,j}, \hat{t}) = \frac{1}{R_{s,j}^2}\int_{\omega_c - \pi B}^{\omega_c + \pi B} d\omega \cdot \exp\left[i\omega\left(t - \frac{2R_{s,j}}{c}\right)\right] \tag{4}$$

which gives

$$S(\bar{r}_{s,j}, \hat{t}) = \frac{B}{R_{s,j}^2}\exp\left[i\omega_c\hat{t}\right] \cdot \operatorname{sinc}\left(B\hat{t}\right) \tag{5}$$

where the term $\left(t - \dfrac{2R_{s,j}}{c}\right)$ shifts the origin of traveling time to the time receiving the reflected signals. By shifting the origin of time to the center of the scatterer, the range distance $R_{s,j}$ has been compressed in the way that the center of the reconstructed image coincides with the center of the scatterer.

Express the radar location vector $\bar{r}_s$ by its position on the synthetic aperture, ξ. It is clear that

$$\xi = |\bar{r}_s|\phi_i = R_s\phi_i \tag{6}$$

The total range-compressed focused signal collected at ξ on the aperture, $D(\xi, \hat{t})$, is the integration of the range-compressed focused signal received from all scatterers in the domain Ω_s. In cylindrical coordinate systems,

$$D(\xi, \hat{t}) = \int_0^{R_s} d\bar{r}_j \int_0^{2\pi} d\phi_j \cdot G(\bar{r}_j, \phi_j)S(\bar{r}_{s,j}, \hat{t}) \tag{7}$$

where $G(\bar{r}_j, \phi_j)$ is a scattering amplitude density function at $(\bar{r}_j, \phi_j)$. Or it can be written to be

$$D(\xi, \hat{t}) = \int_{x_{\min}}^{x_{\max}} dx \int_{y_{min}}^{y_{max}} dy \cdot G(x, y) S(\sqrt{((x_s - x_j)^2 + (y_s - y_j)^2)}, \hat{t}) \tag{8}$$

in Cartesian coordinate systems. $D(\xi, \hat{t})$ is the range-compressed focused signal for the radar located at ξ on the aperture, whose time being shifted by $\frac{2R_{s,j}}{c}$. The backprojection processing can be conducted by defining the backprojected signal $B_{BP}(\xi, t)$ as

$$B_{BP}(\xi, t) = C_{BP} \cdot \frac{\partial D(\xi, \hat{t})}{\partial t} \tag{9}$$

where C_{BP} is the backprojection coefficient hereby defined in order to yield an ideal bandpass transfer function and

$$C_{BP} = \frac{1}{i} \left(\frac{4\pi}{c} \right)^2 \tag{10}$$

when an ideal point scatterer with unit cross section at $\bar{r}_j$ is considered and the frequency integration is performed by ω. When the frequency integration is carried out by f,

$$C_{BP} = \frac{1}{2\pi i} \left(\frac{4\pi}{c} \right)^2 \tag{11}$$

$\frac{\partial D(\xi, \hat{t})}{\partial t}$ is defined continuously, while $D(\xi, t)$ is usually discrete. To prevent a potential discontinuity in $\frac{\partial D(\xi, \hat{t})}{\partial t}$ from occurring, a matched filter $M(\hat{t})$ is applied on the derivative.

$$\frac{\partial D(\xi, \hat{t})}{\partial t} = \frac{\partial}{\partial t} \int_0^{\hat{t}} dt' \cdot D(\xi, \hat{t}) M(\hat{t} - t') = \int_0^{\hat{t}} dt' \cdot D(\xi, \hat{t}) \frac{\partial M(\hat{t} - t')}{\partial t} \tag{12}$$

This convolution operation is also advantageous for computational efficiency since $\frac{\partial M(\hat{t} - t')}{\partial t}$ can be calculated in advance. In the frequency domain, convolution is performed by multiplying a phase factor. Finally, the backprojection image is given by

$$I(\bar{r}, \phi) = y \int_0^{R_s \theta_{\text{int}}} d\xi \cdot B_{BP}(\xi, \hat{t}) \tag{13}$$

which is the integration of $B_{BP}(\xi, \hat{t})$ over the entire aperture length $R_s \theta_{\text{int}}$. $I(\bar{r}, \phi)$ is a two-dimensional, spatial image (range vs. cross-range) of the structure.

3.2 Digital Image Correlation Sensor

Digital image correlation (DIC) is a non-contacting full-field optical measurement technique that has the ability to measure the static and dynamic motion of virtually any surface, with a very high spatial resolution [5-10]. A single camera perpendicular to the surface of a flat test object can be used to obtain in-plane displacements and strains. A calibrated stereo pair of cameras can measure out-of-plane displacements, surface strains, and the shape of a complex object. Sample preparation consists of applying (or projecting) a regular or random high-contrast pattern to the surface. Thousands of unique correlation areas known as facets (typically 15 pixels square) are defined across the entire imaging area. The center of each facet is a measurement point that can be thought of as an extensometer and strain rosette. These facet centers are tracked, in each successive pair of images, with accuracy up to 0.001 pixel. Then, using the principles of photogrammetry, the three-dimensional (3D) coordinates of each facet are determined for each picture set. The results are the 3D shape of the component, the displacements, and the in-plane strains. Therefore, deterioration which manifests itself through surface bulging and deformation, crack growth, or wear can be detected unobtrusively. The three key principles of image correlation photogrammetry are: 1) pattern matching to identify targets in the field of view, 2) sub-pixel resolution, and 3) triangulation.

3.3 Fiber Optic Sensors

Optical fiber sensors provide powerful means for ubiquitous monitoring of civil structures. They possess myriads of attributes commensurate in applications to civil structures that include: flexibility and geometric compatibility for embedment or adhesion to structural elements; serve the dual purpose as the sensor and pathway for the signal; distributed and ubiquitous sensing or serial multiplexing of several sensors along one line; extremely small and lightweight; immune to electrical and electromagnetic interference resulting in high signal to noise ratio; resist corrosion and fatigue; and are incapable of initiating fire or explosions since the signal is optical [11]. Flexibility in transduction of optical fibers by various means including intensity, wavelength, interference and frequency demodulation of optical signals has allowed for sensing of a variety of structural perturbations both at static as well as dynamic sampling frequencies.

From a civil engineer's point of view a very effective way to utilize an optical fiber is to integrate a single sensor throughout the structure for detection of all anomalies and cracks. Distributed sensing using Brillouin Scattering based systems have been in existence since the nineties. However, their applications have been limited to specialty areas in petroleum industry. The sensing principle for Brillouin Scattering sensors is based on the frequency shift of the returned optical signal following transmission throughout the entire length of the optical fiber. These systems are capable of monitoring strains and thermal gradients along kilometers of optical cables. A number of methods are available for interrogation of the Brillouin frequency shift. Commercial systems are either BOTDR (Brillouin Optical Time Domain Reflectometry) or BOTDA (Brillouin Optical Time

Domain Analysis) based. BOTDR based systems use a single pulsed laser source, whereas BOTDA systems employ a continuous-power laser source in addition to the pulsed laser in order to enhance the spatial resolution of the measurements. The accuracy of distributed measurement capability of the BOTDR/BOTDA based systems is dependent on their spatial resolutions.

4 Summary

A new condition assessment paradigm combining various sensing technologies is proposed. In the proposed multi-modal remote sensing system (MRSS), an imaging radar, a high-resolution optical camera, a laser ranger, and a fiber optic sensing system are used to relate local measurement with global measurement. This development represents a new approach for long-term bridge maintenance and a platform to combine the technical benefits of conventional NDT and SHM technologies for Remote Sensing and Spatial Information applications.

Disclaimer. The views, opinions, findings and conclusions reflected in this presentation are the responsibility of the authors only and do not represent the official policy or position of the USDOT/RITA, or any State or other entity.

Acknowledgements. The authors want to thank the U.S. Department of Transportation (USDOT) Research and Innovative Technology Administration (RITA) Commercial Remote Sensing and Spatial Information (CRS&SI) program for the financial support via Grant RITARS-11-H-UML. Assistance provided by graduate students Shafique Ahmed, Christopher Noris, Dewei Meng, and Che-Fu Su on project activities is greatly appreciated. This research effort is also partially supported by the UMass Transportation Information Consortium which is sponsored by UMass President's Science and Technology Initiative 2010.

References

1. National Bridge Inventory: Status of the Nation's Highways, Bridges, and Transit – Conditions and Performance, Report to Congress. Federal Highway Administration, Washington, DC, January 22 (2007)
2. Federal Highway Administration: Highway Bridge Inspection – State-of-the-Practice Survey. FHWA-RD-01-033, NDE Validation Center, McLean, VI (2001)
3. Yu, T.-Y.: Distant damage assessment method for multi-layer composite systems using electromagnetic waves. J. Eng. Mech. ASCE 137(8), 547–560 (2011)
4. Yu, T.-Y.: Damage Detection of GFRP-concrete Systems Using Electromagnetic Waves. Lambert Academic Publishing, Koln (2009)
5. Schmidt, T., Tyson, J., Galanulis, K.: Full-field dynamic displacement and strain measurement using advanced 3D image correlation photogrammetry – Part I. Exp. Tech. 27(3), 47–50 (2003)
6. Mikhail, E., Bethel, J., McGlone, J.: Introduction to Modern Photogrammetry. John Wiley and Sons (2001)

7. Peters, W.H., Ranson, W.F., Sutton, M.A., Chu, T.C., Anderson, J.: Application of digital image correlation methods to rigid body mechanics. Opt. Eng. 22(6), 738–742 (1983)
8. Chu, T.C., Ranson, W.F., Sutton, M.A.: Applications of digital-image-correlation techniques to experimental mechanics. Exp. Mech. 25(3), 232–244 (1985)
9. Kahn-Jetter, Z.L., Chu, T.C.: Three-dimensional displacement measurements using digital image correlation and photogrammic analysis. Exp. Mech. 30(1), 10–16 (1990)
10. Luo, P.F., Chao, Y.J., Sutton, M.A., Peters, W.H.: Accurate measurement of three-dimensional deformations in deformable and rigid bodies using computer vision. Exp. Mech. 33(2), 123–132 (1993)
11. Ansari, F.: Practical implementation of optical fiber sensors in civil structural health monitoring. J. Intelli. Mater. Sys. Struct. 18(8), 879–889 (2007)

Operational Support in Fish Farming through Case-Based Reasoning

Axel Tidemann[1], Finn Olav Bjørnson[1], and Agnar Aamodt[2]

[1] CREATE, SINTEF Fisheries and Aquaculture AS, Trondheim, Norway
axel.tidemann@gmail.com, finnolav.bjornson@sintef.no
[2] Department of Computer and Information Science,
Norwegian University of Science and Technology, Trondheim, Norway
agnar@idi.ntnu.no

Abstract. Farmed fish is the third biggest export in Norway (around NOK 30 billion/€3.82 billion/US$ 5.44 billion in 2010), and large fish farms have biomass worth around NOK 150 million/€19.38 million/US$ 26.72 million. Several processes are automated (e.g. the feeding system), and sensory logging systems are becoming ubiquitous. Still, the key to successful management of a site is the operational knowledge possessed by the fish farmers. In most cases, this information is not stored formally. To capture, store and reuse this knowledge in a more systematic way is called for. We present a system that employs case-based reasoning (CBR) for such knowledge management, combined with sensor data and numerical models. The CBR system will ultimately be the core part of a decision support for regional managers surveying fish farming sites. Data is acquired from multiple fish farms, spanning several years. We present recent results in testing how well the CBR system finds similar cases. An important part of this test is the evaluation of three different methods for case retrieval (kNN, linear programming for setting feature weights, Echo State Network).

Keywords: Case-based Reasoning, Decision Support System, Intelligent Fish Farming.

1 Introduction

An average fish farm in Norway contains values of around NOK 50 million/€6.37 million/US$ 8.52 million [8]. Experiences and know-how is an important part of successful management of a site. We present an approach to re-using operational knowledge through case-based reasoning (CBR) [1]. The application of the CBR system described in this paper is fish farm management at a regional level. A regional manager typically surveys 10-20 sites. In addition to getting feedback on the current state of each site, operations that require planning must be scheduled. Such operations include grading (a process where fish are sorted based on size and put in different cages), delousing, deployment, and delivery to processing plants.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 104–113, 2012.

The paper is organized as follows: the background section cites relevant work, then details about the CBR system are presented. Results are demonstrated, before the paper closes with a discussion and pointers to future work.

2 Background

Case-Based Reasoning (CBR) is a reasoning method that solves a new problem by getting a reminding to a similar problem solved before, with a range of more specific methods for accomplishing that task [3]. CBR has a vocabulary to describe a case as consisting of a problem statement, a solution, and an outcome. CBR has a case base of previously solved cases, and the reasoning process can be described by the following four steps [1]:

First, in the RETRIEVE step, a new problem is described as a query case. The most similar case is found by using the system's similarity assessment method. The best case is selected and the REUSE step takes this case and either reuses it directly or adapts it to a solution that fits the query case. The REVISE step takes the solution and evaluates it, for example by applying it to the problem or having it assessed by a domain expert. The final step is RETAIN, which learns from the revised problem solving experience by updating the case base. The latter is the learning step, in which a new case may be added to the case base or other changes may be made.

CBR systems are called "lazy learners" which means that they delay the inductive step until a new problem instance arrives. Similarity assessment is a core problem in all CBR systems, and the methods range from simple, global similarity metrics, to complex algorithms for local similarity that also take situational context into account. An example of a simple global similarity function is shown in Equation (1).

$$CaseSim(C, Q) = \sum_{i=1}^{n} s(c_i, q_i) w_i \tag{1}$$

s is the similarity function for comparing individual features, and w_i is the relative weight for feature i. C is a retrieved case, Q is the query case, each represented as a set of features. c_i is feature i from the retrieved case, q_i feature i from the query case, and n is the total number of features in a case.

Metaxiotis et al. [14] surveyed expert systems and their role in production planning and scheduling. They conclude that expert systems are generally perceived to be very useful in production planning and scheduling. The benefits reported from the use of expert systems include more accurate decisions, time gains, improved quality and more efficient use of resources. They also believe that the usefulness of expert systems can be improved if they are integrated with operations research techniques like simulation. Liao [12] reviewed expert systems and their applications from 1995-2004. He concludes that expert systems methodologies are tending to develop towards expertise orientation and that expert system application development is a problem-oriented domain. He further suggests that different social science areas, such as psychology, cognitive

science, and human behaviour could implement expert systems as another kind of methodology.

There are several practical applications of decision support systems (DSS) that make use of CBR in the literature. Liu et al. [13] describe a system for knowledge support of problem solving in a production process, based on knowledge discovery and case-based reasoning. Raphael et al. [15] describe a system for computing the cost of construction projects, using a case-based reasoning strategy. Shimin et al. [17] seek to combine case-based reasoning and rule-based reasoning for a system for emergency decision making. Arshadi et al. [5] use data mining for case-based reasoning in a biological domain. They conclude that CBR systems perform remarkably well on complex and poorly formalized domains.

There are a few examples of DSS used in an aquaculture setting in the literature. Schulstad [16] describe work done on a DSS for hatchery production management for Atlantic salmon in Norway. Bolte et al. [10] developed decision support tools for aquaculture to assess economic and ecologic impacts of alternative decisions on aquaculture production. Their main approach was a system based on simulation models and enterprise budgeting. Li et al. [11] describes a web-based expert system for diagnosing fish disease in aquaculture facilities in China. One of their main experiences is that a good expert system requires tight cooperation and collaboration among users, human experts, knowledge engineers and system developers.

Combining CBR with low-level sensor data has also been used in the offshore oil industry. Oil well drilling is a complex and costly process, and unplanned downtime is to be avoided. By combining sensor data with historical cases, the drilling process can be monitored to avoid problems in the future, i.e. the system *predicts* possible states that can occur [2,18].

3 CBR for Decision Support in Fish Farming

The work described here is part of an ongoing research project called *SimFrame* (more details in section 5). Part of the SimFrame project is to build a database that contains sensor data from fish farms, collecting data of various aspects of the daily operation (e.g. sea temperature, feeding, mortality, to name a few variables). Currently, the database contains operational data from 7 different fish farms, spanning several years (the oldest from 2003), and multiple production cycles (a production cycle from deployment to slaughter lasts about 18 months). The data is imported from the commercial system AquaFarmer by Mercatus Software. The database forms the basis for the CBR system; the case base is derived from the SimFrame database.

3.1 Case Domain: Grading Operations

The current focus for the CBR system is grading operations. In a cage with varied fish size, the big fish tend to grow proportionally more than the small fish, hampering growth for the small fish. Sorting the fish into different cages based on weight leads to more optimal growth.

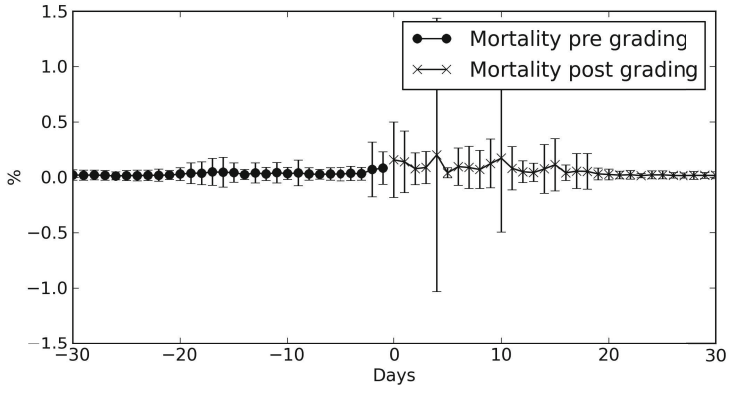


Fig. 1. Relative mortality (percent) pre- and post grading, average for all grading operations stored in the SimFrame database (74 grading operations)

In addition to facilitate growth, grading operations induce fish mortality. The grading operation involves multiple stages, several of which can be harmful for the fish. The fish need to be moved from the cages using well boats. The fish are pumped into well boats and sorted before being deployed in different cages. Calculating the relative mortality for all the grading operations present in the SimFrame database reveals an increase in fish mortality within the first weeks after a grading operation (Figure 1). The goal of the CBR system is to contribute to reduced mortality during these operations. The idea is that features of a grading operation can affect fish mortality in a way that is difficult to predict using mathematical models, but can be found by using experience based artificial intelligence. The CBR system is a continuation of a pilot project [19].

3.2 The Case Base

The current case base is generated from the SimFrame database. The cases focus mainly on environmental measurements and fish statistics, Table 1 lists all the features in the problem description part of a case. Operational data are limited to the date of the operation and the number of cages the fish were sorted to. To limit the number of features we have compressed data series such as temperature and mortality into aggregated measurement variables. These variables are averaged over 30 days before and after the grading operation. The trend is also calculated by linear regression. The continuous variables are scaled to the range $[0-1]$ to facilitate the retrieval mechanism. When comparing similarities, the distance δ between cases C and Q in feature i is calculated the following way:

$$\delta(c_i, q_i) = \begin{cases} |c_i - q_i| & \text{when continuous} \\ 0 & \text{when discrete and } c_i = q_i \\ 1 & \text{when discrete and } c_i \neq q_i \end{cases}$$

Table 1. The 16 case features

Discrete features	Continuous features
Site ID	Sorting date (day of year)
Cage ID	Number of units sorted to
Hatchery company ID	Starvation days
Species origin ID	Fish in sea date (day of year)
	Days in sea
	Number of fish
	Average fish weight
	Temperature, trend
	Temperature, average
	Pre-sorting mortality, trend
	Pre-sorting mortality, average
	Site capacity

The similarity is calculated based on the distance. Since the distance is normalized to the range $[0-1]$, the similarity between two cases in a given feature is:

$$s(c_i, q_i) = 1 - \delta(c_i, q_i)$$

The main focus for the CBR application is to lower fish mortality during grading operations. Therefore, the solution part of a case contains the difference in mortality before and after a grading operation. Before a grading operation is performed, the fish is starved. After starvation the fish need less oxygen, and empty bowels lead to less soiling of the well boat during the grading operation. The length of the starvation period depends on *degree days*: the starving period is calculated as the product of sea temperature (degrees Celsius) and time (days) and should be minimum 50 degree days. The starvation period therefore differs between grading operations. More precisely, the difference in mortality is defined as follows: let m_{pre} be the mortality prior to the starvation period (30 days in the case base), m_{starv} the mortality during the starvation period, m_{post} the mortality after the sorting operation (30 days in the case base), then θ_m is the following:

$$\theta_m = m_{post} - (m_{pre} + m_{starv})$$

θ_m forms the basis for dividing the cases into different classes, for verifying target classification rate when testing the system. θ_m also makes it possible to calculate a *true* similarity R between cases C and Q:

$$R(C, Q) = 1 - |\theta_m(C) - \theta_m(Q)| \tag{2}$$

R can be used to calculate the predicted similarity error between the retrieved case, and the case that was most similar based on θ_m. The predicted similarity error will then show how close the predicted matching case was to the actual best matching case. This reveals additional information about the retrieval mechanism: a low similarity prediction error will signify that the retrieval mechanism found a close match.

3.3 Case Retrieval

A crucial part of a CBR system is the ability to retrieve similar cases. The retrieval mechanism relies on feature weights and similarity metrics to achieve this goal. In this paper, three different ways of implementing the retrieval mechanism are examined. As mentioned in the previous section, the solution variable θ_m makes it possible to divide the cases into classes. Applying k-means clustering ($k = 3$) yields a class division shown in Figure 2.

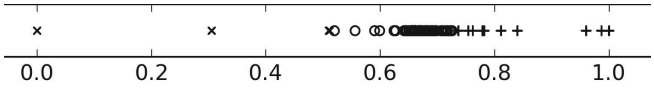


Fig. 2. k-means clustering ($k = 3$) on the scaled outcome variable θ_m. The cross, circle and plus points indicate separate cases.

k-Nearest Neighbour. The baseline index retrieval method is k-Nearest Neighbour (kNN). As mentioned in the previous section, a case is described by several features, both discrete and continuous. The distances from the query case Q to the cases in the case base are calculated, and the most similar cases (i.e. with the smallest distance) are selected. In the baseline method the weights w_i are equal in Equation (1).

Neural Networks. Another approach is to use neural networks to retrieve similar cases, by employing Echo State Networks (ESNs). ESNs are characterized by a fast training algorithm and large memory capacity [9]. ESNs are originally developed for time series prediction, by harnessing a large hidden layer to learn dynamics over time. By letting the hidden layer stabilize before collecting the hidden states, the ESN can be used as classifiers as well [4]. This recent development within neural network research allows applications that require static classifiers take advantage of the fast training algorithm inherent in the ESN architecture, avoiding the computationally expensive backpropagation algorithm that is common for classic neural network approaches.

In order to format the input signals to the neural network, discrete values are enumerated and divided into corresponding input signals. The target vector for the ESN are neurons that code for each case; if there are N cases in the case base, there will be N output neurons.

The rationale behind using neural networks for the retrieval mechanism is to exploit the idea that similar cases have similarly valued features. Instead of having to define the similarity function, the neural network can find the closest match in the feature vector space.

Linear Programming for Optimal Feature Weight Setting. In the baseline method described above, all case weights are equal. An obvious improvement is to employ feature weights. Finding the optimal feature weights can be done

using linear programming (LP) [20]. Briefly explained, an LP problem consists of minimizing an objective function $\mathbf{cx}$ subject to the constraints $\mathbf{Ax} = \mathbf{b}$. Finding the optimal weights can then be formulated as an LP problem by introducing margin variables L_{cq} and G_{cq}, making it possible to formulate the following constraint:

$$\sum_{i=1}^{n} s(c_i, q_i)w_i + L_{cq} - G_{cq} = R(C, Q) \tag{3}$$

L_{cq} represents the part when the predicted similarity is *less* than the real similarity, and G_{cq} represents the part where the predicted similarity is *greater* than the real similarity. The objective function will then be:

$$\text{minimize} \sum_{q=1}^{n} \sum_{c=q+1}^{n} (L_{cq} + G_{cq}) \tag{4}$$

The LP problem can now be solved using the *simplex algorithm* [7], which runs in polynomial time in practice (although exponential time in worst case scenarios) [6]. The optimal solution to the problem finds the feature weights. For more details of this approach, see [20].

4 Results

In order to evaluate the different retrieval mechanisms, *leave one out cross validation* (LOOCV) is employed. The LOOCV approach is as follows: one case from the case base is extracted, and then queried on the remaining case base. The most similar case is verified to see if the retrieved case is the same class as defined by k-means clustering. This process is repeated for each case in the database; after this process the system's classification ability is calculated. The case base is relatively small, only 74 cases. This small number favors LOOCV a train/test partitioning approach of the dataset. However, the case base is continually growing, so in the future the system will employ train/test evaluations as well.

For the ESN method, LOOCV requires the retraining of the network for each query. When querying the case base, the query case must be extracted from the case base, and the ESN trained on the remaining cases. This is due to the nature of the ESN approach: the output layer of the network is trained to predict the case index in the case base, and suppress all other outputs. If the query case is part of the trained dataset, it is trained to output a 0 in all other case indices. In order for the ESN approach to find the closest match in the input feature space, the query case cannot be part of the training data. However, this is not a problem due to the exceptionally fast training algorithm of the ESN architecture: the output layer is trained by simply finding the pseudoinverse matrix, with the additional cost of stabilizing the hidden layer (which in practice amounts to 10-15 steps for each input vector for the current experiment). Another advantage of using this approach is the simplicity, which comes at the cost of spending some

computational time (i.e. retraining the ESN, which still is a lot more efficient than traditional backpropagation networks). The ESN has 50 nodes in the hidden layer, and a spectral radius $\alpha = 0.55$.

In addition to finding the classification rate of the CBR system, the similarity prediction error is calculated. This yields the error (in percent) between the similarity of the case that is predicted to be most similar, and the case that *is* the most similar (as defined by equation (2)). The results of running the three different retrieval mechanisms can be seen in Table 2.

Table 2. Results from evaluating the case base using LOOCV

	Target classification	Similarity prediction error
kNN	75.6%	10.5%
ESN	75.1%	8.9%
LP	82.4%	1.7%

5 Discussion and Future Work

The classification rates using LOOCV of the CBR retrieval mechanisms (see Table 2) reveal that the LP approach is the best (82.4%), with the kNN and ESN doing slightly worse (both ∼75%). The similarity prediction error is a supplement to the classification rate of the retrieval mechanisms. The similarity prediction error is very low for the LP method, only 1.7%. This signifies that there is a very small difference between the actual most similar case, and the retrieved case (as defined by equation (2)). However, the discrepancy between very low similarity prediction error and less than perfect target classification also indicates that there are a lot of similar cases. This indication is supported by closer examining the right-hand side constraints in the formulation of the LP method. Recall that the LP approach consist of solving $\mathbf{Ax} = \mathbf{b}$, where the $\mathbf{b}$ vector represents the real similarity between cases Q and C. The average value of $\mathbf{b}$ was found to be 0.896 (scale $[0 - 1]$, 1 is most similar), i.e. on average most cases were very similar to each other. Further examining the feature weights reveals that 7 out of 16 were set to 0. These were Site ID, Cage ID, Hatchery company ID, Species origin ID, Days in sea, Number of fish and Average fish weight (see Table 1 for a list of all the features). These features did not contribute to finding the most similar case during case retrieval. This indicates that maybe the selected features for a case should be redesigned. On the other hand, using such an automated approach makes it possible to add features and let the system weed out the ones that are not needed. In such a parameter rich domain, it is often difficult to know beforehand which features are the most important.

Closer examination of the classes found by k-means clustering reveals that 81.1% of the cases belong to one class, while the remaining classes contain 12.1% and 6.8% of the cases, respectively (these numbers vary slightly when performing k-means clustering). By doing a hypothesis test using the multinomial distribution (since the samples are in different categories, using the distribution listed

above), it becomes clear that the results are not statistically significant. Nevertheless, the results clearly shows the positive effect of the LP approach to automated assignment of feature weights.

A way to further enhance the retrieval mechanism could be to use a different $R(C, Q)$, that would spread out the cases more. However, it should also be considered that the size of the case base is fairly small, and that most sorting operations are successful. Three other possible directions can be taken: 1) Design prototype cases that represents a synthesis of successful cases, and remove the other cases from the database. 2) Focus only on *outliers*, i.e. extreme cases that are very different from the others. 3) A combination of the two.

An inherent issue with the current implementation is that the cases are data-driven - the classification of cases could preferably be chosen by domain experts. We are currently looking into this issue, as it would be of great interest to see if the resulting classes are different than those from the clustering algorithm. As an aid to this situation, we have developed a registration application to gather more detailed data related to grading operations, that are not present in the automated sensor logging systems. This will further enrich the case base, and hopefully better separate cases from each other.

Acknowledgements. The research presented in this paper is part of an ongoing project called Simulation and Optimization Framework (SimFrame), which in turn is part of the CREATE (Centre for Research-based Innovation in Aquaculture Technology)[1] programme, hosted by SINTEF Fisheries and Aquaculture[2]. CREATE involves several industry partners and research institutions that influences the direction of the research conducted in SimFrame. The main partners are AKVA Group, Egersund Net AS, Erling Haug AS, NOFIMA Marin, the Norwegian University of Science and Technology, Institute of Marine Research and SINTEF ICT. The top three Norwegian salmon farming companies Marine Harvest, Lerøy Seafood Group and SalMar are new partners from 2011. The authors would also like to thank Gunnar Senneset, Hans Bjelland and Brad Schofield for insights and criticisms that helped improve the paper.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications 7(1), 39–59 (1994)
2. Aamodt, A.: Cbr for advice giving in a data-intensive environment. In: Proceeding of the 2008 Conference on Tenth Scandinavian Conference on Artificial Intelligence: SCAI 2008, pp. 201–205. IOS Press (2008)
3. Aha, D.W., Marling, C.: Special issue on Case-Based Reasoning. Knowledge Engineering Review 20 (2005)
4. Alexandre, L.A., Embrechts, M.J.: Reservoir Size, Spectral Radius and Connectivity in Static Classification Problems. In: Alippi, C., Polycarpou, M., Panayiotou, C., Ellinas, G. (eds.) ICANN 2009 Part I. LNCS, vol. 5768, pp. 1015–1024. Springer, Heidelberg (2009)

[1] `www.sintef.no/Projectweb/CREATE`
[2] `www.sintef.no/Home/Fisheries-and-Aquaculture`

5. Arshadi, N., Jurisica, I.: Data mining for case-based reasoning in high-dimensional biological domains. Transactions on Knowledge and Data Engineering 17(8), 1127–1137 (2005)
6. Cormen, T.H.: Introduction to algorithms. The MIT press (2001)
7. Dantzig, G.B.: Linear programming and extensions. Princeton University Press (1963)
8. Fiskeridirektoratet. Key figures from aquaculture industry (2010) ISBN 82-91065-15-2
9. Jaeger, H., Haas, H.: Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. Science 304(5667), 78–80 (2004)
10. John, B., Shree, N., Doug, E.: Development of decision support tools for aquaculture: the pond experience. Aquaculture Engineering 23, 103–119 (2000)
11. Li, D., Fu, Z., Duan, Y.: Fish-expert: a web-based expert system for fish disease diagnosis. Expert Systems with Application 23, 311–320 (2002)
12. Liao, S.H.: Expert system methodologies and applications - a decade review from 1995 to 2004. Expert Systems with Application 28, 93–103 (2005)
13. Liu, D.-R., Ke, C.-K.: Knowledge support for problem-solving in a production process: A hybrid of knowledge discovery and case-based reasoning. Expert Systems with Application 33, 147–161 (2007)
14. Metaxiotis, K.S., Askounis, D., Psarras, J.: Expert systems in production planning and scheduling: A state-of-the-art survey. Journal of Intelligent Manufacturing 13, 253–260 (2002)
15. Raphael, B., Domer, B., Saitta, S., Smith, I.F.C.: Incremental development of cbr strategies for computing project cost probabilities. Advanced Engineering Informatics 21, 311–321 (2007)
16. Schulstad, G.: Design of a computerized decision support system for hatchery production management. Aquaculture Engineering 16, 7–25 (1997)
17. Shimin, D., Shen, H., Liu, H.: Research on case-based reasoning combined with rule-based reasoning for emergency. In: IEEE Interntational Conference on Service Operations and Logisitcs, and Informatics (2007)
18. Shokouhi, S.V., Aamodt, A., Skalle, P., Sørmo, F.: Determining Root Causes of Drilling Problems by Combining Cases and General Knowledge. In: McGinty, L., Wilson, D.C. (eds.) ICCBR 2009. LNCS, vol. 5650, pp. 509–523. Springer, Heidelberg (2009)
19. Tidemann, A., Bjørnson, F.O., Aamodt, A.: Case-based reasoning in a system architecture for intelligent fish farming. In: Eleventh Scandinavian Conference on Artificial Intelligence - SCAI 2011. Frontiers in Artificial Intelligence and Applications, vol. 227, pp. 122–131. IOS Press (2011)
20. Zhang, L., Coenen, F., Leng, P.: Formalising optimal feature weight setting in case based diagnosis as linear programming problems. Knowledge-Based Systems 15(7), 391–398 (2002)

CPMD: A Matlab Toolbox for Change Point and Constrained Motif Discovery

Yasser Mohammad[1], Yoshimasa Ohmoto[2], and Toyoaki Nishida[2]

[1] Assiut University, Egypt
yasserm@aun.edu.eg
[2] Kyoto University, Japan
nishida@i.kyoto-u.ac.jp

Abstract. Change Point Discovery (CPD) and Constrained Motif Discovery (CMD) are two essential problems in data mining with applications in many fields including robotics, economics, neuroscience and other fields. In this paper, we show that these two problems are related and report the development of a MATLAB Toolbox (CPMD) that encapsulates several useful algorithms including new variants to solve these two related problems. The Toolbox is then used to study the effect of distance function choice in CPD.

1 Introduction

Change point discovery (CPD) is an important knowledge discovery problem. The goal of CPD to find the locations at which the underlying generating dynamics of a signal (timeseries) changes. This does not – in many cases – correspond to an easily discernable change in the signal itself.

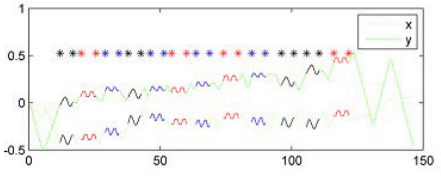
A related problem is motif discovery which can be informally defined as efficiently finding unknown recurring patterns in long time series with no prior knowledge about their locations. The research in this problem have led to many techniques including the PROJECTIONS algorithm [3], PERUSE [22], Gemoda [7] among many others (e.g. Lin et al.[9],Minnen et al. [10], and Tang et al. [24]). With the exception of Gemoda which is quadratic in time and space complexities, these algorithms aim to achieve sub-quadratic time complexity by first looking for candidate motif *stems* using some heuristic method and then doing an exhaustive motif detection instead of motif discovery which is linear in time.

Constrained Motif Discovery [14] relates the aforementioned two problems. In CMD, the goal is to find unknown recurrent patterns but utilizing a set of constraints on motif locations. One way to introduce these constraints when no prior knowledge is available is to use a CPD algorithm and then search for the motifs around the discovered change points. In this sense a CMD algorithm can be used in conjunction with ta CPD algorithm to solve the general motif discovery problem. CMD has applications in activity detection [12], gesture recognition [20], Human-Robot Interaction [13], among other fields.

Other than the aforementioned relation between CPD and CMD, the two problems have a deeper connection. The essence of both of them is a form of

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 114–123, 2012.

heuristic dissimilarity assessment between different subsequences of the time series. For this reason, implementations of these algorithms can benefit from each other as evident from the proposed toolbox. In this paper, we focus on the development of a MATLAB/Octave toolbox for solving these problems. The toolbox is freely available from the authors' website [25].



(a) Input with highlighted output

```
% Assume that the data is in two row-vectors x and y

t=tspca([x;y]); % project the data into 1-D time series
[locations,means,stats]=dgr(t,100,cpd(t,50));

% Now use the found motifs
% locations: the boundaries of motif occurrences
% means    : the mean of each motif
% stats    : optional statistics about the motifs learned
```

(b) Using CPMD to recover the motifs

Fig. 1. Example signal from the evaluation experiment and the execution time of different tested routines

The proposed toolbox can be used – among other things – to recover recurrent patterns in real-world situations. For example, consider a robot navigating its environment while performing one of three predefined tasks as shown in Fig. 1-a. Using the proposed toolbox, it is possible to segment these time series and find the three patterns with all their occurrences in linear time and with a a couple of code lines (Fig. 1-b). The recovered motifs can then be used to train another robot (learning by imitation) or to detect unexpected behaviors (fault detection) or show the emergence of complex behaviors. Section 6 reports briefly some real world applications that utilized the proposed toolbox.

The rest of this paper is organized as follows: Section 2 presents the software architecture and design decisions taken in this toolbox. Section 3 introduces the Change Point Discovery Problem and describes the routines implemented to solve it in the toolbox. Section 4 does the same for the Constrained Motif Discovery Problem. Section 5 introduces some of the essential supporting routines that simplify the use of the toolbox. Section 6 provides some example use cases of the toolbox to evaluate the effect of the distance function in CPD as well as some real world applications of the toolbox. The paper is then concluded.

2 CPMD Design

CPD and CMD are related problems as shown earlier in this paper and this is why we implemented both of them in a common toolbox. Fig. 2 shows an overview of the most important components of the proposed toolbox called Change Point and constrained Motif Discovery (CPMD). CPD and CMD components implement basic algorithms for solving CPD and CMD problems as well as routines for quantitative evaluation of solution's quality given ground-truth data. Supporting routines are used to extend the application of these algorithms

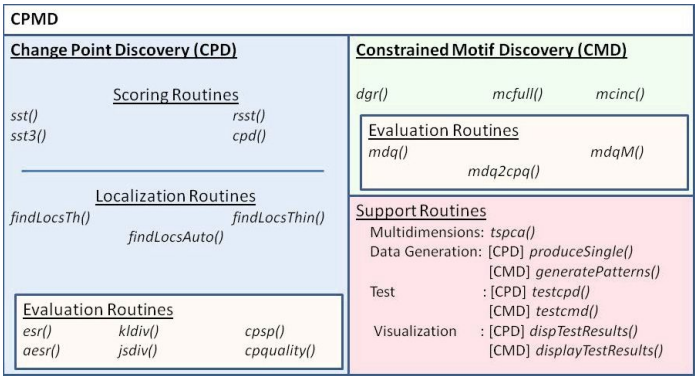


Fig. 2. The CPMD toolbox

to multidimensional time-series as well as providing test and visualization as well as synthetic data generation routines.

An important design decision in this toolbox was that related algorithms were implemented using a single general routine. For example, a single routine can be used to run 48 different related algorithm variants for solving CPD. The rationale for this decision, is that this common implementation reveals the relation between all of these algorithms even on the expense of slight performance degradation. It is important to notice that we did not just combine together any algorithms for solving the same problem but related ones that follow similar logic and algorithmic structure. This means that any two of these algorithm variants share enough steps to qualify as variants of a generalized algorithm.

Another design decision was to use an elaborate optional parameter set for each one of the key routines implemented. The routines provide default values for these parameters that are selected based on either theoretical or experimental evaluation to balance execution time and accuracy (with more focus on accuracy). Nevertheless, making these parameters available to the end user of the routines, allows advanced users to adapt the algorithms to the needs of their datasets without needing to modify the code.

Another important feature of the proposed toolbox is the availability of objective evaluation routines for comparing different solutions to CPD and CMD [18]. Given that no one of the implemented algorithm variants is totally superior to others in all cases, it is important to compare several algorithms before deciding to use one of them in a specific application. As the authors showed in [18], comparing CPD results is not a trivial task and the same is true for CMD. For this reason, the toolbox provide comparison routines for both problems.

3 CPD Routines

One of the first CPD algorithms to be proposed was CUMSUM [23] which was only able to discovery a change in the mean of a random process given the value

of the mean before the change and under the assumption of normal distribution of samples before and after the change. The research in this area have resulted in many more advanced techniques including adaptive CUMSUM [1], wavelet analysis [8], inflection point search [5], autoregressive modeling [4], Discrete Cosine Transform, and Singular Spectrum Analysis (SSA) [21][6], [14]. One of the most promising approaches to CPD is Singular Spectrum Analysis [21]. Algorithms based on this technique require no ad-hoc tuning for every time series, and assume a very general generation process.

CPD algorithms usually proceed in two steps: firstly, a score is assigned to every time step signifying the possibility of change at this point (it is not always a probability because it is not always normalized). We call this step *scoring*. Secondly, the scores are analyzed to localize change points. We call this step *localization*. In many cases these two steps are separate and one can change one of them without affecting the other generating a new algorithm variation. In continuous inspection applications, the focus is usually put onto the localization step because accurate and fast localization is of utmost important [2]. In some other applications (e.g. physiological signal analysis), the scoring step is the important one and may be the localization step can be removed all-together [16]. Given this, it is important to be able to evaluate the performance of both the scoring and localization steps separately and for this reason.

The essence of all CPD scoring algorithms is to find for every point $x(i)$ the difference between a representation of the dynamics of the few points before it (i.e. $x(i-p) : x(i)$) and the few points after it (i.e. $x(i+g) : x(i+f)$) where p, g and f are all integers and; p and f are greater than zero. This difference is the estimate for $c(i)$. The toolbox implements a general *cpd*() function that can be used with user-defined dissimilarity functions to implement most forms of CPD algorithms. The toolbox though implements a specific set of Singular Spectrum Analysis (SSA) algorithms that can be used directly by the user.

In SSA based algorithms, the dynamics of the points before and after the current point are represented using the Hankel matrix generated using the corresponding subsequences. Singular Value Decomposition (SVD) is then used to find the singular values and vectors of the Hankel Matrix and analysis of these singular vectors is the basis for assigning CP scores to different time-steps.

There are four routines that implement the scoring step: *sst()*, *sst3()*, *rsst()* and the most general *cpd()*. The routine named *sst*() implements the algorithm proposed by Ide and Inoue in [6] with default parameter settings based on the analysis in their paper. The routine *rsst*() implements the similar Robust Singular Spectrum Transform proposed by Mohammad and Nishida in [15]. The routine *sst*3() implements the algorithm proposed by Moskvina and Zhigljavsky [21] .

The final – and most important – CPD scoring routine in the toolbox is called *cpd*(). This routine has 36 optional parameters that can be used to control nearly every aspect of the algorithm. Special values of these algorithms generate identical scores as *sst*(), *sst*3(), and *rsst*(). In that sense, this routine is all what the user needs to find the change scores given the inputs. The routine has also one

non-optional parameter (other than the input time-series) specifying the number of rows in the Hankel matrices to be used. The smaller this number is, the more detailed is the result and the the coarser the change scores. The most important of the optional parameters is the distance function to be used. Any distance function can be used (which means that the routine can be used to simulate algorithms that are not SSA based). There are four predefined functions implemented in the toolbox: $distAvgEigDist()$ implements the distance function used in SST generalized to one or more Eigen vectors, $distWeightedEigDist()$ implements RSST's distance function, $distAvgVHDist()$ implements Moskvina's distance function and $distAngBetweenSubspaces()$ that simply finds the angles between the two subspaces representing the future and the past. In section 6, we will analyze the effect of this choice on the accuracy of the discovered change points. $cpd()$ also allows the user to use a common subspace representing the complete signal during score calcualtion. None of the algorithms presented in literature used this feature but the evaluations we reported in [18] show that it can provide superior results if combined with normalization for most distance functions.

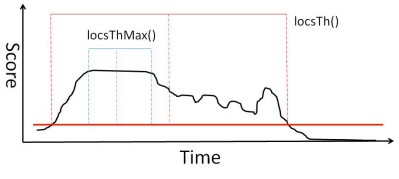


Fig. 3. The possible localization positions using the $findLocsTh()$ function

The localization step is less studied in the literature of SSA-based CPD. In most cases a change is announced if the score was over some predetermined threshold [6][15]. Moskvina et al. [21] provide a technique for automatically evaluating that works only with their algorithm (implemented in $sst3()$). The toolbox provides several algorithms that can be used for this step. The algorithms provided in the toolbox are divided into two groups $findLocsTh*()$ and $findLocsAuto*()$. The first group require the specification of a threshold and are illustrated in Fig. 3 while the second group calculates an appropriate threshold automatically.

There are two variations of threshold-based localization: The function $findLocsTh()$ localizes the change at any point between the start and ending points at which the change score exceeds the threshold value controlled by a linear parameter (the Figure shows the three states when this parameter is zero, o.5 and 1). It can also localize the change using the points at which the score attained its maximum value after exceeding the threshold and before going under it again (as illustrated in Fig. 3). Again, this can be controlled by a linear parameter.

Another threshold-based localization mechanism implemented in the toolbox using the function $findLocsThin()$, starts by thinning the change score signal

and then keeps the local maxima as localized change points. This method leads usually to more false positives but reduces the false negative rate in our experiments.

Depending on the application, one or the other of these two localization mechanisms may provide better results.

4 CMD Routines

Unconstrained motif discovery is a known problem in data-mining and several algorithms have tried to solve it [3],[22],[7] [9], [10] [24]. With the exception of Gemoda [7] which is quadratic in time and space complexities, these algorithms aim to achieve sub-quadratic time complexity by first looking for candidate motif *stems* using some heuristic method and then doing exhaustive motif detection instead of motif discovery which is linear in time. CMD algorithms find recurrent patterns in single-dimension time-series subject to user defined constraints on the possible locations of motifs.

The toolbox implements CMD using three routines called $mcfull()$, $mcinc()$ and $dgr()$. The routines $mcfull()$ and $mcinc()$ implement the algorithms proposed in [14].

The most general CMD function in the toolbox is $dgr()$. The name stands for Distance-Graph Relaxation. DGR is a modified and generalized algorithm based on the one proposed in [13] for discovering gestures in a Human-Robot Interaction context. The algorithm works in two stages: in the first stage a set of short stem motifs by joining short subsequences of small distance around change points. This leads to a set of motif candidate. Each candidate is extended as much as possible without increasing the variance of the distances between individual points of its occurrences to the level of statistical significance (controlled by a significance level optional parameter) using a t-test for the comparisons. After extension, the motifs are passed to the finalization stage. During this stage, overlapping motifs are analyzed to either separate them or combine them. Similar stems are combined and an optional full time-series scan is done to discover all occurrences of the motifs in the time series.

The routine receives the input time series (which may be multidimensional), and the constraint in the same format as the output of any CPD or localization function of the toolbox. The only obligatory parameter is an expected motif length range. This range is used internally to select various optional parameters but is not limiting the outputs of the routine. All discovered motifs will be at least the same length as the minimum length if supplied but the routine also finds motifs of any length larger than that (even longer than the supplied upper limit). This is one of the major advantages of this routine over available motif discovery algorithms as most of them require the specification of a specific motif length [11]. The routine has 51 optional parameters that can be used to control every aspect of it. As with *cpd*() a default value for each of these parameters is already available that were selected mostly after experimental comparison of different options.

5 Supporting Routines

CPD and CMD algorithms implemented in the toolbox were designed to handle single dimensional data streams. The toolbox provides a routine for projecting multidimensional time series using PCA (called *tspca*()) that is used automatically by *dgr*() to deal with multiple dimensions. The toolbox also provides several visualization and performance profiling routines.

The toolbox provides several approaches to CPD and CMD that could be extended through the optional parameters of *cpd*() and *dgr*(). Different variations will have different accuracy and performance in different applications. For this reason, it is essential to have a set of evaluation routines that allows the user of the toolbox to select the best variant for her application.

Comparison between change detection algorithms is usually done using one of three approaches: The first approach is using the traditional confusion matrix based statistics including the F score, Mathews Correlation Coefficient (implemented in the toolbox using the routine *cpquality*()). The second approach is using information theoretic divergences between the true change point locations and estimated locations (implemented in the toolbox using the routines *kldiv*() and *jsdiv*()). A novel approach was proposed in [18] that utilizes aspects from all of these approaches and provide more subjectively-acceptable as well as rigourously supported measure of CPD quality. This is called Equal Sampling Ratio (ESR) and is implemented via *esr*() and its aggregated version *aesr*().

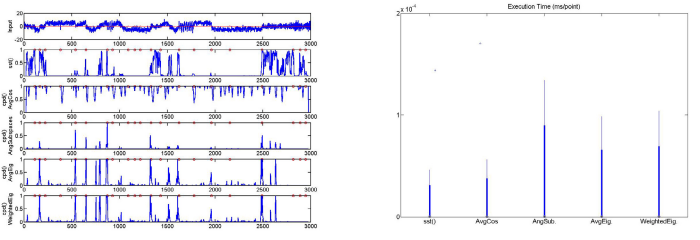
CMD evaluation is also not straight forward because of possible delays and overlaps between different motifs. The toolbox helps in that by providing two evaluation routines: *mdqM*() finds confusion matrix based statistics aggregated over all of the motifs while *mdq*() finds these statistics for each motif separately. The toolbox provides also a function *mdq2cpq*() that can be used to convert the problem from motif discovery evaluation into a CPD quality evaluation.

6 Use Case and Real-World Applications

Other than the aforementioned routines, the toolbox provides a set of simple test routines that can be modified easily to compare different variations of CPD and CMD algorithms (*testcpd*() and *testcmd*() in order). There are also generation routines that can be used to generate synthetic data with controllable shape, motifs, change locations, noise parameters, etc for fast evaluation and comparison with other approaches. In this section, we use these test routines to study the effect of the distance measure on the accuracy of *cpd*().

Here we report a preliminary evaluation of the effect of distance function choice on the accuracy of CPD results using only routines from the toolbox. The dataset we use consists of 100 time series of length 3000 each that are generated using one of seven different generating processes with random parameters (corresponding to periodic sins and cosines, linear and sawtooth signals, and an all-zeros signal) as well as a random auto-regressive model generator that can generate stable AR models with any number of parameters ranging from 1 to

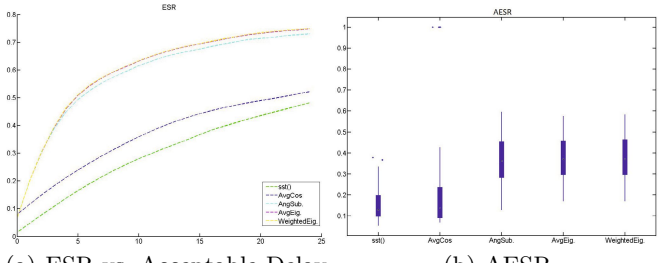
20 (using the toolbox function *produceSingle*()). At every timestep and with a probability of 1% the generating process may change. This results on a rich array of time series. From each one of these 100 series, we generate 8 time series by adding a random noise component that has a zero mean and a standard deviation that varies from zero to half the range of the signal. This results of 800 time series with 2400000 points containing around 24000 change points. The code for this test can be found in the script *testDist* in the toolbox.



(a) Example output (b) Execution Time

Fig. 4. Example signal from the evaluation experiment and the execution time of different tested routines

Fig. 4-a shows an example signal and the change points discovered by *sst*(), *rsst*(), and the four variations of the default implementation of *cpd*() using the four built-in distance functions. Fig. 4-b shows the execution time of the five tested algorithms. The distance function leading to fastest execution is *distAvgVHDist*() while the slowest one is *distAngBetweenSubspaces*(). The differences shown in the figure are not statistically significant though.



(a) ESR vs. Acceptable Delay (b) AESR

Fig. 5. The ESR and AESR scores of all tested routines on the complete dataset

Fig. 5 presents the ESR scores and the acceptable-delay independent AESR scores of the five tested algorithms. From this figure, it is clear that using any

of the four distance functions other than $distAvgVHDist()$ outperformed $sst()$. A t-test shows that these differences are statistically significant.

This toolbox have been already used by our research group in several real-world applications: For example, in [16], $rsst()$ was used to measure the naturalness of human-human and human-robot interactions by analyzing psychophysiological signals including Blood Volume Pulse, Skin Conductance, and Respiration Rate of one partner to infer the naturalness of the other's behavior. RSST based statistics were shown to provide superior classification accuracy to traditional metrics achieving 95.2% accuracy in distinguishing natural and unnatural behaviors.

In [13], both $rsst()$ and $dgr()$, were used to discovery free-hand gestures in a guided navigation task, allowing the robot to learn free hand gestures and automatically building controllers for achieving accurate navigation after few trials. The approach also allowed the robot to extend learned behavior when interacting with new partners. In [19], the authors used the toolbox to build a self-initiated imitation engine that was able in a series of synthetic and real world experiments to learn various kinds of navigation behaviors based on their saliency (measured by $cpd()$) and recurrence (measured through $dgr()$).

In [17], $cpd()$ was used as the basis for a new change-causality discovery technique that was able to recover the causal structure of events in a human-robot guided navigation settings.

7 Conclusion

This paper, presented the Change Point and Constrained Motif Discovery toolbox (CPMD), which implements some of the leading algorithms in the two fields as well as new variations that were not previously published. The toolbox provides also a set of evaluation routines, test routines, and signal generation routines, that can be used to evaluate the relative merits of different approaches. The implementation of the toolbox is modular and extendible through extensive use of optional parameters that give the final user full control of all aspects of the algorithms implemented. The paper also reported the results of extensive analysis of the effect of the distance function in CPD and the effect of stem extension method in CMD. Example real-world applications are also reported.

References

1. Alippi, C., Roveri, M.: An adaptive CUSUM-based test for signal change detection. In: 2006 IEEE International Symposium on Circuits and Systems, p. 4 (2006)
2. Basseville, M., Nikiforov, I.: Detection of abrupt changes: theory and application, vol. 15. Citeseer (1993)
3. Chiu, B., Keogh, E., Lonardi, S.: Probabilistic discovery of time series motifs. In: KDD, pp. 493–498 (2003)
4. Gombay, E.: Change detection in autoregressive time series. J. Multivar. Anal. 99(3), 451–464 (2008)

5. Hirano, S., Tsumoto, S.: Mining similar temporal patterns in long time-series data and its application to medicine. In: ICDM, p. 219 (2002)
6. Ide, T., Inoue, K.: Knowledge discovery from heterogeneous dynamic systems using change-point correlations. In: Proc. SIAM Intl. Conf. Data Mining (2005)
7. Jensen, K.L., Styczynxki, M.P., Rigoutsos, I., Stephanopoulos, G.N.: A generic motif discovery algorithm for sequenctial data. BioInformatics 22(1), 21–28 (2006)
8. Kadambe, S., Boudreaux-Bartels, G.: Application of the wavelet transform for pitch detection of speech signals. IEEE Transactions on Information Theory 38(2), 917–924 (1992)
9. Lin, J., Keogh, E., Lonardi, S., Patel, P.: Finding motifs in time series. In: The 2nd Workshop on Temporal Data Mining, at the 8th ACM SIGKDD International, pp. 53–68 (2002)
10. Minnen, D., Essa, I., Isbell, C.L., Starner, T.: Detecting Subdimensional Motifs: An Efficient Algorithm for Generalized Multivariate Pattern Discovery. In: ICDM (2007)
11. Minnen, D., Starner, T., Essa, I., Isbell, C.: Improving activity discovery with automatic neighborhood estimation. In: Int. Joint Conf. on Arti. Intell. (2007)
12. Miwa, H., Itoh, K., Matsumoto, M., Zecca, M., Takanobu, H., Rocella, S., Carrozza, M., Dario, P., Takanishi, A.: Effective emotional expressions with expression humanoid robot we-4rii: integration of humanoid robot hand rch-1. In: IROS, vol. 3, pp. 2203–2208 (2004)
13. Mohammad, Y., Nishida, T.: Learning interaction protocols using augmented baysian networks applied to guided navigation. In: IROS, pp. 4119–4126 (2010)
14. Mohammad, Y., Nishida, T.: Constrained motif discovery in time series. New Generation Computing 27(4), 319–346 (2009)
15. Mohammad, Y., Nishida, T.: Robust Singular Spectrum Transform. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 123–132. Springer, Heidelberg (2009)
16. Mohammad, Y., Nishida, T.: Using physiological signals to detect natural interactive behavior. Applied Intelligence 33, 79–92 (2010)
17. Mohammad, Y., Nishida, T.: Discovering causal change relationships between processes in complex systems. In: 2011 IEEE/SICE SII (2011) (to appear)
18. Mohammad, Y., Nishida, T.: On comparing SSA-based change point discovery algorithms. In: 2011 IEEE/SICE SII (2011) (to appear)
19. Mohammad, Y., Nishida, T.: Self-initiated imitation learningdiscovering what to imitate. In: AAMAS (submitted, 2012)
20. Mohammad, Y., Nishida, T., Okada, S.: Unsupervised simultaneous learning of gestures, actions and their associations for human-robot interaction. In: IROS, pp. 2537–2544 (2009)
21. Moskvina, V., Zhigljavsky, A.: An algorithm based on singular spectrum analysis for change-point detection. Communications in Statistics, Simulation and Computation 32(4), 319–352 (2003)
22. Oates, T.: PERUSE: An unsupervised algorithm for finding recurring patterns in time series. In: ICDM, pp. 330–337 (2002)
23. Page, E.S.: Continuous inspection schemes. Biometrika 44, 100–115 (1954)
24. Tang, H., Liao, S.S.: Discovering original motifs with different lengths from time series. Know.-Based Syst. 21(7), 666–671 (2008)
25. CPMD Toolbox, `http://www.ii.ist.i.kyoto-u.ac.jp/~yasser/cpmd/cpmd.html`

Time Invariant Gesture Recognition by Modelling Body Posture Space

Binu M. Nair and Vijayan K. Asari

Computer Vision and Wide Area Surveillance Laboratory,
Electrical and Computer Engineering,
300 College Park Avenue, Kettering Labs KL-302A,
University of Dayton, Dayton, OH - 45469, USA
{nairb1,vasari1}@udayton.edu
http://visionlab.udayton.edu

Abstract. We propose a framework for recognizing actions or gestures by modelling variations of the corresponding shape postures with respect to each action class thereby removing the need for normalization for the speed of motion. The three main aspects are the shape descriptor suitable for describing its posture, the formation of a suitable posture space, and a regression mechanism to model the posture variations with respect to each action class. Histogram of gradients(HOG) is used as the shape descriptor with the variations being mapped to a reduced Eigenspace by PCA. The mapping of each action class from the HOG space to the reduced Eigen space is done using GRNN. Classification is performed by comparing the points on the Eigen space to those determined by each of the action model using Mahalanobis distance. The framework is evaluated on Weizmann action dataset and Cambridge Hand Gesture dataset providing significant and positive results.

Keywords: Histogram of gradients(HOG), Generalized Regression Neural Nets(GRNN), Human Action Modelling, Principal Component Analysis(PCA), K-Means Clustering.

1 Introduction

Human gesture recognition has been a widely researched area over the last few years due to potential applications in the field of security and surveillance. Early research on gesture recognition used the concept of space time shapes, which are concatenated silhouettes over a set of frames, to extract certain features corresponding to the variation within the spatio-temporal space. Gorelick et al. [7] modelled the variation within the space time shape using Poisson's equation and extracted space time structures which provides discriminatory features. Wang et al. recognized human activities using the derived form of the Radon transform known as the R-Transform [17,16]. A combination of a 3D distance transform along with the R-Transform is used to represent a space time shape at multiple levels and used as corresponding action features [11].

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 124–133, 2012.

Action sequences can also be represented as a collection of spatio-temporal words with each word corresponding to a certain set of space-time interest points which are detected by set of 2D spatial gaussian filter and 1D gabor temporal filters [12]. Here, Niebles et.al computes the probability distributions of the spatio-temporal words corresponding to each class of human action using a probabilistic Latent Semantic Analysis model. Another algorithm which is similar is given by Batra et.al where a dictionary of mid-level features called space time shapelets is created which characterize the local motion patterns within a space time shape thereby representing an action sequence as a histogram of these space time shapelets over the trained dictionary [2]. However, these methods are susceptible to illumination variation or require good foreground segmentation of the silhouettes. Another approach is to model the non-linear dynamics of the human action by tracking the trajectories of certain points in the body and capture features from those trajectories. Ali et al. used the concepts from Chaos Theory to reconstruct the phase space from each of the trajectories and compute the dynamic and metric invariants which are then used as action feature vectors [1]. This method will be affected by partial occlusions as some trajectories maybe missing which may affect the metrics extracted. Scovannar et al. used a 3D-SIFT to represent spatio-temporal words in a bag of words model representation of action videos [13]. Sun et al extended the above methodology which combined local descriptors based on SIFT features and hositic moment-based features [15]. The local features comprised of the 2D SIFT and 3D SIFT features computed from suitable interest points and the holistic features are the Zernike moments computed from motion energy images and motion history images. The approach taken here assumes that the scene is static as it relies on the frame differencing to get suitable interest points.

A different approach for characterizing human action sequences is to consider these sequences as multi-dimensional arrays called tensors. Kim et al. presented a new framework called Tensor Cannonical Correlation Analysis where descriptive similarity features between two video volumes are used in nearest neighbour classification scheme for recognition [8]. Lui et.al however, studied the underlying geometry of the tensor space occupied by human action sequences and performed factorization on this space to obtain product manifolds [10]. Classification is done by projecting a video or a tensor onto this space and classifying it using a geodesic distance measure. In this type of methodology, unlike in the space time approach, it shows much improved performance on datasets with large variations in illumination and scale. However, the classification is done per video sequence and not one a set of frames constituting a part of a video sequence. A 3D gradient-based shape descriptor representing the local variations was introduced by Klaser et al. [9] and is based on the 2D HOG descriptor used for human body detection [5,6]. Here, each space time shape is divided into cubes where in each cube, the histogram is computed from the spatial and temporal gradients. Chin et al performed an analysis on modelling the variation of the human silhouettes with respect to time [4]. They studied the use of different dimensionality reduction techniques such as PCA and LLE and the use of neural

networks to model the mapping. The proposed technique in this paper uses the histogram of spatial gradients in a region of interest, finds an underlying function which captures the temporal variance of these 2D shape descriptors with respect to each action and classifies a set of contiguous frames irrespective of the speed of the action or the time instant of the body posture.

2 Proposed Methodology

In this paper, we focus on three main aspects of the action recognition framework. The first is that of feature extraction where a shape descriptor is computed for the region of interest in each frame. The second is that of a computation of an appropriate reduced space which spans the shape change variations across time. The third aspect is that of suitable modelling of the mapping from the shape descriptor space to the reduced space. A suitable block diagram illustrating the framework is shown Figure 1.

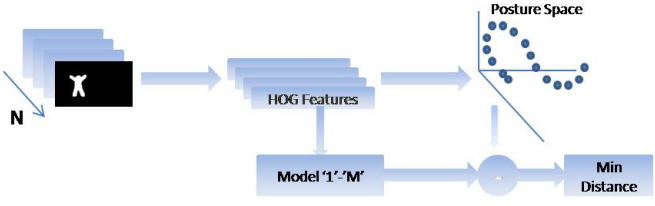
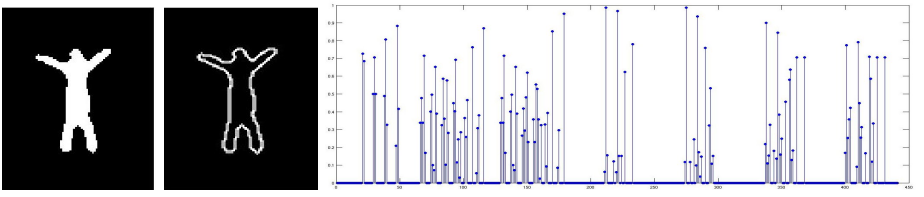


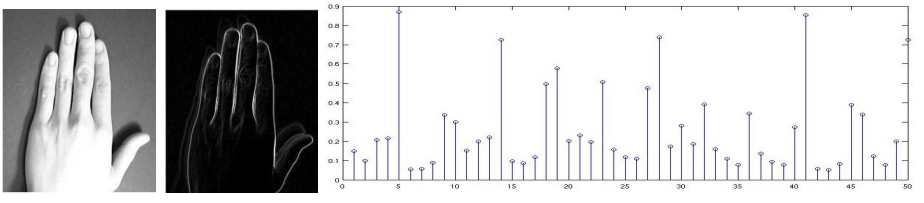
Fig. 1. Gesture Recognition Framework

Histogram of gradients is used as the shape descriptor as it provides a more local representation of the shape and it is partially invariant to illumination. To obtain a reduced dimensional space where the inter-frame variation among the HOG descriptors are large, we use the Principal Component Analysis. The inter-frame variation between the HOG descriptors is due to the change in shape of a body or hand with respect to a particular action being performed. So, we propose a modelling of these posture or shape variations with respect to each action, the underlying idea being the posture variations differ with action class. A modelling of actions in this manner removes the need for normalization with respect to time and that a slow or fast moving action of the same class will not cause any difference. Only the postures of each frame are correspondingly mapped onto a reduced space containing variations in time, thereby making the framework time-invariant. In other words, while classifying a slow action, the posture variations will occupy a small part of the manifold when compared to a fast moving action where the posture variations occupy a large section of the action manifold. Moreover, due to varying speed of the action in different individuals, some of the postures in the action sequence may not be present during the training phase. So, when these particular postures occur in a test sequence of an action, the action model can estimate where that posture lies on the reduced space. This approach gives a more accurate estimation of the



(a) Silhouette (b) Gradient (c) HOG Descriptor

Fig. 2. HOG descriptor extracted from a binary human silhouette from the Weizmann Database [7]



(a) Hand (b) Gradient (c) HOG Descriptor

Fig. 3. HOG descriptor extracted from a gray scale hand image from Cambridge Hand Gesture Database [8]

corresponding location of that particular shape on the action manifold than the approach which uses nearest neighbours to determine the corresponding reduced posture point. In this paper, we use a separate model for each action class and the modelling is done using generalized regression neural networks which is a multiple-input multiple-output network.

2.1 Shape Representation Using HOG

The histogram of gradients is computed by first taking the gradient of the image in the x and y directions and calculating the gradient magnitude and orientation at each pixel. The image is then divided into overlapping K blocks and the orientation range is divided into n bins. From each block, the gradient magnitudes of those pixels corresponding to the same range of orientation (belonging to the same bin) are added up to form a histogram. The histograms from the various blocks are normalized and concatenated to form the HOG shape descriptor. An illustration of the HOG descriptor extracted from masked human silhouette image are shown in Figure 2. It can been seen that since the binary silhouette produces a gradient where all of its points correspond to the silhouette, the HOG descriptor produces a discriminative shape representation. Moreover, due to the block operation during the computation of the HOG, this descriptor provides a more local representation of the particular posture or shape. An illustration of the HOG descriptor(first 50 elements) applied on a gray scale hand image is

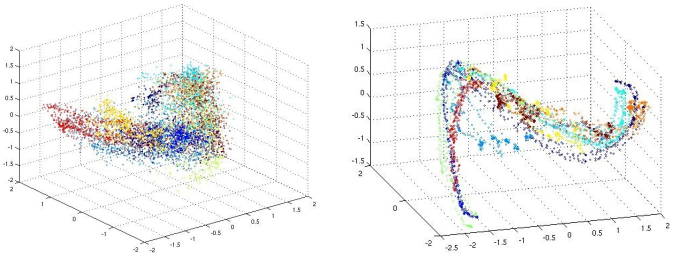
shown in Figure 3. Unlike the binary image, there is some noise in the gradient image which gets reflected onto the HOG descriptor. Since the HOG descriptors are illumination invariant, we can assume that under varying illumination conditions, the feature descriptors do not vary much.

2.2 Computation of Reduced Posture Space Using PCA

The next step in the framework is determine an appropriate space which represents the inter-frame variation of the HOG descriptors. An illustration of the reduced posture or shape space using PCA is shown for the Weizmann dataset and the Cambridge Hand dataset using three Eigenvectors in Figure 4. Each action class of the reduced posture points shown in Figure 4 are color-coded to illustrate how close the action manifolds are and the separability existing between them. We can see that there are lot of overlaps between different action manifolds in the reduced space and our aim is to use a functional mapping for each manifold to distinguish between them. We first collect the HOG descriptors from all the possible postures of the body irrespective of the action class and form a space denoting what is known as an action space denoted by S_D. We can express the action space mathematically as

$$S_D = \{\boldsymbol{h}_{k,m} : 1 \leq k \leq K(m)\, and\, 1 \leq m \leq M\} \tag{1}$$

where $K(m)$ is the number of frames taken over all the training video sequences from the action m out of M action classes and $h_{k,m}$ being the corresponding HOG descriptor of dimension $D \times 1$. The reduced action or posture space is obtained by extracting the principal components of the matrix $\mathbf{H}\mathbf{H}^T where\, \mathbf{H} = [\boldsymbol{h}_{1,1}\, \boldsymbol{h}_{1,1}\, \boldsymbol{h}_{1,1} \ldots \boldsymbol{h}_{K(M),M}]$ using PCA. This is done by finding the Eigenvectors or Eigenpostures $\boldsymbol{v}_1 \boldsymbol{v}_2 \ldots \boldsymbol{v}_d$ corresponding to the largest variances between the HOG descriptors. In this reduced space, the inter-frame variation between the extracted HOG descriptors due to the changes of shape of the body (due to the motion or the action) are maximized by selecting the appropriate number of Eigenpostures and at the same, reducing the effect of noise due to illumination



(a) Weizmann Dataset (b) Cambridge Hand Dataset

Fig. 4. Reduced Posture Space for the HOG descriptors extracted from video sequences

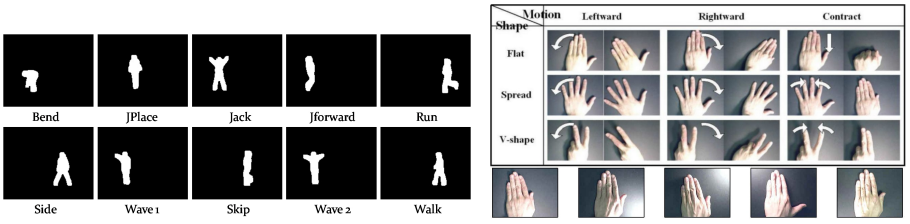
by removing those Eigenpostures having low Eigenvalues. In other words, the Eigenvectors with the highest Eigenvalues corresponds to the direction along which the variance between the HOG descriptors due to the posture or shape change is maximum and all the other Eigenvectors with lower Eigenvalues can be considered as directions which corresponds to the noise in the HOG shape descriptor.

2.3 Modelling of Mapping from HOG Space to Posture Space Using GRNN

The mapping from the HOG descriptor space ($D \times 1$) to the reduced posture or shape space ($d \times 1$) can be represented as $S_D \mapsto S_d$ where $S_d = \{\boldsymbol{p}_{k,m} : m = 1\,to\,M\}$ and $\boldsymbol{p}$ is a vector representing a point in the reduced posture space. In this framework, we aim to model the mapping from the HOG to the posture space for each action m separately using the Generalized Regression Neural Network [14,3]. This network is a one-pass learning algorithm which provides fast convergence to the optimal regression surface. It is memory intensive as it requires the storage of the training input and output vectors where each node in the first layer is associated with one training point. The network models the equation of the form $\hat{\boldsymbol{y}} = \frac{\sum_{i=1}^{N} \boldsymbol{y}_i \, radbasis(\boldsymbol{x}-\boldsymbol{x}_i)}{\sum_{i=1}^{N} radbasis(\boldsymbol{x}-\boldsymbol{x}_i)}$ where $(\boldsymbol{y}_i, \boldsymbol{x}_i)$ are the training input/output pairs, $\hat{\boldsymbol{y}}$ is the estimated point for the test input $\boldsymbol{x}$. In our algorithm, since a lot of training points are present, a lot of nodes have to be implemented for each class which is not memory efficient. To get suitable training points that marks the transitions in the posture space for a particular action class m, k-means clustering is done to get $L(m)$ clusters. So, the mapping of the HOG descriptor space to its reduced space for a particular action class m can be modelled by a general regression equation given as

$$\hat{\boldsymbol{p}} = \frac{\sum_{i=1}^{L(m)} \bar{\boldsymbol{p}}_{i,m} \exp(\frac{D_{i,m}^2}{2\sigma^2})}{\sum_{i=1}^{L(m)} \exp(\frac{D_{i,m}^2}{2\sigma^2})} \;;\; D_{i,m} = (\boldsymbol{h} - \bar{\boldsymbol{h}}_{i,m})^T (\boldsymbol{h} - \bar{\boldsymbol{h}}_{i,m}) \tag{2}$$

where $(\bar{\boldsymbol{p}}_{i,m}, \bar{\boldsymbol{h}}_{i,m})$ are the ith cluster centres in the HOG descriptor space and the posture space. Selection of the standard deviation σ for each action class is taken as the median Euclidean distance between the corresponding action's cluster centres. The action class is determined by first projecting the consecutive set of R frames onto to the Eigenpostures. These projections of the frames given by $\boldsymbol{p}_r : 1 \leq r \leq R$ is compared with the estimated projections $\hat{\boldsymbol{p}}(m)_r$ of the corresponding frames estimated by each of the GRNN action model using the Mahalanobis distance. The action model which gives the closest estimates of the projections is selected as the action class.



(a) Weizmann Dataset [7] (b) Cambridge Hand Dataset [8]

Fig. 5. Datasets for Testing

3 Experiments and Results

The algorithm presented in this paper has been evaluated on two datasets,the Weizmann Human Action [7] and the Cambridge Hand Gestures [8]. The histogram of gradients feature descriptor has been extracted by dividing the detection region into 7×7 overlapping cells. From each cell, a histogram of gradient is computed with 9 orientation bins which are normalized by taking the $L2-$norm, and the normalized histograms are concatenated to form the feature vector of size 441×1.

3.1 Weizmann Action Dataset

This action dataset consists of 10 actions classes, each of them contain $9-10$ samples performed by different people. The background in these video sequences are static with uniform lighting at low resolution and so, silhouettes of the person can be extracted by a simple background segmentation. HOG features, computed for these silhouettes represent the shape or the postures of the silhouette at one particular instant. During the training phase, all the frames of every training sequence of each class are taken together to get the HOG feature set for each action class. The test sequence is split up into overlapping windows (partial sequences) of size N with an overlap of $N-1$. The HOG features of each frame of the window is compared with the estimated features from each action class model corresponding to this particular frame using Mahalanobis distance, and the appropriate distance from each class is computed by taking the $L2-$ norm of the distances for each frame. The action model which gives the minimum final distance measure to the testing partial sequence is determined to be its action class. Table 1 gives the results for the framework with GRNN having 10 clusters with a window size of 20 frames. The testing is done by using leave-10 out procedure where 10 sequences, each one corresponding to a particular action class are considered as the testing set while the remaining sequences are taken as the training set. The variation of the overall accuracy for different window sizes of $10, 12, 15, 18, 20, 23, 25, 28, 30$ of the test partial sequences are shown in Figure 6.

Table 1. Confusion Matrix for Weizmann Actions: a1 - bend; a2 - jplace ; a3 - jack ; a4 - jforward ; a5 - run ; a6 - side ; a7 - wave1 ; a8 - skip ; a9 - wave2 ; a10 - walk

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10
a1	99									
a2		100								
a3			100							
a4				99						
a5					98			1		
a6			9			90				
a7			2				97			
a8					2			97		
a9			3						96	
a10						4				95

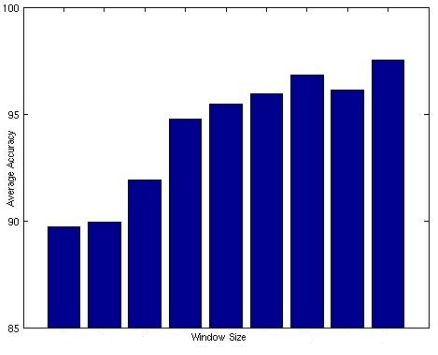


Fig. 6. Average Accuracy computed for the action classes for window size 10, 12, 15, 18, 20, 23, 25, 28, 30

3.2 Cambridge Hand Gesture Dataset

The dataset contains 3 main actions classes showing different postures of the hand, flat, spread out and V-shape. Each of the main classes has three other sub-classes which differs in the direction of movement. In total, we have 9 different action classes which differs in the posture of the hand as well as its direction of motion. The main challenge is to differentiate between different motion and shape at different illumination conditions. The dataset is shown in Figure 5(b). There are 5 sets, each containing different illuminations of all the action classes with class having 20 sequences. From each of the video sequences, we applied skin segmentation to get a rough region of interest, and extracted the HOG based shape descriptor from the gray scale detection region. Unlike the descriptors extracted from silhouettes in the Weizmann dataset, these descriptors contain noise variations due to different illumination conditions. The testing strategy we used is the same as that of the Weizmann with leave-9 out video sequences

Table 2. Confusion Matrix and Overall Accuracy for Cambridge Hand Gesture Dataset

(a) Confusion Matrix

	a1	a2	a3	a4	a5	a6	a7	a8	a9
a1	94.0		1			5			
a2		91.0	6			3			
a3	2	1	95.0			2			
a4				91.0	1	8			
a5				5	85.0	10			
a6					1	99.0			
a7							83.0	3	14
a8								86.0	14
a9						1	13	9	77.0

(b) Overall Accuracies for each Set

	Set 1	Set 2	Set 3	Set 4	Set 5
Acc	96.11	73.33	70.00	86.67	87.72

where each test sequence corresponds to an action class. The confusion matrix for the action classes obtained from the framework with 4 clusters is given in Table 2(a). We can see that if all the illumination conditions are trained into the system, the overall accuracy obtained with the framework is high. Using the same testing strategy, we tested the system for overall accuracy for each set and this is given in Table 2(b). For set1, the overall accuracy is high as the non-uniform lighting does not affect the feature vectors and noise is diminished by the partial illumination variant property of the HOG descriptor. For sets 4 and 5, it shows moderate accuracies while sets 2 and 3 give an average overall accuracy.

4 Conclusions and Future Work

In this paper, we presented a frame work for recognizing actions from partial video sequences which is invariant to the speed of the action being performed. We illustrated this approach using the Histogram of Gradients shape descriptor and computed the mapping from the HOG space to the reduced dimensional posture space using Principal Component Analysis. The mapping from the HOG space to the reduced posture space for each action class is learned separately using Generalized Regression neural network. Classification is done by projecting the HOG descriptors of the partial sequence onto the posture space and comparing the reduced dimensional representation with that of the estimated posture from the GRNN action models using Mahalanobis distance. The results shows the accuracy of the framework as illustrated on the Weizmann database. However, when using the gray scale images to compute the HOG, severe illumination conditions can affect the framework as illustrated by the Hand Gesture database results. In future, our plan is to extract a shape descriptor which represents a shape from a set of corner points where relationships between them are determined in the spatial and temporal scale. Other regression and classification schemes will also be investigated in this framework.

References

1. Ali, S., Basharat, A., Shah, M.: Chaotic invariants for human action recognition. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8 (October 2007)
2. Batra, D., Chen, T., Sukthankar, R.: Space-time shapelets for action recognition. In: IEEE Workshop on Motion and video Computing, WMVC 2008, pp. 1–6 (January 2008)
3. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics), 1st edn. Springer (2006), corr. 2nd printing edn. (October 2007)
4. Chin, T.J., Wang, L., Schindler, K., Suter, D.: Extrapolating learned manifolds for human activity recognition. In: IEEE International Conference on Image Processing, ICIP 2007, vol. 1, pp. 381–384 (October 2007)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893 (June 2005)
6. Dalal, N., Triggs, B., Schmid, C.: Human Detection Using Oriented Histograms of Flow and Appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part II. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006)
7. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. Transactions on Pattern Analysis and Machine Intelligence 29(12), 2247–2253 (2007)
8. Kim, T.K., Wong, S.F., Cipolla, R.: Tensor canonical correlation analysis for action classification. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8 (June 2007)
9. Klaser, A., Marszalek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: Proceedings of the British Machine Vision Conference (BMVC 2008), pp. 995–1004 (September 2008)
10. Lui, Y.M., Beveridge, J., Kirby, M.: Action classification on product manifolds. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 833–839 (June 2010)
11. Nair, B., Asari, V.: Action recognition based on multi-level representation of 3d shape. In: Proceedings of the International Conference on Computer Vision Theory and Applications, pp. 378–386 (March 2010)
12. Niebles, J., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. In: British Machine Vision Conference, BMVC 2006 (2006)
13. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional sift descriptor and its application to action recognition. In: Proceedings of the International Conference on Multimedia (MultiMedia 2007), pp. 357–360 (September 2007)
14. Specht, D.: A general regression neural network. IEEE Transactions on Neural Networks 2(6), 568–576 (1991)
15. Sun, X., Chen, M., Hauptmann, A.: Action recognition via local descriptors and holistic features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009, pp. 58–65 (June 2009)
16. Tabbone, S., Wendling, L., Salmon, J.: A new shape descriptor defined on the radon transform. In: Computer Vision and Understanding, vol. 102, pp. 42–51 (2006)
17. Wang, Y., Huang, K., Tan, T.: Human activity recognition based on r transform. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8 (June 2007)

Go beyond the SCOPE: A Temporal Situation Calculus-Based Software Tool for Time Petri Nets

Xing Tan

Center for Technology in Government
University at Albany/SUNY
xtan2@albany.edu

Abstract. This paper presents a straightforward and mathematically rigorous software tool for Time Petri Nets (TPN). More precisely, we first build up a formal process ontology of TPNs by extending SCOPE (Situation Calculus Ontology for PEtri nets) with temporal constraints. We then demonstrate that, with this temporal SCOPE, sequential, iterative, or nondeterministic composite sequences of transition firings in TPNs can be further axiomatized through macro-expanding the basic action of *transition firing*. Finally, it is shown that theorem-proving, i.e., computing the consequences of such an ontology, can be carried out efficiently through logic programming in Prolog. Potential future applications of our approach are also discussed.

Keywords: Situation Calculus, GOLOG, Time Petri Nets.

1 Introduction

Petri nets have been used broadly as a generic modeling tool to describing scale-up, complicated systems. Time Petri Pets (TPN) are an important extension to the classical Petri nets. Transitions in a TPN are associated with time intervals. Design, analysis, and simulation of Petri nets-based systems often depend on the support of automated software tools. Because of the complexity of TPNs, one central requirement of such tools is that they should be able to correctly capture the behaviors of marking transitions, i.e., the change on the number of tokens at places upon actions of transitions firing.

This paper provides a software tool for TPN, methodologically through logic programming, theoretically through process ontology building and then theorem proving. That is, we first build up a formal process ontology of TPNs by extending SCOPE (standing for Situation Calculus Ontology for PEtri nets, proposed earlier in [9]) with temporal constraints on transition firings. We then demonstrate that, with this temporal SCOPE, sequential, iterative, or nondeterministic composite sequences of transition firings in TPNs can be further axiomatized through macro-expanding the basic action of *transition firing*. Finally, it is shown that theorem-proving, i.e., computing the consequences of such an ontology, can be carried out efficiently through logic programming in Prolog.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 134–143, 2012.

In a sense, our work can be understood as simply using TPNs to exercise the paradigm introduced in the book *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems* [8]. In [8], a dynamical system is axiomatized as a logical theory of basic actions, in the language of Situation Calculus. The generated theory is then macro-expanded on the basic actions into a GOLOG program, which is a theory composed of complex actions. A GOLOG program can be further extended in different ways to match varying system modeling requirements. For example, temporal GOLOG incorporates a time line; ConGOLOG handle concurrency explicitly; whereas RGOLOG allows exogenous actions, and interrupts. Finally, these GOLOG programs are implemented in Prolog.

The implemented tool for TPNs is notable for its conciseness. Its core portion is domain-independent and only contains four Prolog clauses. Among them, one is used to specify the precondition requirement for a transition to fire. That is, a transition is enabled to fire iff all places that enter it contain at least one token. The other three are used to specify how the systems evolve over the actions of transition firing. That is, the number of tokens at a place will never change upon an occurrence of a transition firing, unless this transition node enters, or leaves, the place.

Rigorously proving the correctness of the implementation is somewhat intricate and is not covered in this current paper. Basically, a correctness proof should be carried out in two steps. The first step involves showing that the intended interpretation of an instance of Temporal SCOPE, captured by the graph-theoretic definitions of an instance of a TPN, is actually a model of the set of axioms for the Temporal SCOPE $\mathcal{D}_{tscope}$. Next, we need to justify that whenever the implemented Prolog program succeeds on a sentence, the sentence is logically entailed by $\mathcal{D}_{tscope}$; whereas whenever it fails, the negation of the sentence is entailed by $\mathcal{D}_{tscope}$.

The remainder of this paper is organized as follows. Preliminaries on Situation Calculus, GOLOG, and TPNs are covered in Section 2. Section 3 is the major section. Temporal SCOPE and an example of its GOLOG program are introduced in 3.1 and 3.2, respectively. A Prolog implementation of the program is presented in Section 3.3. Finally the work is summarized in Section 4.

2 Preliminaries

In this section, we first give a brief overview of Situation Calculus and GOLOG. We then briefly introduce TPN.

2.1 Situation Calculus and GOLOG

Situation Calculus is a logical language for representing actions and changes in a dynamical domain; it is first proposed by McCarthy and Hayes in 1969 [7]. The language $\mathcal{L}$ of Situation Calculus as stated by [8] is a second-order many-sorted language with equality.

Three disjoint sorts: *action*, *situation*, *object* (for everything else in the specified domain) are included in $\mathcal{L}$. For example, *rain* denotes the act of raining, and $putdown(x, y)$ denotes the act of *object x putdown y on the ground*. A *situation* characterizes a sequence of actions in the domain. The constant situation S_0 is to denote the empty sequence of actions, whereas $do(a, s)$ denotes the successor situation after performing action a in situation s. The situation $do(clean, do(spray, s))$ denotes the situation resulting from first *spraying water on the car* and then *cleaning the car body*. Note that the situation term distinguishes itself from $do(spray, do(clean, s))$. It is easy to see that a situation corresponds to a finite sequence of actions.

The binary predicate $\sqsubset$ specifies the order between situations. For example, $s \sqsubset s'$ stands for that the situation s' can be reached by performing one or several actions from s. $s \sqsubseteq s'$ is an abbreviation of $s \sqsubset s' \vee s = s'$. In addition, a predicate $Poss(a, s)$ is applied to specify the legality of performing action a in situation s, For example, $Poss(rain, s) \equiv heavyCloudy(s)$ says that it is possible to rain only iff the sky is with heavy cloud.

A Situation Calculus theory in general might contain situation independent relations, like $matchLocation(Albany)$, and situation independent functions, like $size(Plot2)$. However, in many of the more interesting cases, the values of relations and functions change between situations. Accordingly, a relational/functional fluent, in $\mathcal{L}$ is defined as a predicate/function, whose last argument is always a situation (e.g., $captain(John, do(catchFever, S_0))$ is a relational fluent, whereas $weight(John, do(recover, s))$ is a functional fluent).

GOLOG is a logic programming language for description and execution of complex actions using domain-specific Situation Calculus primitive actions. It provides imperative programming constructs, including

- a, a primitive action;
- $\alpha; \beta$, action α is followed by action β;
- $p?$, test action on the condition p;
- **if** p **then** α **else** β, conditionals;
- $\alpha|\beta$, nondeterministic choice of action α or action β;
- $(\pi\, x)\alpha(x)$, nondeterministic choice of arguments;
- $\alpha^\star$, nondeterministic iteration.
- Procedures.

The semantics of GOLOG programs is defined on the abbreviation $Do(\delta, s_1, do(\vec{a}, s_1))$, which denotes that execution of GOLOG program δ in the situation s_1 leads to $do(\vec{a}, s_1))$, an abbreviation to the situation

$$do(\alpha_n, do(\alpha_{n-1}, \ldots, do(\alpha_1, S_0) \ldots)).$$

The structure of δ is defined inductively through macro-expansions on the above constructs.

2.2 Time Petri Nets

Definition 1. *A Petri net (PN) is a pair* (N, M_0) *where N is a triple* (P, T, F) *such that P is a finite set of node elements called places, T is a finite set of node*

elements called transitions, $F \subseteq (P \times T) \cup (T \times P)$ *consists of ordered pairs, and* M_0 *is the initial marking, a mapping in the form* $M\colon P \to \mathcal{N}$*, indicating the initial assignment of a non-negative integer* k *to each place* p *in* P*. (In this case, we say that the place* p *is marked with* k *tokens.)*

A marking M *for* N *in PN is defined as a vector* $(M(p_1), \ldots, M(p_m))$*, where* $p_1, \ldots, p_m$ *is an enumeration of* P *and* $M(p_i)$ *tokens are assigned to node* p_i*, for all* i *such that* $1 \leq i \leq m$*.*

The elements in $P \cup T$ are called nodes of PN. Given a node $u \in PN$, the set ${}^\bullet u = \{v | (v, u) \in F\}$ is the pre-set of u, where each v is the input of u, and the set $u^\bullet = \{v | (u, v) \in F\}$ is the post-set of u, where each v is the output of u. In any marking M, a place p is marked if $M(p) > 0$. A transition t is *enabled* in M if every place in ${}^\bullet t$ is marked in M. An enabled transition in M can occur (fire) and there may exist several enabled transitions in M. One of them will actually fire, leading to the successor marking M'.

Definition 2. *A marking transition from* M *to* M' *due to the firing of* t *(written as* $M \stackrel{t}{\Longrightarrow} M'$*) is defined by*

$$M'(p) = \begin{cases} M(p) - 1 & \text{if } p \in {}^\bullet t \text{ and } p \notin t^\bullet \\ M(p) + 1 & \text{else if } p \notin {}^\bullet t \text{ and } p \in t^\bullet \\ M(p) & \text{otherwise} \end{cases}$$

for every place p*.*

Graphically, a Petri Net PN can be represented by a bipartite graph, where each place is represented by a circle, each transition is represented by a rectangle, the flow relation of the Petri net F is represented by arcs from places to transitions or from transitions to places, and k black dots will be placed into the circle for place p if it is marked with k tokens at M.

Time Petri Nets extend basic Petri nets by attaching each transition node t a time interval $[i_{min}, i_{max}]$, denoting the earliest firing time i_{min} and the latest firing time of t. Note that in our implementation, we also allow transition firings having durations.

3 Go beyond the SCOPE

The theory $\mathcal{D}_{scope}$ for Situation Calculus Ontology of PEtri nets (SCOPE) is first proposed in [9]. Section 3.1 proposes $\mathcal{D}_{tscope}$, which extends $\mathcal{D}_{scope}$ to accommodate time. Section 3.2 is a specification of an example GOLOG program built from $\mathcal{D}_{tscope}$. Section 3.3 is a Prolog implementation of the program.

3.1 Temporal SCOPE

Still, the only action in $\mathcal{D}_{tscope}$ is *fire* and the only fluent is *Tkns*. A temporal version of the primitive action precondition axiom is defined for *fire*, whereas the temporal successor state axiom is defined for the fluent *Tkns*. The relation *pre* and *post* is used to specify the topology of a given net. Finally, the set of foundational axioms is extended to temporal domain as well.

- **Primitive Action:**
  - $fire(t, \tau)$. The transition t fires at time τ.
- **Fluent:**
  - $Tkns(p, s)$. In situation s, the number of tokens at place p.
- **Situation-Independent Relations:**
  - $pre(m, n)$. Node m enters node n.
  - $post(m, n)$. Node n enters node m.
- **Foundational Axioms $\mathcal{D}_f$:**
  - $do(a_1, s_1) = do(a_2, s_2) \supset a_1 = a_2 \wedge s_1 = s_2$.
  - $(\forall P).P(S_0) \wedge (\forall a, s)\big(P(s) \supset P(do(a, s))\big) \supset (\forall s)P(s)$.
  - $\neg s \sqsubset S_0$.
  - $s \sqsubset do(a, s') \equiv s \sqsubseteq s'$.
  - $start(do(a, s)) = time(a)$.
- **Primitive Action Precondition Axiom:**
  - $(\forall s, p, t)\ (Poss(fire(t, \tau), s) \equiv pre(p, t) \supset Tkns(p, s) \geq 1)$. The transition t is enabled to fire at situation s iff each place that enters the transition node t contains at least one token.
- **Successor State Axiom:**
  - $(\forall s, p, a, n)\ (Tkns(p, do(a, s)) = n \equiv \gamma_f(p, n, a, s) \vee$
    $(Tkns(p, s) = n \wedge \neg\exists n'\ \gamma_f(p, n', a, s)))$,
    where $\gamma_f(p, n, a, s) \stackrel{def}{=} \gamma_{f_e}(p, n, a, s) \vee \gamma_{f_l}(p, n, a, s)$, referring to the two sets of firing actions that cause the number of tokens at place p on situation $do(a, s)$ to be equal to n:
    - $\gamma_{f_e}(p, n, a, s) \stackrel{def}{=} (\exists t)(pre(t, p) \wedge \neg post(t, p) \wedge n = Tkns(p, s) + 1 \wedge a = fire(t, \tau))$ (at situation s, the number of tokens at place p is $(n-1)$, and transition t, which enters p, fires at s);
    - $\gamma_{f_l}(p, n, a, s) \stackrel{def}{=} (\exists t)(pre(p, t) \wedge \neg post(p, t) \wedge n = Tkns(p, s) - 1 \wedge a = fire(t, \tau))$ (at situation s, the number of tokens at place p is $(n+1)$, and transition t, which leaves p, fires at s);

  Note that the above Successor State Axiom summarizes all conditions where the number of tokens at place p is n at situation $do(a, s)$: n could be achieved by action a from situation s, or at situation s the number of tokens at p is already n and the action a that occurs in s will not change it to some other values.

3.2 A SCOPE-Based GOLOG Program

In this section, we use GOLOG to depict a coffee delivery scenario (an example adopted from Example 7.4.1 of [8]). We first describe the domain as a TPN, and then transform the net into a GOLOG Program.

In this scenario, the kitchen, and the offices of employees (Bill, Joe, Mary and Sue) are at various locations. There is a coffee maker in the kitchen. A robot begins its day by scheduling coffee deliveries to everyone during their preferred time. The robot can carry one cup of coffee at a given time and traveling between locations takes time too. A TPN description is shown in figure 1. A GOLOG description is presented as follows.

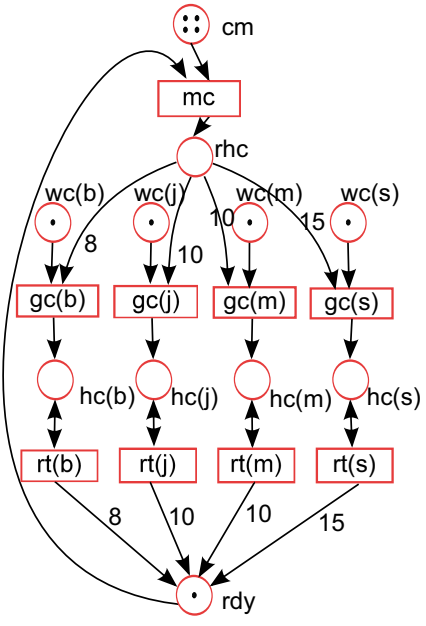


Fig. 1. A Petri Net Example

- **Nodes**
  - Place nodes: cm, Coffee Maker; $wc(X)$, X wants coffee; $hc(X)$, X has coffee; rdy, robot is ready; rhc, robot has coffee.
  - Transition nodes: mc, make coffee; $gc(X)$, robot gives coffee to X; $rt(X)$, robot return from the office of X to the kitchen.
- **Topology of the Net** $pre(cm, mc), pre(rdy, mc), pre(mc, rhc), \ldots$ $pre(rt(b), rdy), pre(rt(j), rdy), pre(rt(m), rdy), pre(rt(s), rdy)$.[1]
- **Initial Situation:**
  - Unique names axioms stating that following terms are pairwise unequal: b, j, s, m, ktn(Coffee Maker in the kitchen), $ofc(b)$, $ofc(j)$, $ofc(s)$, $ofc(m)$.
  - Initial Markings:
    $Tkns(cm, s0) = 4$, $Tkns(rdy, s0) = 1$, $Tkns(wc(b), s0) = 1$,
    $Tkns(wc(j), s0) = 1$, $Tkns(wc(m), s0) = 1$, $Tkns(wc(s), s0) = 1$,
    $Tkns(rhc, s0) = 0$, $Tkns(hc(b), s0) = 0$, $Tkns(hc(j), s0) = 0$,
    $Tkns(hc(m), s0) = 0$, $Tkns(hc(s), s0) = 0$.
  - The time for initial situation S_0 is 0: $start(S_0, 0)$.
- **Coffee Delivery Preferences**
  $wtsCf(p, t_1, t_2) \equiv$
  $(p = b \wedge t_1 = 100 \wedge t_2 = 110) \vee$ $(p = j \wedge t_1 = 90 \wedge t_2 = 100) \vee$
  $(p = s \wedge t_1 = 140 \wedge t_2 = 160) \vee (p = m \wedge t_1 = 130 \wedge t_2 = 170)$

[1] The specification simply matches Figure 1.

- **Robot Travel Time**
  $tvlT(ktn, ofc(b)) = 8$, $tvlT(ktn, ofc(s)) = 15$,
  $tvlT(ktn, ofc(j)) = tvlT(ktn, ofc(m)) = 10$,
  $tvlT(loc, loc) = 0$, $tvlT(loc_1, loc_2) = tvlT(loc_2, loc_1)$.
- **Transition Firing Times**
  $time(fire(t, \tau)) = \tau$.
- **GOLOG Procedures** The procedure for delivering coffee is defined as a recursive procedure, which calls the other procedure at each recursion level.
  proc $dlvrCf(\tau)$
  $[(\forall p, n)(Tkns(wc(p), n) \supset n = 0)]? \mid$
  $dlvrOneCf(\tau);\ dlvrCf(now)$
  endProc
  The procedure for delivering one coffee at the time τ for some person. it is assumed that initially the robot is ready to make coffee.
  proc $dlvrOneCf(\tau)$
  $\{(\pi\, p, n_1, \tau, \tau_0, tT, i_{min}, i_{max}, wait)$
  $[(Tkns(hc(p), n_0) \wedge n_0 = 0 \wedge (Tkns(wc(0), n_1) \wedge \geq 1 \wedge wait \geq 0 \wedge$
  $wtsCf(p, i_{min}, i_{max}) \wedge tvlT(ktn, ofc(p), tT) \wedge$
  $(i_{min} \leq \tau + tT + wait) \wedge (\tau + tT + wait \leq i_{max}) \wedge$
  $\tau = \tau + tT + wait \wedge \tau_0 = \tau + wait)]?;$
  $fire(mc, \tau_0);\ fire(gc(p), \tau);$

  $[(\pi\, \tau_1, \tau_2, tT)$
  $(tvlT(ofc(p), ktn, tT) \wedge now(\tau_1) \wedge \tau_2 = \tau_1 + tT)?; fire(rt(p), \tau_2)$
  $]\}.$
  endProc

3.3 Implementation through Theorem-Proving

Temporal GOLOG interpreter is offered by [8]. Slightly tuned version, together with the full version of the Prolog code for this section will be uploaded to internet. One of the major implementation technique is the Lloyd-Topor transformations and their details can also be found in [8].

The Precondition Axiom in SCOPE is logically equivalent to

$$Poss(fire(t, \tau), s) \equiv \neg(\exists p)\, \neg\big(pre(p, t) \supset (\exists n)\, Tkns(p, n, s) \wedge n \geq 1\big),$$

Thus: $Poss(fire(t), s) \equiv \neg aux\_p(t, s)$, where the new predicate aux_p is defined by $aux\_p(t, s) \equiv (\exists p)\, \neg\big(pre(p, t) \supset (\exists n)\, NumTkns(p, n, s) \wedge n \geq 1\big)$,

The if-half of these two definitions are: $\neg(aux\_p(t, s)) \supset poss(fire(t), s), and$

$$(\exists p)\, \neg\big(pre(p, t) \supset (\exists n)\, Tkns(p, n, s) \wedge n \geq 1\big) \supset aux\_p(t, s).$$

By introducing $aux\_q(p, s) \equiv (\exists n).\, Tkns(p, n, s) \wedge n \geq 1$

The second sentence becomes $(\exists p)\, \neg\big(pre(p, t) \supset aux\_q(p, s)\big) \supset aux\_p(t, s)$. Hence, the Lloyd-Topor transformations applied to the Precondition Axiom yield the following Prolog clauses: $\neg(aux\_p(t, s)) \supset poss(fire(t), s)$,

$(\exists p)\big(\neg(aux\_q(p,s)) \wedge pre(p,t)\big) \supset aux\_p(t,s)$, and $(\exists n)\ \big(numTkns(p,n,s) \wedge n \geq 1\big) \supset aux\_q(p,s)$. Replace $\neg$ by Prolog `not`, and replace $\wedge$ and $\vee$ with Prolog disjunction and conjunction operators "`;` " and "`,`", respectively, we obtain:

```
% Precondition for the action "fire"
poss(fire(T, Tau),S):- not (pre(P,T), not (tkns(P,N,S), N >=1)).
```

The sufficient condition of the Successor State Axiom in SCOPE is

$$\gamma_F^+(p,t,n,a,s) \vee (Tkns(p,n,s) \wedge \neg\, \gamma_F^-(p,t,a,s)) \supset Tkns(p,n,do(a,s)),$$

where $\gamma_F^+(p,t,n,a,s)$ corresponds to the Prolog part `achieveN`, and $\neg\gamma_F^-(p,t,a,s))$ corresponds to the Prolog part `falsifyN`. Using similar Lloyd-Topor transformation techniques, the Successor State Axiom for the fluent *tkns* is stated in Prolog as follows, where "`is`" is a special predefined operator to force mathematical operations:

```
% Fluent "tkns" in Successor State
tkns(P,N,do(A,S)):-
   achieveN(P,N,A,S); tkns(P,N,S), not falsifyN(P,A).

achieveN(P,N,A,S):-
   (A= fire(T, Tau), pre(T,P), not pre(P,T), tkns(P,N0,S), N is N0+1);
   (A= fire(T, Tau), pre(P,T), not pre(T,P), tkns(P,N0,S), N is N0-1).

falsifyN(P,A) :-
   A = fire(T, Tau),((pre(T,P), not pre(P,T)) ; (pre(P,T), not pre(T,P))).
```

The two GOLOG procedures are as follows.

```
proc(dlvrCf(Tau),
   (?(all(p, all(num,tkns(wc(p),num) => num =:= 0 ))) #
      dlvrOneCf(Tau) : pi(t, ?(now(t)): dlvrCf(t))
   )
).

proc(dlvrOneCf(Tau),
   pi(p,
      % p does not have coffee and p wants coffee, then give coffee to p.
      pi(num0, pi(num1, pi(tau, pi(tau0, pi(tT, pi(imin, pi(imax, pi(wait,
         ?(tkns(hc(p),num0) & num0 =:= 0 & tkns(wc(p),num1) & num1 >=1 &
           wtsCf(p,imin,imax) & wait $>= 0 & tvlT(ktn,ofc(p),tTime) &
           imin $<= Tau+tTime+wait & Tau+tTime+wait $<= imax &
           tau $= Tau+tTime+wait & tau0 $=Tau+wait):
         fire(mc, tau0): fire(gc(p), tau)
      )))))):
      % return from p's office back to the coffee maker.
      pi(tau1, pi(tau2, pi(tT,
         ?(tvlT(ofc(p), ktn, tT) & now(tau1) & tau2$=tau1+tT):
         fire(rt(p), tau2)
      )))
   )
).
```

Two different schedules (with minimized time constraints) are obtained by running the program and the resulting net marking can also be tested.

```
[fire(mc, 80), fire(gc(j), 90), fire(rt(j), 100), fire(mc, 100),
fire(gc(b), 108), fire(rt(b), 116), fire(mc, 120), fire(gc(m), 130),
fire(rt(m), 140), fire(mc, 140), fire(gc(s), 155), fire(rt(s), 170)]
More?
[fire(mc, 80), fire(gc(j), 90), fire(rt(j), 100), fire(mc, 100),
fire(gc(b), 108), fire(rt(b), 116), fire(mc, 125), fire(gc(s), 140),
fire(rt(s), 155), fire(mc, 155), fire(gc(m), 165), fire(rt(m), 175)]
More?

P = rdy N = 1, P = hc(s) N = 1, P = rhc N =0, P = wc(s) N = 0,
P = cm N =0, P = hc(m) N = 1, P = wc(m) N = 0, P = hc(b) N = 1,
P = wc(b) N = 0, P = hc(j) N = 1, P = wc(j) N = 0,
```

4 Summary

In this paper, we approach TPN representation and reasoning through mathematical logics. As a result, a software tool for TPNs is implemented. The isomorphic correspondence between the situation s in situation calculus (which is a sequence of actions starting from the initial state S_0), and the marking M in Petri nets (which results from a sequence of transition firings starting from the initial marking M_0), is our basic insight, which enabled us to design a tool that adopts in a natural way the simplicity and the expressive power of TPNs.

Formal methods, mostly model checking, have been applied to study the dynamical behaviors of Petri nets for more than a decade (for example, see [3,1]). The execution semantics of these methods are captured by Linear Temporal Logic (LTL), which is a special dialect of modal logic, with modalities referring to time. When compared, the current approach as an alternative formalism distinguishes itself in three different ways. First, Situation Calculus is mostly first-order logic based and thus is balanced with expressive power and computational complexity, additionally, note that various practical automated solvers for first-order logic described problems are available these days. Second, both actions and markings as situations are treated as objects, so that they could be quantified over directly. Third, a large portion of the tool is domain-independent thus its reusability is high. Previous efforts directly integrating Petri nets and Prolog programming include [2] and [5]. More recently, in [4], high level Petri nets are transformed into Prolog for model checking. These approaches, however, take Prolog simply as an implementation environment and are unable to provide axiomatical justification on their ways of transformation from Petri nets to Prolog, or logics in general.

This work opens up several new avenues for TPN-system representation and reasoning. Meanwhile, we are in particular working towards using a TPN GOLOG program to analyze the tracing processes in supply chain event management [6]

to facilitate informed customer choices, an ongoing project supported by NSF[2]. Finally, we remark that a similar effort is taken to axiomatize the execution semantics of UML Activity Diagrams ([10], [11]).

References

1. Chatain, T., Jard, C.: Time Supervision of Concurrent Systems Using Symbolic Unfoldings of Time Petri Nets. In: Pettersson, P., Yi, W. (eds.) FORMATS 2005. LNCS, vol. 3829, pp. 196–210. Springer, Heidelberg (2005)
2. Domenici, A.: Petri nets in logic. Microprocessing and Microprogramming 30, 193–198 (1990)
3. Esparza, J., Heljanko, K.: Implementing LTL Model Checking with Net Unfoldings. In: Dwyer, M.B. (ed.) SPIN 2001. LNCS, vol. 2057, pp. 37–56. Springer, Heidelberg (2001)
4. Farwer, B., Leuschel, M.: Model checking object Petri nets in Prolog. In: Moggi, E., Warren, D.S. (eds.) The 6th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming, Verona, Italy, pp. 20–31 (2004)
5. Garzotto, F., Ghezzi, C., Mandrioli, D., Morzenti, A.: On the Specification of Real-Time Systems Using Logic Programming. In: Nichols, H.K., Simpson, D. (eds.) ESEC 1987. LNCS, vol. 289, pp. 180–190. Springer, Heidelberg (1987)
6. Liu, R., Kumar, A., van der Aalst, W.: A formal modeling approach for supply chain event management. Decision Support Systems 43, 761–778 (2007)
7. McCarthy, J., Hayes, P.: Some philosophical problems from the standpoint of artificial intelligence. Machine Intelligence 4, 463–502 (1969)
8. Reiter, R.: Knowledge in action: logical foundations for specifying and implementing dynamical systems. MIT Press, Cambridge (2001)
9. Tan, X.: SCOPE: A Situation Calculus ontology of Petri nets. In: Galton, A., Mizoguchi, R. (eds.) 6th International Conference of Formal Ontology in Information Systems, Toronto, Canada, pp. 227–240 (2010)
10. Tan, X., Gruninger, M.: On the computational complexity of the reachability problem in UML activity diagrams. In: IEEE International Conference on Intelligent Computing and Intelligent Systems, Shanghai, China, vol. 2, pp. 572–576 (2009)
11. Tan, X., Gruninger, M.: Towards axiomatizing the semantics of UML activity diagrams: a situation-calculus perspective. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence, Toronto, Canada (2010)

[2] National Science Foundation Award Number: 0955935. For more information, please go to "`http://www.nsf.gov`".

An Algorithm of Channel Assignment of MAC Layer in Ad Hoc Network Based on Dynamic Game with Perfect and Complete Information*

Xiaochen Lai[1,2], Quanli Liu[3], Wei Wang[3], Likun Li[4], Simin Lu[4], and Di Wu[4]

[1] School of Software, Dalian University of Technology, Dalian, China
[2] Faculty of Electronic Information and Electrical,
Dalian University of Technology, Dalian, China
far.away@tom.com
[3] Faculty of Electronic Information and Electrical,
Dalian University of Technology, Dalian, China
{liuql,wangwei}@dlut.edu.cn
[4] School of Software, Dalian University of Technology, Dalian, China
{lilikun,Simin.Lu,diwu.michael}@mail.dlut.edu.cn

Abstract. The nodes in Ad Hoc networks compete channels when communicating, with the features of no center and self-organization. In traditional channel assignment strategy of MAC layer, each node does not consider the demands to channel resource of other nodes, which hinders improving the network performance. An algorithm of channel assignment based on complete and perfect information dynamic game theory is proposed, supposing all the nodes are rational and greedy. Each node selects channels dynamically by backward induction due to strategies of other nodes, thus lead to Nash equilibrium finally. Experiments show that the network throughput is improved and the packet loss rate is reduced by this method effectively.

Keywords: Ad Hoc Network, channel assignment, dynamic game with perfect and complete information, backward induction, Nash equilibrium.

1 Introduction

Ad Hoc network[1] consists of a set of mobile nodes which may connect to the network at any time and communicate wirelessly, depending on no fixed telecommunication infrastructure. The Ad Hoc is a decentralized self-organizing network with dynamic network topology. It is widely applied to Wireless Local Area Network, family network, search and rescue operations, business education, sensor network, and so on[2]. However, its flexibility and zero dependence on infrastructure may also introduce some problems, such as wireless bandwidth limit, poor safety performance, weak network scalability, multi-hop routing, etc. In this paper, we look at the problem of channel assignment of MAC layer protocol.

* Supported by "the Fundamental Research Funds for the Central Universities" under Grant DUT12JR03.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 144–155, 2012.

MAC protocol is designed to solve the problem of how to assign wireless channel among competing users, i.e. how wireless nodes access wireless channel to transmit data. At present, the IEEE 802.11 MAC protocol is adopted as the WLAN standard[3], which chooses CSMA/CA technology to reduce the probability of collision. Research shows that it transmits packets in a single channel with no ability of channel assignment[4]. It can not make full use of channels when several channels are available. The common multi-channel protocols are classified into two categories: packet-oriented protocols and connection-oriented protocols. The former have the virtues of low cost and easy to realize, such as the DCA protocol. The latter are more reliable in transmission, while the typical protocols are MMAC, HRMA and McMAC.

The DCA (Dynamic Channel Assignment)[5] protocol uses a fixed control channel to transmit control packets. The rest channels are divided into several independent data channels to transmit data packets. This protocol is easy to control, but it's difficult to achieve high utilization rate of channels.

HRMA (Hop-Reservation Multiple Access) is a multi-channel protocol[6] with multi-slot interaction. Slots are produced by synchronization technology and assigned to RTS packet, CTS packet, DATA packet and ACK packet respectively, which avoid collisions between RTS packet and CTS packet. Collisions between DATA packet and ACK packet can also be avoided because they are sending on the basis of RTS/CTS handshake. Even though the protocol resolves the collisions of control packets, it needs more extremely strict synchronization steps.

The MMAC(Multi-channel MAC)[7-8] protocol takes a connection-oriented channel assignment strategy. The idea comes from a protocol called 802.11 PSM. The channels are divided into several data channels and a control channel, which is shared by all the nodes in the network. In MMAC protocol, the condition that many pairs of nodes selecting a same data channel is possible because each transmission may not occupy the whole stage of data transmission. Under this condition, we need to determine the ownership of the channel by arranging the time sequence or competition. MMAC also requires synchronization.

The channel assignment strategy of McMAC(Multi channel MAC) protocol[9] is also connection-oriented. Each node selects a seed to produce a pseudo-random jumping sequence. When the node is idle, it selects channels based on the jumping sequence to realize channel transformation. Each node encapsulates the seed into the packets to be sent, so neighbor nodes will know the channel sequence that the node selects.

None of the protocols above takes the channel assignment strategy of its neighbor nodes into consideration, which may cause collisions easily. The channel assignment is carried out by nodes themselves in Ad Hoc network. Each node needs to consider the behavioral strategy of its neighbors to meet their own needs to the greatest extent. Meanwhile, the behavior of nodes will affect their neighbors too. The combination of behavioral strategies of all nodes decides the final payoff of each node. So the channel assignment among nodes belongs to multi-person decision-making problem, which fits to be solved by game theory. Based on dynamic game theory with complete and perfect information, a protocol of channel assignment named DGPCI-DCA is

proposed. The nodes under communication form a district, while the state and possibility of channel selection scheme of neighbors in the district should be considered when a node selects channel to communicate. Because the time of communication of neighbors can not be predicted, we assume that the channel occupancy around nodes reflects the possibility of the nodes to start communication. Then, the payoff of each channel nodes select can be calculated by backward induction, namely the probability to avoid conflicts. Each node chooses the channel of the greatest payoff to transmit data and then the nodes of the district achieve the Nash equilibrium[10], while we can now get the best network performance. The simulation results point out that the DGPCI-DCA shows good characteristics in terms of throughput and packet loss, outperforming the IEEE 802.11 and DCA which belong to packet-oriented protocol too.

The remainder of this paper is organized as follows. Section 2 states the introduction of game theory. In section 3 we propose the DGPCI-DCA protocol. Section 4 shows an instance based on the protocol. Section 5 presents the experiments and results. Section 6 briefly concludes this paper and presents our future work.

2 Game Theory

Game theory mainly focuses on the study of multi-person decision-making methods based on the following hypothesis: All the players are rational and greedy. None of them would choose strictly worst strategy and each of them is assumed to know that other players are rational.

2.1 Nash Equilibrium

In game theory, Nash equilibrium is a solution of a game involving two or more players, in which each player is assumed to know the equilibrium strategies of the other players. If each player has chosen a strategy and no player can benefit by changing his or her strategy while the other players keep theirs unchanged, then the current set of strategy choices and the corresponding payoffs constitute Nash equilibrium.

Definition: a game with n players is shown below.

$$G = \{S_1, S_2, \cdots, S_n; u_1, u_2, \cdots, u_n\} \tag{1}$$

S_i is the strategy set for player i and u_i is the utility function for player i. S_i is composed of s_i, which is a decision-making behavior of player i. To player i, given a strategy profile $\{s_1^*, s_2^*, \cdots, s_n^*\}$, if s_i^* is the optimal response strategy to $\{s_1^*, \cdots, s_{i-1}^*, s_{i+1}^*, \cdots, s_n^*\}$, or s_i^* is not worse than $\{s_1^*, \cdots, s_{i-1}^*, s_{i+1}^*, \cdots, s_n^*\}$, which selected by the other n-1 players, the strategy

profile $\{s_1^*, s_2^*, \cdots, s_n^*\}$ is called a Nash equilibrium of the game. That is, given the utility function u_i to player i, it is true of the formula below to all the s_i belong to S_i:

$$u_i(s_1^*, \cdots, s_{i-1}^*, s_i^*, s_{i+1}^*, \cdots, s_n^*) \geq u_i(s_1^*, \cdots, s_{i-1}^*, s_{i+1}^*, \cdots, s_n^*) \quad (2)$$

In other words, s_i^* is the solution of the optimization problem in formula (3) below:

$$\max_{s_1 \in S_1} u_i(s_1^*, \cdots, s_{i-1}^*, s_i^*, s_{i+1}^*, \cdots, s_n^*) \quad (3)$$

The protocol is related to the concept of Nash equilibrium closely. To arrive at an agreement in a game among players, the strategy profile of an effective protocol must be Nash equilibrium. Otherwise, at least one player violates the protocol.

2.2 Dynamic Game with Perfect and Complete Information

Dynamic game with perfect and complete information assumes that each player is well aware of the whole game process before each stage of the game. The stages of player are shown below:

Stage 1: Player 1 chooses action a_1 from strategy set A_1;

Stage 2: Player 2 chooses action a_2 from strategy set A_2, after the action of a_1 was observed;

Stage 3: The utility of player 1 and player 2 is $u_1(a_1, a_2)$ and $u_2(a_1, a_2)$ respectively.

Dynamic game with perfect and complete information has three basic characteristics. Firstly, actions take place in sequence. Secondly, all previous actions could be observed before taking next step. Thirdly, under every possible action combination, the players' gain is common knowledge. Therefore, this kind of game problem can be solved by backward induction method. In the second stage of the game, because player 1 has selected action a_1, the decision-making problem player 2 is facing can be expressed by formula (4) :

$$\max_{a_2 \in A_2} u_2(a_1, a_2) \quad (4)$$

Given each a_1 in A_1, there is only one solution to the optimization problem of player 2, expressed with $R_2(a_1)$, which is the action taken by player 2 in response to player 1 and called the optimal response. Because player 1 is as the same with player 2 which both of them are able to solve the problem of player 2, and player 1 can expect the response of player 2 to action a_1 that player 1 has made, the problem player 1 meet in the stage 1 is to solve the optimization problem below:

$$\max_{a_1 \in A_1} u_1(a_1, R_2(a_1)) \tag{5}$$

Assume that the optimization problem of player 1 also has the unique solution a_1^*, $(a_1^*, R_2(a_1^*))$ is called backward induction solution of this game. Backward induction solution does not contain the threat of distrust. That is, player 1 predicts that the player 2 will make the optimal response $R_2(a_1)$ to any action a_1 of player 1. The forecast excludes the threat of distrust of player 2. That is, player 2 will not make the response does not meet its own interests in the second stage.

3 The DGPCI-DCA Protocol

3.1 Protocol Overview

This paper puts forwards a channel assignment protocol of MAC layer, called DGPCI-DCA, based on dynamic game with perfect and complete information in Ad Hoc Network. According to the self-organizing and decentralized features of Ad Hoc network, the channels are assigned based on the available channel conditions among sending and receiving nodes and their neighbor nodes. Nodes in the network are rational and greedy. The response strategy of other nodes can be obtained by backward induction. This will ultimately lead to the Nash equilibrium of complete information dynamic game, thus to reduce network packet loss rate, improves network throughput and reduces MAC delay.

In DGPCI-DCA, all channels are divided into a control channel and some data channels. Each node has to maintain a private table called channel-node information table, which contains the occupation information of channels. There are four types of packets used in transmission between nodes, namely RTS, CTS, DATA, and ACK. The sending node initiates a connection request by sending RTS through control channel. The receiving node replies a confirmation message by sending CTS, RTS and CTS packets containing information of available channels of the node itself. All nodes can monitor the RTS and CTS packets, and update their channel-node information table accordingly. The receiving node starts the backward induction according to its channel-node information table in order to select the optimal channel. When asked, it will reply a CTS packet to the sender with the selected channel number and start communication.

3.2 Algorithm Description

In game $\Gamma = \{N, C_i, i \in N\}$, N represents the set of network nodes, C_i represents the set of optional channel to node i, while the element, c_{ij}, represents the channel of node i with the channel number j. Each node in network knows available channels of other nodes. When node i selected data channel c_{ij}, it can be predicted that other nodes will make the most beneficial channel selection of $R(c_{ij})$. Then, the utility function of node i is shown below:

$$u_i = u(c_{ij}, R_1(c_{ij}), \ldots, R_{i-1}(c_{ij}), R_{i+1}(c_{ij}), \ldots, R_n(c_{ij})), c_{ij} \in C_i \tag{6}$$

The optimal channel of node i is the channel c_{ij} that can lead to the maximum value of u_i .If node i wants to select the channel c_{ij} and the neighbors now have not selected the channel c_{ij}, it is not likely to conflict and the performance of network is better. Therefore, the utility function of node i selecting channel c_{ij} is designed as follows:

$$u_i(c_{ij}) = \begin{cases} \prod_{k=1,k\neq i,k\neq m}^{n} (1 - p(k)\frac{1}{n_k}), 0 \le i \le N, c_{ij} unoccupied \\ 0, \mathrm{c_{ij}} occupied \end{cases} \tag{7}$$

$$p(k) = \frac{n - n_k}{n} \tag{8}$$

In formula (7), i is the number of node which will start communication now, and j is the number of channel, while m is the node which node i is about to communicate with. n is the total number of all the data channels and n_k is the number of available channels of node k. If node k will start communication, $\frac{1}{n_k}$ is the probability that it selects channel c_{ij}. $p(k)$ is defined in formula (8), and it represents the busy degree of network near the node k, which we take it as the probability of node k to join the network. If all the channels of node k are available, the value of $p(k)$ is 0, while the value is 1, if all the channels of node k are not available.

3.3 Algorithm Procedures

There are four working states of DGPCI-DCA: monitoring control channel, sending packet, receiving packet and monitoring data channel. Nodes monitor the control channel when idle, and update the channel-node information table according to the RTS and CTS packets received. When a node receives a RTS packet, it selects the optimal channel by backward induction based on its channel-node information table. Then, it should inform other nodes the selected channel by sending CTS packet. Nodes may monitor data channel only after the consultation is completed, while monitor control channel in other idle time. Details of the procedures of algorithm are shown in Figure 1.

3.4 Data Structure of the Algorithm

Each node maintains a two-dimensional array called dca_table, the first index is node number and the second index is channel number. The value of the array element

dca_table[i][j] is the time that node i occupies channel j. Each node obtains its neighbors' information and updates the array by monitoring the control channel. By querying the array, each node gets its neighbors' status, and then calculates the optimal channel, according to DGPCI-DCA.

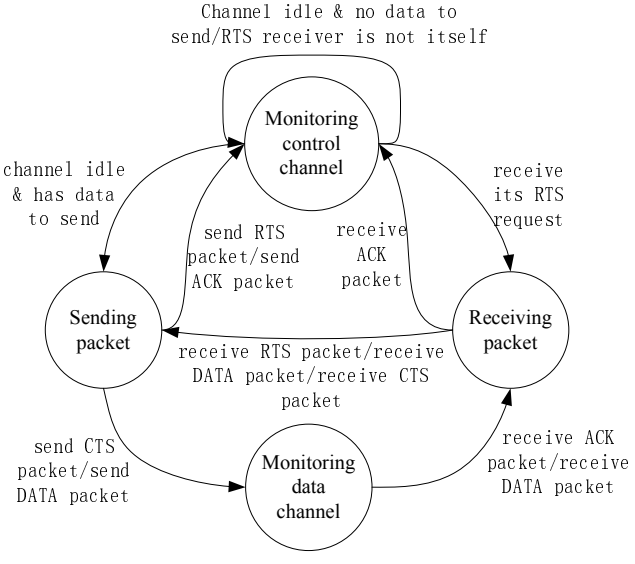


Fig. 1. The transition diagram of working statuses of DGPCI-DCA

4 Instance Analyses

Supposed the network topology structure is as shown in Figure 2, which has four data channels $c_1 \sim c_4$ and a control channel cc. In the figure, each black dot represents a network node and the line between two nodes represents a connection on the channel, where node A and node B are establish a connection while node C and node D are temporarily idle. The status of nodes and channels are as follows:

(1) Nodes status:
Node A and B are negotiating on the control channel cc;
The status information sent by node A includes the available channel c_3 and c_4;
The status information sent by node B includes the available channel c_3 and c_4;
The status information sent by node C includes the available channel c_3 and c_4;
The status information sent by node D includes the available channel c_3;
Node A can receive the status information from node C;
Node B can receive the status information from node C and D, as well as the request information from node A.

(2) Channel status:

c_1: monitored to be occupied by node A、B、C and D.

c_2: monitored to be occupied by node A、B、C and D.

c_3: monitored to be idle by node A、B、C and D.

c_4: monitored to be idle by node A、B and C.

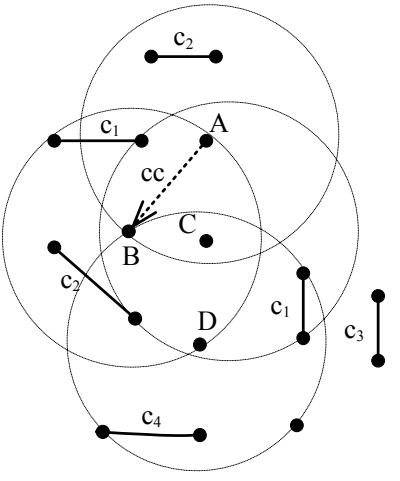


Fig. 2. The network topology structure of instance

For node B, after receiving node A's connection request, it monitors that channel c_1 is occupied. According to formula (7), it figures out that the benefit is 0 when selecting channel c_1; 0 when selecting channel c_2; 0.1875 when selecting c_3 and 0.75 when selecting c_4. And then, the node B will select the channel c_4 to communicate with A, as $u_2(c_{24}) > u_2(c_{2i}), i \neq 4$.

For node C and D, channel c_3 is available for communication. When using the DCA, node B will select channel c_3 to communicate with node A, according to the default order, which will occupy all channels near node C and node D, so the node C and D will have to wait for an idle channel to communicate, which will increase the MAC delay and reduce network throughput.

5 Protocol Simulations

Based on NS2 simulation platform, we compare the DGPCI-DCA to the IEEE 802.11 and DCA from three aspects: throughput, packet loss rate and MAC delay.

5.1 Simulation Scenarios

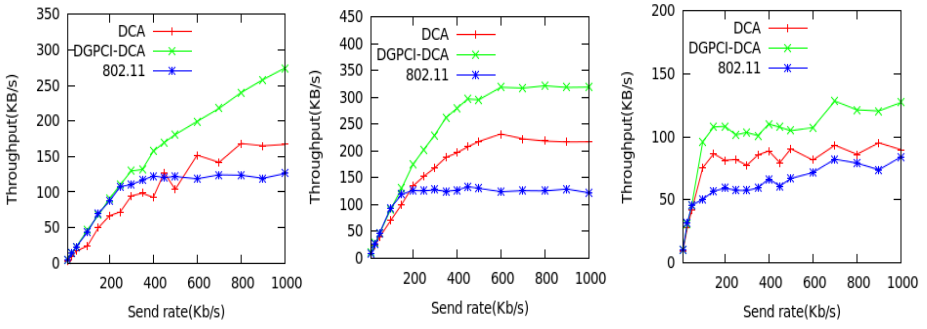
In the simulation model, the nodes of the network form a regular grid topology. On the condition that the packet send rate is rising from 0 to 1000Kb/s, we compare the throughput, packet loss rate and MAC delay of different protocols. The detailed parameters of the experiments are shown in Table 1.

Table 1. Simulation parameters

Parameter	Value
Number of channels	3
Area of the region	500m*500m
Radius of signal transmission	250m
Length of DIF	50us
Length of SIFS	10us
Time of signal propagation	5us
Bandwidth	2Mbps
Packet size	1kB
Length of simulation time	50s
Transport layer protocol	UDP
Traffic generator	CBR

5.2 Results and Analyses of the Simulations

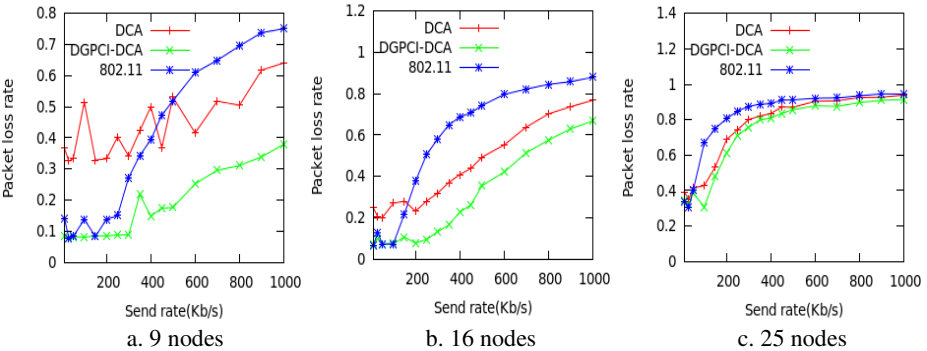
The first set of experiments compare the throughput of different protocols, including DCA, DGPCI-DCA and IEEE 802.11, by varying the packet send rate. The number of nodes is 9, 16 and 25 as shown in Fig.3a, b and c, respectively. The throughput of the three protocols increases following the increment of the packet send rate. The IEEE 802.11 approaches to its upper bound threshold first since the IEEE 802.11 only employs one channel and can't provide enough bandwidth. On the contrary, the throughput of DCA and DGPCI-DCA increases with the increment of packet send rate continuously, because the two protocols use more than one channel for data transmissions. When the nodes density increases, the DGPCI-DCA performs significantly better than DCA all along. This is because that the DGPCI-DCA figures out an idle channel of least probability of collision based on the game theory, which resulting in a greater saturation throughput. When the number of nodes changes from 9 to 16, the network load increases and the channel resource is fully utilized, so the throughput of DCA and DGPCI-DCA increases correspondingly. However, as the number of nodes reaches to 25, all nodes in the network compete for a single control channel. The collisions will increase, leading to a increasing of times of retransmission and a higher packet loss rate, reducing the number of received packets. As a result, the saturation throughput of DGPCI-DCA and DCA descends, but the former is still better than the latter.



a. 9 nodes b. 16 nodes c. 25 nodes

Fig. 3. Throughput against send rate derived by DCA, DGPCI-DCA and IEEE 802.11 under different number of nodes

The second set of experiments compares the packet loss rate of the three protocols by varying the packet send rate. The number of nodes is 9, 16 and 25 as shown in Fig.4a, b and c, respectively. From the result, we can see that the multi-channel protocols perform better than single-channel protocol in the packet loss rate as a whole. However, when the packet send rate is low, the multi-channel protocols need some extra cost to choose channels, for example, sending extra control information. Thus, the packet loss rate of multi-channel protocols is higher than single-channel protocol in light network load. With the increasing of packet send rate, the collisions of IEEE 802.11 increase and the packet loss rate increases rapidly, exceeding the DCA and DGPCI-DCA. The packet loss rate of these two protocols is also growing, but the DGPCI-DCA performs better than the DCA following the increment of network load. This is because the former reduces the collisions by gaming among nodes to a large extent. When the number of nodes is 9, the network is in a low traffic load and the packet loss rate of DGPCI-DCA is equivalent to 20%~60% that of DCA. When the number of nodes reaches to 16, the packet loss rate of DGPCI-DCA is equivalent to 50%~76% that of DCA. When the number reaches to 25, the packet loss rate of DGPCI-DCA is still lower than DCA. The



a. 9 nodes b. 16 nodes c. 25 nodes

Fig. 4. Packet loss rate against send rate derived by DCA, DGPCI-DCA and IEEE 802.11 under different number of nodes

result shows that DGPCI-DCA is beneficial for successful packet transmission, especially in the environment of light network density.

Fig.5 shows the MAC delay among DCA, DGPCI-DCA and IEEE 802.11. The network density varies from 9, 16 to 25, as shown in Fig. 5a, b and c, respectively. MAC delay includes the queuing delay, back delay, channel negotiation delay and transmission delay. When using IEEE 802.11, all the nodes communicate in one channel. The back delay will increase since there are more collisions, which leads to a poor performance of IEEE 802.11, whatever the number of nodes is. As for multi-channel protocols, the result of channel assignment affects the queuing delay and back delay directly. On one hand, the DGPCI-DCA needs to collect more information from neighbors than DCA. The computational complexity is higher too. Thus, the channel negotiation delay of DGPCI-DCA is larger than DCA. On the other hand, by taking the DGPCI-DCA, the possibility of choosing a better channel is larger, which reduces the collisions effectively. The queuing delay and back delay of DGPCI-DCA are less than those of DCA. Therefore, in terms of MAC delay, there is no much difference between DGPCI-DCA and DCA, while multi-channel protocols are much better than single-channel protocol.

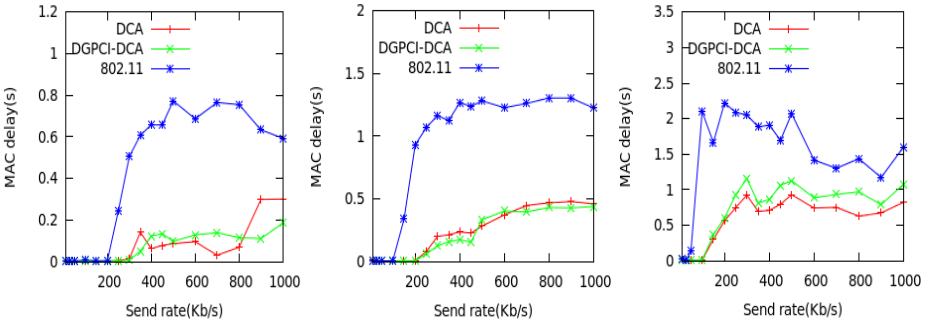


Fig. 5. MAC delay against send rate derived by DCA, DGPCI-DCA and IEEE 802.11 under different number of nodes

The analysis above shows that the throughput and packet loss rate of DGPCI-DCA are better than DCA and IEEE 802.11. The MAC delay of DGPCI-DCA is similar to DCA and better than IEEE 802.11.

6 Conclusions

In this paper, we propose a new packet-oriented channel assignment protocol, called DGPCI-DCA. Nodes play a game with their neighbors to choose a optimal channel to transmit data by means of backward induction, when they need to start communication. Collisions are reduced and the channel utilization rate are improved in this way. Simulation shows that the performance of DGPCI-DCA is better than DCA in network throughput and packet loss rate, and similar in MAC delay. Our

future work will focus on applying the algorithm to network layer of Ad Hoc. For example, in terms of routing, there exist problems of routing strategies, routing control and congestion control, etc, which are competitions for resource among multi-person and can be solved by the game theory too. Therefore, the algorithm based on dynamic game can also be introduced to network layer to solve problems of routing.

References

1. Frodigh, M., Johansson, P., Larsson, P.: Wireless ad hoc networking-the art of networking without a network. Ericsson Review (April 2000)
2. Chlamtac, I., Conti, M., Liu, J.J.-N.: Mobile ad hoc networking: imperatives and challenges. Ad Hoc Networks 1(1), 13–64 (2003)
3. IEEE STD. 802.11. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications (1999)
4. Xu, S., Saadawi, T.: Does the IEEE 802.11 MAC protocol work well in multi-hop wireless ad hoc networks? IEEE Communications Magazine 39(6), 130–137 (2001)
5. Wu, S., Lin, C., Tseng, Y., Sheu, J.: A New Multi-Channel MAC Protocol with On Demand Channel Assignment for Mobile Ad-Hoc Networks. In: International Symposium on Parallel Architectures, Algorithms and Networks, vol. (12), pp. 232–248 (2000)
6. Tzamaloukas, Garcia-Luna-Aceves, J.J.: Channel-Hopping Multiple Access. In: Proc. IEEE ICC 2000, New Orleans, Louisiana, vol. (8), pp. 18–22 (2000)
7. Mo, J., So, H.W., Walrand, J.: Comparison of Multi-Channel MAC Protocols for Wireless Networks. In: ACM/IEEE MSWIM, vol. (10), pp. 209–218 (2005)
8. Vaidya, S.: Multi-channel MAC for Ad Hoc networks: Handling Multi-Channel Hidden Terminals Using a Single Transceiver. In: ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), vol. (3), pp. 222–233 (2004)
9. So, H.W., Walrand, J.: McMAC: a Multi-Channel MAC Proposal for Ad-Hoc Wireless Networks. Technical Report (4), 116–122 (2005)
10. Robert, G.: A Primer in Game Theory. FT Prentice Hall Publisher, London (1992)

A Semantic Approach for Question Analysis

Dai Quoc Nguyen[1], Dat Quoc Nguyen[1], and Son Bao Pham[1,2]

[1] Faculty of Information Technology
University of Engineering and Technology
Vietnam National University, Hanoi
{dainq,datnq,sonpb}@vnu.edu.vn
[2] Information Technology Institute
Vietnam National University, Hanoi

Abstract. The first step that a question answering system must perform is to transform an input question into an intermediate representation. All published works so far use rule-based approaches to realize this transformation in question answering systems. Nevertheless, in existing rule-based approaches, manually creating the rules is error-prone and expensive in time and effort. In this paper, we focus on introducing a rule-based approach that offers an intuitive way to create compact rules for extracting intermediate representation of input questions. Experimental results are promising where our system achieves reasonable performance and demonstrate that it is straightforward to adapt to new domains and languages.

1 Introduction

The goal of question answering systems is to give answers to the user's questions instead of ranked lists of related documents as used by most current search engines [3]. Natural language question analysis component is the first component in any question answering systems. This component creates an intermediate representation of the input question, which is expressed in natural language, to be utilized in the rest of the system.

For the task of translating a natural language question into an explicit intermediate representation of the complexity in question answering systems, all published works so far use rule-based approach to the best of our knowledge. Some question answering systems such as Aqualog [5] and VnQAS [7] manually defined a list of sequence rule structures to analyze questions. However, in these rule-based approaches, manually creating the rules is error-prone and expensive in time and effort.

In this paper, we present an approach to return an intermediate representation of question via FrameScript scripting language [6]. Natural language questions will be transformed into intermediate representation elements which include the construction type of question, question class, keywords in question and semantic constraints between them. Framescript allows users to intuitively write rules to directly extract the output tuple.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 156–165, 2012.

In section 2, we provide some related works and describe our overall system architecture in section 3. We present our approach for question analysis in section 4 and describe our experiments in section 5. Discussion and conclusion will be presented in section 6.

2 Related Works

2.1 Question Analysis in Question Answering Systems

Early NLIDB systems used pattern-matching technique to process user's question and generate corresponding answer [1]. A common technique for parsing input questions in NLIDB approaches is syntax analysis where a natural language question is directly mapped to a database query (such as SQL) through grammar rules.

The PRECISE system [10] maps the natural language question to a unique semantic interpretation by analyzing some lexicons and semantic constraints. In [12], the authors described a template-based system to translate English question into SQL query by matching the syntactic parse of the question to a set of fixed semantic templates. Some other systems based on semantic grammar rules such as Planes [14], Eufid [13]. Semantic grammar-based approaches were considered as an engineering methodology, which allows semantic knowledge to be easily included in the system.

Recently, some question answering systems that used semantic annotations generated high results in natural language question analysis. Aqualog [5] and the first Ontology-based question answering system for Vietnamese [7] perform semantic and syntactic analysis of the input question based on semantic annotations in the use of JAPE grammar rules in GATE framework [2]. Nguyen et al. [9] proposed a language independent approach utilizing JAPE grammars to systematically construct a knowledge base for processing natural language questions. The difference between these approach and our approach is that we use FrameScript scripting language [6] to analyze input questions.

2.2 FrameScript Scripting Language

FrameScript [6] is a language for creating a multi-modal user interfaces. It evolves from Sammut's Probot [11] to enable rule-based programming, frame representations and simple function evaluation.

Each script in FrameScript [6] includes a list of rules which will be matched against user inputs to give the corresponding responses. A scripting rule in the FrameScript language consists of a pattern and responses with the form:
pattern ==> responses.

A pattern expression allows the use of non-terminals to reuse other pattern expressions. Response expressions contain two different types namely sequences and alternatives. A sequence of responses has the form surrounded by brackets: [response 1 | response 2 | ... | another response] and it is also possible to specify

additional conditions to decide which responses will be selected. Furthermore, once the pattern is matched, its components are numbered in order starting from 1. These component are segments of the input that can be referred to in a response using '^'. When '^' is followed by an integer, the corresponding numbered pattern component is used in the output response. In addition, responses utilize the '#' to perform actions. Many examples using *#goto(a_ script, <<trigger>>)* to transform from the current script to another one are described in our companion paper [8].

3 Our Question Answering System Architecture

The architecture of our question answering system is shown in Figure 1. It includes two components: a Natural language question analysis engine and an Answer retrieval. The question analysis component takes the user question as an input and returns an intermediate query-tuple representing the question in a compact form. The role of this intermediate representation is to provide structured information of the input question for later processing such as retrieving answers. Similar to VnQAS [7], the answer retrieval component includes two main modules: Ontology Mapping and Answer Extraction. It takes an intermediate representation produced by the question analysis component and an Ontology as its input to generate semantic answers.

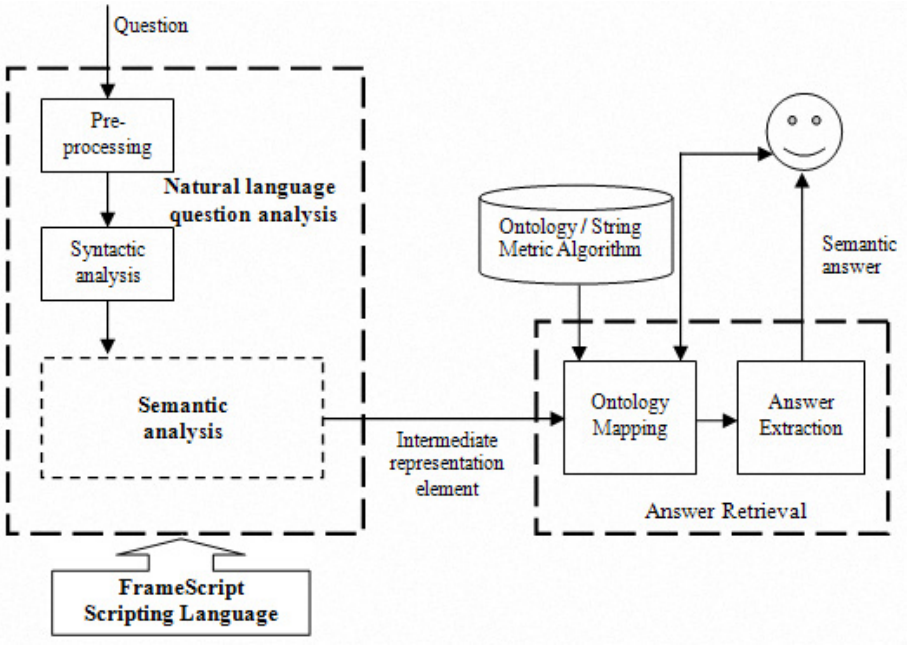


Fig. 1. Architecture of our question answering system

Unlike existing approaches for English [5] and Vietnamese [7] where the intermediate representation couldn't be extracted directly in rules, we will describe an approach to directly extract the representation of user's question using FrameScript scripting language [6].

4 Using FrameScript Language for Question Analysis

The natural language analysis component presented in Figure 1 consists of three modules: preprocessing, syntactic analysis and semantic analysis.

To set the context for the discussion of question analysis, we will first describe the intermediate representation used in our approach, and then focus on proposing our approach to obtain this intermediate representation for a given question.

4.1 Intermediate Representation of an Input Question

Similar to VnQAS [7], the intermediate representation used in our approach aims to cover a wider variety of question types. It consists of a *question-structure* and one or more *query-tuple* in the following format:

(question-structure, question-class, $Term_1$*, Relation,* $Term_2$*,* $Term_3$ *)*

where $Term_1$ represents a concept (object class), $Term_2$ and $Term_3$, if exist, represent entities (objects), *Relation* (property) is a semantic constraint between terms in the question. This representation is meant to capture the semantics of the question.

Simple questions corresponding to basic constructions only have one *query-tuple* and its *question-structure* is the query-tuple's question-structure. More complex questions such as composite questions are constructed by several sub-questions, each sub-question is described by a separate *question-structure*, and the *question-structure* capture this composition attribute. This representation is chosen so that it can represent a richer set of question types. Therefore, some terms or relation in the query-tuple can be missed. Composite questions such as:

"list all students in the Faculty of Information Technology whose hometown is Hanoi?"

has question structure of type *And* with two query-tuples where **?** represents a missed element: *(UnknRel , List , students , ? , Faculty of Information Technology, ?)* and *(Normal , List , students, hometown, Hanoi, ?).*

4.2 Preprocessing Module

The preprocessing module identifies part-of-speech tags in a user's question. After that, we use part-of-speech tags to create basic scripts for detecting words. The basic scripts *Noun*, *Verb*, *Determiner*, *Adjective*, *Adverb*, *Conjunction* and

Preposition are used to identify corresponding nouns, verbs, determiners, adjectives, adverbs, conjunctions and prepositions. In fact, these scripts will be used in creating rules in the syntactic and semantic analysis modules at later stages.

```
Noun ::
    { NN | NNS | NNP | NNPS | NP | NPS | CD } ;;
Verb ::
    { FVG | VBN | VBZ | VBG | VBD | VBP | VB } ;;
Determiner ::
    { DT | PRP } ;;
Adjective ::
    { JJ | JJR | JJS } ;;
Adverb ::
    { RB | RBR | RBS } ;;
Conjunction ::
    { CC } ;;
Preposition ::
    { PREP | TO | IN } ;;
```

4.3 Syntactic Analysis Module

This module is responsible for determining noun phrases, question phrases and relation phrases between noun phrases or noun phrases and question phrases.

Concepts and entities are normally expressed in noun phrases. Therefore, it is important that we can reliably identify noun phrases in order to generate the *query-tuple*. Hence we build the script called *NounPhrase* (such as a sample script below) to specify patterns of noun phrases by utilizing scripts generated from the preprocessing module.

```
Composite ::
    { <Noun> | <Conjunction> | <Adjective>
    | <Adverb> <Adjective> } ;;
NounPhrase ::
    { <Noun> | <Determiner> <Noun>
    | <Composite> <Noun>
    | <Determiner> <Composite> <Noun> } ;;
```

For example, given the following question: *"which projects are about ontologies and the semantic web?"*, Three noun phrases *'projects'"*, *"ontologies"* and *"the semantic web"* will be identified.

In addition, we identify the question words, such as $HowWhy_{cause \mid method}$, $YesNo_{true\ or\ false}$, $What_{something}$, $When_{time \mid date}$, $Where_{location}$, $Many_{number}$, Who_{person}, to provide question classes. Accordingly, question phrases are detected by using noun phrases and question words to give information about question categories. Following VnQAS [7], we define the following question categories: *HowWhy, YesNo, What, When, Where, Who, Many, ManyClass, List* and *Entity*. This can be achieved by using the scripts such as the following:

```
Entity ::
    { <Wh_determiner> <Noun> } ;;
ManyClass ::
    { <Wh_adverb> <Many> <Noun> } ;;
```

For example, in the question: *"How many persons work on AKT project?"*, the phrase *"How many persons"* is identified by using the *ManyClass* script shown above.

The next step is to identify the relation phrases or semantic constraints between noun phrases or noun phrases and question phrases. We create a *Relation* script shown in the following sample script which is used to match relation phrases:

```
Noun_Adj ::
    { NN | NNS | NNP | NNPS | NP | NPS
    | CD | PRP | JJ | JJR | JJS } ;;
Relation ::
    { <Verb> | <Verb> <Noun_Adj> <Preposition>
    | <Verb> <Adverb> <Noun_Adj> <Preposition>};;
```

For instance, with the following question: *"who are the researchers in semantic web research area?"*, the phrase *"are the researchers in"* is the relation phrase detected by using the *Relation* script linking the question phrase *"who"* and the noun phrase *"semantic web research area"*.

Based on the scripts described above, we can determine noun phrases, relation phrases and question phrases of a user's question. In the next section, we describe in details the rules used to directly produce the intermediate representation of a question.

4.4 Semantic Analysis Module

The semantic analysis module identifies the *question-structure* and produces the *query-tuple*s as the intermediate representation *(question-structure, question-class,* $Term_1$*, Relation,* $Term_2$*,* $Term_3$ *)* of the input question by using the scripts generated by the previous modules. We define the following question structures: *Normal, UnknTerm, UnknRel, Definition, Compare, ThreeTerm, And, Or, Clause, Combine, Affirm, Affirm_3Term and Affirm_MoreTuples* [7].

Existing scripts of *NounPhrase* and *Relation* are potential candidates for terms and relations respectively, while question phrases are used to detect question classes. We directly specify the rule's response expression to return the output consisting of a question structure and query-tupples.

For example, the following two rules:

```
<ManyClass> <Relation> <NounPhrase> ==>
    [ Normal, (Normal, ManyClass, ^1, ^2, ^3, ?) ]
<Entity> <Relation> <NounPhrase> <AND> <NounPhrase> ==>
    [ And, (Normal, Entity, ^1, ^2, ^3, ?), (Normal, Entity, ^1, ^2, ^5, ?) ]
```

are used to process the following two input questions resspectively where **?** represents a missed element in the tuples:

"How many subjects are there in the semester?"

[ManyClass *How many subjects*] [Relation *are there in*] [NounPhrase *the semester*]

and

"Which projects are about ontologies and the semantic web?"

[Entity *Which projects*] [Relation *are about*] [NounPhrase *ontologies*] [AND *and*] [NounPhrase *the semantic web*]

If an input question matches a rule's pattern, the rule's response expression specifies and extracts the corresponding elements in the intermediate representation[1]. For instance, the intermediate representation of the first question has:

question-structure of *Normal*

and *query-tuple (Normal, ManyClass, subjects, there, semester, ?).*

The the above second question has an intermediate representation consisting of:

question-structure of *And*

and two *query-tuple*s, that is, *(Normal, Entity, projects, are, ontologies, ?)* and *(Normal, Entity, projects, are, semantic web, ?).*

A nice feature in FrameScript is that it allows one to specify additional conditions in the response expression, instead of pattern expression, to select the appropriate response. A clear advantage is to group ambiguous cases together as well as conditions to resolve them in a single rule. For example, consider the following rule with conditional response expressions:

```
<What> <Relation> <NounPhrase> ==>
      [ ^(^2 == is or ^2 == are) ->
            Definition, (Definition, What, ?, ?, ^3, ?)
      | UnknTerm, (UnknTerm, What, ?, ^2, ^3, ?) ]
```

Using this rule, the intermediate representation of the question:

"what is the role of the academic regulation?"

has *question-structure* of *UnknTerm* and *query-tuple (UnknTerm, What, ?, role, academic regulation, ?).* However, also using this rule, the question:

"what is the standard program?"

has an intermediate representation containing the *question-structure* of *Definition* and tuple *(Definition, What, ?, ?, standard program, ?).*

Actually, creating the rules manually via three modules represented above is a language independent process as it is straightforward to adapt to a new domain and a new language.

5 Experiments

We take 170 English question examples of AquaLog's corpus[2] to build a set of 52 rules, which consumed about 12 hours of actual time to create all rules. Table 1

[1] Stopwords are removed.

[2] `http://technologies.kmi.open.ac.uk/aqualog/examples.html`

shows the number of rules for each question-structure type. A point worth noting is that in 3 rules for *question-structure* of *UnknTerm*, there are 2 rules having conditional response expressions to resolve ambiguity between *UnknTerm* and *Definition*. These rules can be considered as composite rules and they can reveal ambiguity between different types of question structures. Table 2 presents a list of pairs of ambiguous question structure types together with the number of rules to resolve the ambiguity.

Table 1. Number of rules corresponding with each question-structure type

Question-structure type	Number of rules
Definition	1
UnknTerm	3
UnknRel	4
Normal	8
Affirm	6
ThreeTerm	10
And	13
Or	1
Clause	6

Table 2. Number of rules with conditional responses

Question-structure type	Question-structure type	Number of rules
UnknTerm	Definition	2
ThreeTerm	Normal	1
ThreeTerm	UnknTerm	1
Combine	Normal	1

As the intermediate representation of our system is different to AquaLog and there is no common test set available, it is difficult to directly compare our approach with Aqualog on the English domain.

To demonstrate that our approach could be applied to a new open domain, we use the above 52 rules to test the data of 500 questions[3] from TREC 10. Table 3 shows the number of correctly analyzed questions corresponding with each question-structure type.

Table 4 gives the sources of errors for 259 incorrect cases. It clearly shows that most errors come from unexpected structures. This could be rectified by adding more rules, especially when we construct a larger variety of question structure types from a bigger training data such as 5500 questions [4].

This experiment is indicative of the ability in using our system to quickly build rules for a new domain. We believe that our approach could be applied to

[3] `http://cogcomp.cs.illinois.edu/Data/QA/QC/TREC_10.label`

Table 3. Number of questions corresponding with each question-structure type

Question-structure type	Number of questions
Definition	130
UnknTerm	66
UnknRel	4
Normal	20
ThreeTerm	15
And	6

Table 4. Error results

Reason	Number of questions
Have special characters (such as / – “ ” **'s**) and abbreviations	64
Not have compatible patterns	185
Semantic error in elements of the intermediate representation	10

a new language because creating the rules manually for question analysis is a language independent process.

6 Conclusion

In this paper, we introduced a rule-based approach for converting a natural language question into an intermediate representation in a question answering system. Our system utilizes FrameScript to help users to create intuitive and compact rules for extracting elements of the intermediate representation. We constructed rules including patterns and associated responses, in which pattern is used to match user' questions and its corresponding response as output is sent to return the intermediate representation of question. Experimental results of our system on a wide range of questions are promising with reasonable performance. We believe our approach can be applied to question answering in open domain against text corpora that requires an analysis to turn an input question to an explicit representation of some sort. Our method could be combined nicely with the processing of annotating corpus and it is straightforward to apply for a new domain and a new language.

In the future, we will extend our system to assist the rule creation process on a wide range of questions in open domain and to improve the generalization capability of existing rules.

Acknowledgements. This work is partially supported by the Research Grant from Vietnam National University, Hanoi No. QG.10.23.

The authors would like to acknowledge Vietnam National Foundation for Science and Technology Development (NAFOSTED) for their financial support to present the work at the conference.

References

1. Androutsopoulos, I., Ritchie, G., Thanisch, P.: Natural Language Interfaces to Databases - An Introduction. Natural Language Engineering 1, 29–81 (1995)
2. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 168–175 (2002)
3. Hirschman, L., Gaizauskas, R.: Natural Language Question Answering: The View from here. Natural Language Engineering 7, 275–300 (2001)
4. Li, X., Roth, D.: Learning Question Classifiers. In: Proceedings of the 19th International Conference on Computational Linguistics, COLING 2002, vol. 1, pp. 1–7 (2002)
5. Lopez, V., Uren, V., Motta, E., Pasin, M.: AquaLog: An Ontology-Driven Question Answering System for Organizational Semantic Intranets. Web Semantics: Science, Services and Agents on the World Wide Web 5, 72–105 (2007)
6. McGill, M., Sammut, C., Westendorp, J., Kadous, W.: Framescript: A Multi-modal Scripting Language. In: The School of Computer Science and Engineering, UNSW (Copyright ©2003-2008)
7. Nguyen, D.Q., Nguyen, D.Q., Pham, S.B.: A vietnamese question answering system. In: Proceedings of the 2009 International Conference on Knowledge and Systems Engineering, pp. 26–32 (2009)
8. Nguyen, D.Q., Nguyen, D.Q., Pham, S.B.: A Vietnamese Text-Based Conversational Agent. In: Jiang, H., et al. (eds.) IEA/AIE 2012. LNCS (LNAI), vol. 7345, pp. 699–708. Springer, Heidelberg (2012)
9. Nguyen, D.Q., Nguyen, D.Q., Pham, S.B.: Systematic Knowledge Acquisition for Question Analysis. In: Proceedings of Recent Advances in Natural Language Processing (RANLP 2011), pp. 406–412 (2011)
10. Popescu, A.M., Etzioni, O., Kautz, H.: Towards A Theory of Natural Language Interfaces to Databases. In: Proceedings of the 8th International Conference on Intelligent User Interfaces, IUI 2003, pp. 149–157 (2003)
11. Sammut, C.: Managing Context in A Conversational Agent. Electronic Transactions on Artificial Intelligence 5, 189–202 (2001)
12. Stratica, N., Kosseim, L., Desai, B.C.: NLIDB Templates for Semantic Parsing. In: Proceedings of the 8th International Conference on Applications of Natural Language to Information Systems, pp. 235–241 (2003)
13. Templeton, M., Burger, J.: Problems in Natural-Language Interface to DBMS with Examples from EUFID. In: Proceedings of the First Conference on Applied Natural Language Processing, pp. 3–16 (1983)
14. Waltz, D.L.: An English Language Question Answering System for A Large Relational Database. Communications of the ACM 21, 526–539 (1978)

Solving Periodic Event Scheduling Problems with SAT

Peter Großmann[1,2], Steffen Hölldobler[1], Norbert Manthey[1], Karl Nachtigall[2], Jens Opitz[2], and Peter Steinke[1]

[1] Faculty of Computer Science, TU Dresden
[2] Faculty of Transport and Traffic Science, TU Dresden
pg@iccl.tu-dresden.de, peter.steinke@tu-dresden.de

Abstract. In this paper, periodic event scheduling problems (PESP) are encoded as satisfiability problems (SAT) and solved by a state-of-the-art SAT solver. Two encodings, based on direct and order encoded domains, are presented. An experimental evaluation suggests that the SAT-based approach using order encoding outperforms constraint-based PESP solvers, which until now were considered to be the best solvers for PESP. This opens the possibility to model significantly larger real-world problems.

1 Introduction

Many real-world applications like computing a time table for a given railway network or setting up a traffic light system for a particular city are based on periodic events and constraints imposed on these events. Events and their constraints can be modelled by so-called periodic event networks. The periodic event scheduling problem (PESP) consists of such a network and is the question, whether all the events can be scheduled such that a set of constraints – specified by the network – is satisfied. The problem is $\mathcal{NP}$-complete [9] and the currently best solutions are obtained by constraint-based solvers notably PESPSOLVE [8,10] or by LP solvers, which solve linearized PESP instances by introducing modulo parameters [6]. However, these solvers are still quite limited in the size of the problem which they can tackle. For example, PESPSOLVE is able to schedule the inter city express trains, but cannot schedule all passenger trains in Germany.

In recent years, the performance of SAT solvers has been significantly increased and SAT solvers are now applied in real-world settings such as hardware and software verification, planning, termination analysis, or bioinformatics [1]. Hence, it was a natural question to investigate the performance of state-of-the-art SAT solvers on periodic event scheduling problems.

To this end, PESP instances have to be encoded as SAT instances. After some preliminaries presented in Section 2, two encodings are discussed in Section 3: a naive, direct encoding and a more advanced, order encoding. For both encodings, soundness and completeness can be shown. In Section 4, several real-world benchmark problems are presented to the state-of-the-art PESP solver

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 166–175, 2012.

PESPSOLVE and to the state-of-the-art SAT solver RISS [5] using both encodings. The results indicate, that the SAT approach based on order encoding is by far superior to the PESP solver. The paper is concluded in Section 5 by a short discussion and an outline of future work.

2 Notations and Preliminaries

2.1 Satisfiability Problem

A *satisfiability problem* (SAT) consists of a propositional formula F and is the question whether there exists an interpretation (or assignment) J from the set of propositional formulas to the set $\{\top, \bot\}$ of truth values such that J assigns $\top$ to F. In such a case, J is called *model* for F ($J \models F$) and F is said to be *satisfiable*. SAT is $\mathcal{NP}$-complete [2].

It is well-known, that each propositional formula can be transformed into a semantically equivalent formula in conjunctive normal form (CNF), where a formula is in CNF if it is a conjunction of clauses, a clause is a disjunction of literals, and a literal is either an atom or the negation of an atom. Most modern SAT solvers accept SAT instances in CNF. For more details about SAT see [1].

2.2 Periodic Event Scheduling Problem

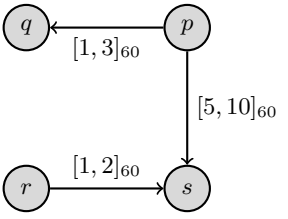
Let $a, b \in \mathbb{Z}$ and $t \in \mathbb{N}$. $[a, b] := \{x \in \mathbb{Z} \mid a \leq x \leq b\}$ denotes the interval from a to b and $[a, b]_t := \bigcup_{z \in \mathbb{Z}} [a + z \cdot t, b + z \cdot t] \subseteq \mathbb{Z}$ the interval from a to b modulo t.

Let $(\mathcal{V}, \mathcal{E})$ be a directed graph, $t \in \mathbb{N}$, and a a mapping, which assigns to each edge $(i, j) \in \mathcal{E}$ a set of intervals modulo t. $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$ is called *periodic event network* (PEN) with t being its period, $\mathcal{V}$ its set of (periodic) events, and $a(i, j)$ its set of constraints for each edge $(i, j) \in \mathcal{E}$. Let $\mathcal{N}$ be a PEN. The function Π, called *schedule*, assigns to each event in $\mathcal{N}$ an integer called the event's potential. In the sequel, let $\mathcal{N}$ be a PEN and Π a schedule for $\mathcal{N}$.

Let $[l, u]_t$ be a constraint in $a(i, j)$ for some edge (i, j) in $\mathcal{N}$. This constraint specifies a time consuming process in that the time between event i and j must be in $[l, u]_t$. Formally, $[l, u]_t$ *holds under* Π iff $\Pi(j) - \Pi(i) \in [l, u]_t$. A schedule Π is *valid* for $\mathcal{N}$ ($\Pi \models \mathcal{N}$) iff all constraints occurring in $\mathcal{N}$ hold under Π.

Two schedules Π and Φ for $\mathcal{N}$ are said to be *equivalent* ($\Pi \equiv \Phi$) iff for all events $i \in \mathcal{V}$ we find, that $\Pi(i) \bmod t = \Phi(i) \bmod t$. $\equiv$ is an equivalence relation. Furthermore, if Π is a valid schedule for $\mathcal{N}$ and $\Pi \equiv \Phi$, then Φ is also a valid schedule for $\mathcal{N}$. This implies, that for each valid schedule Φ for $\mathcal{N}$ there exists an equivalent schedule Π for $\mathcal{N}$, such that Π is valid and for all events $i \in \mathcal{V}$ we find $\Pi(i) \in [0, t-1]$. For the rest of this work we only consider schedules in this interval.

The *periodic event scheduling problem* (PESP) consists of a PEN and is the question whether there exists a valid schedule. Figure 1 shows a simple PEN,



Let $(\{p, q, r, s\}, \mathcal{E}, a, 60)$ be a PEN with

$$\mathcal{E} = \{(p, q), (p, s), (r, s)\},$$
$$a(p, q) = \{[1, 3]_{60}\},$$
$$a(p, s) = \{[5, 10]_{60}\},$$
$$a(r, s) = \{[1, 2]_{60}\}.$$

Fig. 1. A periodic event network with 4 events and 3 constraints

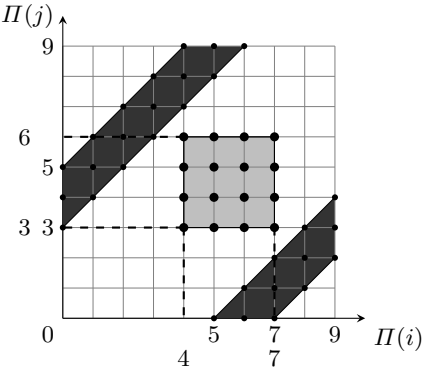


Fig. 2. Feasible (dark grey) and infeasible (light grey) regions for the constraint $[3, 5]_{10} \in a(i, j)$. The light grey square shows a particular infeasible rectangle.

which can be interpreted as two trains, who meet at a station every 60 minutes, where the time between arrival (p) and departure (q) of the first train must be between 1 and 3 minutes, the time between arrival (r) and departure (s) of the second train must be between 1 and 2 minutes, and there must be between 5 and 10 minutes time between the arrival of the first train and the departure of the second train. The schedule $\Phi = \{p \mapsto 144, q \mapsto 147, r \mapsto 148, s \mapsto 150\}$ for the PEN in Fig. 1 is valid because

$$\Pi(q) - \Pi(p) = 147 - 144 = 3 \in [1, 3]_{60},$$
$$\Pi(s) - \Pi(p) = 150 - 144 = 6 \in [5, 10]_{60},$$
$$\Pi(s) - \Pi(r) = 150 - 148 = 2 \in [1, 2]_{60}.$$

It is equivalent to the schedule $\Pi = \{p \mapsto 24, q \mapsto 27, r \mapsto 28, s \mapsto 30\}$.

As another example consider a constraint $[3, 5]_{10} \in a(i, j)$, where (i, j) is an edge in a PEN $\mathcal{N}$ and let Π be a schedule for $\mathcal{N}$. If $\Pi(i) = 6$ and $\Pi(j) = 1$, then the constraint holds under Π as $1 - 6 = -5 \in [3, 5]_{10}$. If $\Pi(i) = \Pi(j) = 5$, then this constraint does not hold under Π as $5 - 5 = 0 \notin [3, 5]_{10}$. In Fig. 2 all pairs of values for the events i and j within the interval $[0, 9]$ are shown.

For more details about the definition of the PESP see [10,4].

3 Encoding PESP as SAT

This section describes how a PESP can be encoded as a SAT problem. Similarly as converting constraint satisfaction problems into SAT, the finite domains of the events can be translated in very different ways. Due to the lack of space, we present in this work only the *order encoding* [11] in details. At the end of this section we will also present a short draft for the *direct encoding*.

Because domains in PESP are subsets of $\mathbb{Z}$, especially they are sets of intervals, it is natural to apply the order relation $\leq$ on $\mathbb{Z}$. As proposed in Section 2.2, we will only search for schedules, which map to $[0, t-1]$. For the proofs of the lemmas presented in this section we refer to [4].

Let $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$ be a PEN and $q_{n,i}$ is a propositional variable with $i \in [-1, t-1]$, $n \in \mathcal{V}$, and Π be a schedule. Then the variable $q_{n,i}$ is interpreted as $\Pi(n) \leq i$ and $\neg q_{n,i}$ as $\neg(\Pi(n) \leq i)$ which is equivalent to $\Pi(n) \geq i+1$. The function *enc* maps the set of events $\mathcal{V}$ to a propositional formula in CNF, such that this ordering holds:

$$enc : n \mapsto (\neg q_{n,-1} \wedge q_{n,t-1}) \bigwedge_{i \in [0,t-1]} (\neg q_{n,i-1} \vee q_{n,i})$$

Hence we encode that $\Pi(n) \leq i$ implies $\Pi(n) \leq i+1$ for $-1 \leq i < t$. Recall that we only consider schedules in the interval $[0, t-1]$, such that $\Pi(n) \geq 0$ and $\Pi(n) \leq t-1$ always holds. To encode the potential of all events of $\mathcal{N}$, we define

$$\Omega_{\mathcal{N}} := \bigwedge_{n \in \mathcal{V}} enc(n). \tag{1}$$

To encode a constraint $c = [l, u]_t \in a(i, j)$ we take a deeper look at all feasible pairs $(\Pi(i), \Pi(j))$, that hold under c. We call the union of all these pairs the *feasible region* and the union of every other pair the *infeasible region*. Let $P_c := \{(\Pi(i), \Pi(j)) \mid \Pi(j) - \Pi(i) \notin c\}$ be the set of pairs, which define the infeasible region of a constraint $c \in a(i, j)$. Furthermore, let $S_c := [0, t-1] \times [0, t-1] \setminus P_c$ be the set of feasible pairs of c, which define the feasible region. Figure 2 shows the feasible and infeasible regions for the constraint $[3, 5]_{10}$.

Example 1. Let $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$ be a PEN with $i, j \in \mathcal{V}$ and $c = [3, 5]_{10} \in a(i, j)$. The rectangle $r = ([4, 7] \times [3, 6])$ is part of the infeasible region P_c, hence we know for a valid schedule Π that $(\Pi(i), \Pi(j)) \notin r$. In other words, for each pair $(p, q) \in r$ there exists no valid schedule Π with $\Pi(i) = p$ and $\Pi(j) = q$. The rectangle r is visualized in Fig. 2. We can derive the following propositional formula:

$$\begin{aligned}
& \neg((\Pi(i) \leq 7) \wedge (\Pi(i) \geq 4) \wedge (\Pi(j) \leq 6) \wedge (\Pi(j) \geq 3)) \\
\Leftrightarrow\ & \neg((\Pi(i) \leq 7) \wedge (\neg(\Pi(i) \leq 3)) \wedge (\Pi(j) \leq 6) \wedge (\neg(\Pi(j) \leq 2))) \\
\Leftrightarrow\ & \neg(q_{i,7} \wedge \neg q_{i,3} \wedge q_{j,6} \wedge \neg q_{j,2}) \\
=\ & [\neg q_{i,7}, q_{i,3}, \neg q_{j,6}, q_{j,2}] =: F
\end{aligned}$$

Since F is a clause, we can directly add this to the formula $\Omega_{\mathcal{N}}$.

Formally, if $c \in a(i,j)$ is a constraint, then we can encode a rectangle $[i_1, i_2] \times [j_1, j_2]$ that is a subset of the infeasible region P_c with a single clause:

$$enc\_rec([i_1, i_2] \times [j_1, j_2]) = [\neg q_{i,i_2}, q_{i,i_1-1}, \neg q_{j,j_2}, q_{j,j_1-1}]$$

To define a rectangle for the feasible region we would need more than a single clause, hence, we decided to encode a constraint by defining the infeasible region with rectangles to keep the number of clauses low. Empirical results indicate, that a lower number of clauses typically results in a lower run time behavior of a SAT solver [3].

To describe the whole infeasible region, we need some helper functions. These helper functions will be combined into a function, which calculates as few rectangles as possible to cover the whole infeasible region of a constraint. Our approach at first minimizes the number of clauses and then maximizes the space, which is covered by the corresponding rectangles. Note that other coverings are also possible, e.g. minimum number of clauses with a minimum number of space covering. Since this is a heuristically chosen approach, we leave further empirical investigations open for future work.

Figure 2 shows the two feasible regions. Each of them has a lower and an upper bound. In order to describe the infeasible region in between, we need the next lower bound $l = 3$, which represents the upper bound of the infeasible region and the previous upper bound $u = -5$, which represents the lower bound of the infeasible region with respect to this example. It can be directly concluded, that $u < l$.

With $\lfloor \cdot \rfloor$ being the round down function, $\lceil \cdot \rceil$ being the round up function and the two integers $u, l \in \mathbb{Z}$ with $u < l$, we can define

$$\begin{aligned}
\delta &\colon \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z},\ \delta(l,u) = l - u - 1 \\
\delta y &\colon \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z},\ \delta y(l,u) = \left\lfloor \frac{\delta(l,u)}{2} \right\rfloor \\
\delta x &\colon \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z},\ \delta x(l,u) = \left\lceil \frac{\delta(l,u)}{2} \right\rceil - 1.
\end{aligned}$$

With these definitions, we are able to determine a rectangle between u and l, such that it has maximum area having minimum perimeter. Basically each rectangle has a width of $\delta x(l,u)$ and a height of $\delta y(l,u)$.

In order to cover each infeasible pair in the area between u and l, we need approximately t rectangles. The function $\zeta : \mathcal{P}(\mathbb{Z}) \to \mathcal{P}(\mathcal{P}(\mathbb{Z}) \times \mathcal{P}(\mathbb{Z}))$,

$$\zeta([l,u]_t) = \{H \times G \in \mathcal{P}(\mathbb{Z})^2 \mid |H| = \delta x(l,u), |G| = \delta y(l,u), (H \times G) \cap S_c = \emptyset\},$$

with $H, G \in \mathcal{P}(\mathbb{Z})$ being intervals, maps to the set of all infeasible rectangles of constraint $c = [l,u]_t$. The sufficiency, that all infeasible pairs of a constraint are excluded, is given by the following lemma.

Lemma 1. *Let $c = [l, u]_t$ be a constraint. Then the following holds*

$$\textit{(i)}\quad P_c \subseteq \bigcup_{A \in \zeta(c)} A\ , \qquad\qquad \textit{(ii)}\quad S_c \cap \bigcup_{A \in \zeta(c)} A = \emptyset.$$

Firstly, Lemma 1 shows, that encoding the function $\zeta(c)$ excludes all infeasible pairs in the modulo interval and, secondly, that there is no intersection between the feasible and the excluded region.

With $\Psi_{\mathcal{N}}$ being the encoding of all edges $e \in \mathcal{E}$, such that

$$\Psi_{\mathcal{N}} := \bigwedge_{e \in \mathcal{E}} \bigwedge_{c \in a(e)} \bigwedge_{A \in \zeta(c)} enc\_rec(A), \tag{2}$$

we can encode a PESP instance with

$$enc\_pesp(\mathcal{N}) = (\Omega_{\mathcal{N}} \wedge \Psi_{\mathcal{N}}) \tag{3}$$

After a model J has been found for this formula, extracting the value of the event n with the domain $[0, t-1]$ from J is done by the function $\xi_n(J)$, where $\xi_n(J) = k$ such that $J \not\models q_{n,k-1} \wedge J \models q_{n,k}$ and $k \in [0, t-1]$. ξ_n is well defined due to the following lemma.

Lemma 2. *Let $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$ be a PEN, $n \in \mathcal{V}$ be an event and J an interpretation. Then*

$$J \models enc(n) \Leftrightarrow \exists! k \in [0, t-1] : \forall i \in [-1, k-1] : J \not\models q_{n,i} \text{ and } \forall j \in [k, t-1] : J \models q_{n,j}$$

Now we can extract the schedule Π on a per-element basis from a model J for $\Omega_{\mathcal{N}}$ by

$$\forall n \in \mathcal{V} : \Pi(n) = \xi_n(J). \tag{4}$$

Lemma 3. *Let $r \subseteq P_c$ be a rectangle in the infeasible region of the constraint $c \in a(i, j)$. Then $J \models enc\_rec(r) \Leftrightarrow (\xi_i(J), \xi_j(J)) \notin r$ with J being an interpretation.*

Lemma 3 states that a model for an encoded constraint rectangle r is only satisfied by interpretations that do not assign values to the participating events withing this rectangle.

Theorem 1 (Soundness and Completeness).
Let $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$ be a PEN and $F := enc\_pesp(\mathcal{N})$ be the order encoded propositional formula of $\mathcal{N}$. Then $\exists J : J \models F \Leftrightarrow \exists \Pi : \Pi \models \mathcal{N}$ with J being an interpretation and Π being a schedule of $\mathcal{N}$.

Proof.

$$
\begin{aligned}
& \exists J : J \models F \\
\Leftrightarrow\ & \exists J : J \models (\Omega_{\mathcal{N}} \wedge \Psi_{\mathcal{N}}) && (3) \\
\Leftrightarrow\ & \exists J : J \models \Omega_{\mathcal{N}} \wedge J \models \Psi_{\mathcal{N}} \\
\Leftrightarrow\ & \exists J : J \models \bigwedge_{n \in \mathcal{V}} enc(n) \wedge J \models \Psi_{\mathcal{N}} && (1) \\
\Leftrightarrow\ & \exists J : \forall n \in \mathcal{V} : \Pi(n) := \xi_n(J) \wedge J \models \Psi_{\mathcal{N}} && (4), \text{Lemma 2} \\
\Leftrightarrow\ & \exists J : \forall n \in \mathcal{V} : \Pi(n) := \xi_n(J) && (2) \\
& \qquad \wedge J \models \bigwedge_{e \in \mathcal{E}} \bigwedge_{c \in a(e)} \bigwedge_{A \in \zeta(c)} enc\_rec(A) \\
\Leftrightarrow\ & \exists J : \forall n \in \mathcal{V} : \Pi(n) := \xi_n(J) \\
& \qquad \wedge \forall e \in \mathcal{E}\ \forall c \in a(e)\ \forall A \in \zeta(c) : J \models enc\_rec(A) \\
\Leftrightarrow\ & \exists J : \forall n \in \mathcal{V} : \Pi(n) := \xi_n(J) \wedge \forall e \in \mathcal{E} : e \text{ holds under } \Pi && \text{Lemmas 3\&1} \\
\Leftrightarrow\ & \exists J : \forall n \in \mathcal{V} : \Pi(n) := \xi_n(J) \wedge \Pi \models \mathcal{N} \\
\Leftrightarrow\ & \exists \Pi : \Pi \models \mathcal{N} && (4), \text{Lemma 2}
\end{aligned}
$$

□

Direct Encoding. Similarly to the presented encoding, the direct encoding approach can be used for an event $n \in \mathcal{V}$, with the propositional variable $q_{n,i}$ representing the value of the event n being exactly i (i.e. $\Pi(n) = i$). This way, the value extraction of the event from an interpretation is straightforward. Encoding a node is then done by the function

$$
enc : n \mapsto (q_{n,0} \vee \cdots \vee q_{n,t-1}) \bigwedge_{i \in [0,t-2], j \in [i+1,t-1]} (\neg q_{n,i} \vee \neg q_{n,j})
$$

which leads to a quadratic number of clauses with respect to the domain size.

The order encoding approach tries to exclude all pairs $(\Pi(i), \Pi(j))$ for events i, j, that are part of the infeasible region P_c. Likewise, the direct encoding will follow this method, such that for each pair $(\Pi(i), \Pi(j)) \in P_c$ a clause can be encoded with $\neg(\Pi(i) \wedge \Pi(j)) \equiv [\neg q_{i,\Pi(i)}, \neg q_{j,\Pi(j)}]$. Consequently, the direct encoding needs a quadratic number of clauses. An interpretation J that models this encoding is guaranteed to not assign the two events i and j conflicting values. As for the order encoding, encoding all constraints is done by connecting the encoding for all events and all constraints conjunctively. It is shown in [4] that using the direct encoding is also sound and complete.

4 Results

Until this point, it has been shown that it is possible to reduce PESP to SAT. In this section, we show that the SAT based approach can outperform the currently used native domain solver by up to four orders of magnitude.

Table 1. PESP instances and corresponding encodings

	PESP $\mathcal{N} = (\mathcal{V}, \mathcal{E}, a, t)$		direct encoding F		order encoding G	
instance	$\lvert\mathcal{V}\rvert$	$\sum_{e\in\mathcal{E}} \lvert a(e)\rvert$	$\lvert vars(F)\rvert$	$\lvert F\rvert$	$\lvert vars(G)\rvert$	$\lvert G\rvert$
k	18	730	2 160	544 628	2 142	26 536
swg_2	60	1 145	7 200	2 037 732	7 140	83 740
fernsym	128	3 117	15 360	6 657 955	15 232	353 276
swg_4	170	7 107	20 400	6 193 570	20 230	399 191
swg_3	180	2 998	21 600	4 874 144	21 420	214 011
mb11mg	200	8 298	23 280	8 864 714	23 086	646 870
b	219	5 571	22 320	5 448 546	22 134	287 641
swg_1	221	7 443	26 520	7 601 906	26 299	462 217
mb13mg	231	9 805	27 000	10 473 825	26 775	777 894
save	277	1 535	21 000	6 776 095	20 825	229 542
seg_2	611	9 863	73 320	25 101 341	72 709	1 115 210
we	846	14 690	78 480	28 107 997	78 659	2 049 188
seg_1	1 483	10 351	177 960	34 323 942	176 477	1 348 045

The PESP instances, which are used for the comparison, model public railway transport networks of up to 400 trains, which are instances like the whole inter city network of Germany (*fernsym*), as well as subnetworks like south west ($swg_i, i \in \{1, \ldots, 4\}$, *mb11mg*, *mb13mg*) and south east (seg_1, seg_2, *we*) Germany, respectively. The periods are set to two hours. The number of constraints ranges from 1 000 to 15 000. Table 1 shows the sizes of each PESP instance, as well as the resulting formula size after reducing the problem to SAT by using the direct and order encoding, respectively.

The native domain solver for solving PESP instances is described and analyzed in [10] and [8]. The authors claim that this solver is the best known algorithm to solve PESP instances. The solver is equipped with a decision tree method with respect to the events' potentials. Additionally, constraint propagation techniques are employed by propagating the current valid assignments of the events across the network. Once the propagation detects a potential with an empty domain, a conflict is found. In this case, the tree search will backtrack. This solver will be referred to as PESPSOLVE.

The comparison has been performed on an Intel Xeon processor with 3 GHz and 8 GB of main memory. The runtime timeout has been set to 24 hours. As SAT solver, we have chosen RISS, presented in [5], for several reasons: Firstly, it implements state-of-the-art SAT technology in a modular way. Secondly, the solver provides a huge set of parameters, that can be tuned, and thirdly, it uses a state-of-the-art SAT preprocessor [7]. Table 2 shows that the SAT approach outperforms the native domain solver by far, although in this work the parameters of the solver have not been adjusted to the specific application domain. The results clearly show, that the SAT-based approaches perform much better than PESPSOLVE for both encodings. Furthermore, it can be seen, that the order encoding is a better choice among the two encodings, because the number of clauses is significantly smaller for bigger instances, in fact there are around

Table 2. PESP instances and corresponding solving times in seconds

instance	PESPSOLVE	*direct* + RISS	*ordered* + RISS
k	47 112	timeout	26
swg_2	512	37	2
fernsym	2 035	294	7
swg_4	912	752	8
swg_3	66	50	2
mb11mg	timeout	timeout	110
b	timeout	37 112	145
swg_1	timeout	18	7
mb13mg	timeout	timeout	986
save	timeout	38 411	10
seg_2	timeout	timeout	11
we	timeout	timeout	timeout
seg_1	timeout	16	10

period times as many clauses. The runtime per PESP instance for the SAT approach is the sum of the reduction time and the solving time. Comparing the run times of PESPSOLVE and the SAT approach always results in a significant speedup. The value of the speedup increases up to at least 12 342 (instance swg_1). In fact, this value is even higher, since PESPSOLVE was not able to solve this network within the given timeout. Concerning the number of instances, that can be solved within the timeout, the SAT based approach also shows a higher performance than PESPSOLVE. By using the order encoding approach, all except one of the tested networks can be solved.

5 Conclusion

In this work it is shown that the periodic event scheduling problem can be reduced to satisfiability testing efficiently. Solving the SAT instance with a state-of-the-art SAT solver requires a very short time frame compared to a state-of-the-art PESP solver. The reduction to SAT allows to tackle a whole set of larger instances, which cold not be solved before. Two encodings have been analyzed for the reduction, namely the direct encoding and the order encoding. The appropriate encoding seems to be the later one, because the speedup that has been measured is four orders of magnitude compared to the PESP solver.

To improve the power of the reduction even further, preprocessing could be applied to the periodic event network. Passing more information into the SAT instance could be done by propagating information to neighboring events and thus cutting off parts of the search space. In addition, redundant constraints could be removed from the network before it is encoded. Both approaches might help the SAT solver to solve instances faster.

Based on the presented method scaling the PESP instances becomes more interesting. From the used application railway networks the largest possible

instance is to combine all subnetworks of Germany to a single network. Solving this instance in reasonable time is a huge open goal in transport engineering.

Another interesting part, the optimization of timetables, which has been already discussed a lot in [10] could be supported by a fast feasibility checking as presented in this work. The optimization objective restricts the bounds on several nodes. By using an adapted branch and bound algorithm and the presented reduction to SAT, also optimizing networks might become more powerful than using a specialized PESP optimization tool. Such a method could be improved further by using incremental SAT solving. The comparison with respect to optimization is future work.

The final conclusion that can be drawn from the presented work is the following: the currently best PESP solver is now SAT based.

References

1. Biere, A., Heule, M., van Maaren, H., Walsh, T. (eds.): Handbook of Satisfiability. IOS Press (2009)
2. Cook, S.A.: The Complexity of Theorem-Proving Procedures. In: Harrison, M.A., Banerji, R.B., Ullman, J.D. (eds.) STOC, pp. 151–158. ACM (1971)
3. Eén, N., Biere, A.: Effective Preprocessing in SAT Through Variable and Clause Elimination. In: Bacchus, F., Walsh, T. (eds.) SAT 2005. LNCS, vol. 3569, pp. 61–75. Springer, Heidelberg (2005)
4. Großmann, P.: Polynomial Reduction from PESP to SAT. Tech. Rep. 4, Knowledge Representation and Reasoning Group, Technische Universität Dresden, 01062 Dresden, Germany (October 2011)
5. Hölldobler, S., Manthey, N., Saptawijaya, A.: Improving Resource-Unaware SAT Solvers. In: Fermüller, C., Voronkov, A. (eds.) LPAR-17. LNCS, vol. 6397, pp. 519–534. Springer, Heidelberg (2010)
6. Liebchen, C., Möhring, R.H.: The Modeling Power of the Periodic Event Scheduling Problem: Railway Timetables — and Beyond. In: Geraets, F., Kroon, L.G., Schoebel, A., Wagner, D., Zaroliagis, C.D. (eds.) Railway Optimization 2004. LNCS, vol. 4359, pp. 3–40. Springer, Heidelberg (2007)
7. Manthey, N.: Coprocessor – a Standalone SAT Preprocessor (2011), `http://arxiv.org/abs/1108.6208`
8. Nachtigall, K.: Periodic Network Optimization and Fixed Interval Timetable. Habilitation thesis, University Hildesheim (1998)
9. Odijk, M.A.: Construction of periodic timetables, Part 1: A cutting plane algorithm (1994)
10. Opitz, J.: Automatische Erzeugung und Optimierung von Taktfahrplänen in Schienenverkehrsnetzen. Reihe: Logistik, Mobilität und Verkehr, Gabler Verlag | GWV Fachverlage GmbH (2009)
11. Tamura, N., Taga, A., Kitagawa, S., Banbara, M.: Compiling finite linear CSP into SAT. Constraints 14(2), 254–272 (2009)

Improving the Performance of Constructive Multi-Start Search Using Record-Keeping

Dan E. Tamir[1], Charles R. King[1], and Mark McKenney[2]

[1] Texas State University – San Marcos, Department of Computer Science, USA
{Dan.Tamir,Charles.King}@txstate.edu
[2] Southern Illinois University Edwardsville, Department of Computer Science, USA
marmcke@siue.edu

Abstract. State-space search redundancy, that is, multiple explorations of the same state, is an inherent problem in many heuristic search algorithms. It is prevalent in constructive multi-start algorithms. Record-keeping mechanisms, however, can minimize redundancy and enable exploiting time/space tradeoffs. This paper investigates the utility of record-keeping procedures in the context of Iterative Hill Climbing applied to the Traveling Salesperson Problem using several restart mechanisms including Greedy Randomized Adaptive Search, and Greedy Enumeration. Record-keeping methods such as unbounded memory, dedicated memory, and cache memory, as well as a novel "book-keeping" method utilizing a Bloom filter are investigated. Experiments performed using TSPLIB benchmarks and random TSP instances with 100 cities show that under the above mentioned restart and record-keeping mechanisms the IHC produces competitive results. In addition, the research shows that record-keeping, in specific Bloom filters, can considerably improve both the time performance of IHC and the quality of solutions produced.

Keywords: combinatorial optimization, heuristic search, the traveling salesperson problem, iterative hill climbing, Bloom filter.

1 Introduction

The study reported in this paper stems from a multidisciplinary research where multi-start heuristic search algorithms, primarily iterative hill climbing (IHC) [1]; utilizing millions of restarts (i.e., climbers) have been applied to the combinatorial problem of finding a maximal parsimony phylogenetic tree (MPPT) [2]. IHC has been the algorithm of choice for the numerous MPPT due to the enormous combinatorial complexity of the problem. Nevertheless, in order to simplify the investigation, the current research concentrates on the application of IHC to the traveling salesman problem (TSP). The first phase of the current research has evaluated several restart mechanisms and demonstrated that Greedy Randomized Adaptive Search (GRASP), and greedy enumeration (GE) significantly outperform other restart procedures [3- 5].

In this paper, we review the main restart methods, investigate the utility of several record keeping mechanisms, and focus on improving the search space coverage of constructive multi start algorithms using a constant amount of temporal and spatial resources [6]. One of the main contributions of the paper is the introduction of a new

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 176–186, 2012.

memory model for record keeping in heuristic search in the form of a Bloom filter [5]. Additional contributions relate to the thorough analysis of other record keeping mechanisms including a dedicated memory and cache memory.

Experiments performed, and verified with high level of significance and confidence via statistical analysis, using TSPLIB benchmarks, and random TSP instances with up to 100 cities are used to evaluate the utility of record-keeping. The empirical results demonstrate that under the above mentioned restart and record-keeping mechanisms the IHC executed with up-to 100,000 restarts produces competitive results (in most of the cases within 1.2% of OPT). Furthermore, the research shows that record-keeping can considerably improve the time performance of IHC as well as the quality of solutions produced. The main findings of the research are:

1. With 100,000 restarts and sufficient memory, every record-keeping mechanism evaluated is capable of reducing the total running time by 90.8% resulting in an achievable speedup upper-bound determined empirically to average 10.9X.
2. The cache achieves the upper-bound using less than 900K Bytes (900KB).
3. The Bloom filter outperforms the rest of the record-keeping mechanisms and achieves the upper-bound using less than 10K Bytes.
4. Under a fixed number of steps, and with the same amount of memory as in (3) and (4) respectively, both mechanisms achieve the upper-bound of solution quality improvement (0.518%).

2 Background and Literature Review

Consider a graph $G(V, E)$ where V is a set of vertices and E is a set of edges. A *Hamiltonian tour* in G is a cycle that starts from a vertex $v_0 \in V$, traverses every other vertex of G exactly one time, and terminates at v_0 [7,9]. A problem of interest in a weighted Hamiltonian graph $G(V, E, W)$, where W is a set of edge weights, is to find the minimal Hamiltonian tour. The traveling salesman problem (TSP) is equivalent to the problem of seeking a minimal Hamiltonian tour in a weighted graph. The TSP is known to be NP-Complete [7,8]. It is one of the most commonly addressed combinatorial optimization problems and is chosen as the platform of study for this research [7-9]. This paper is only concerned with the symmetric TSP, requiring that the graphs be undirected. In reference [10], however, we consider record keeping in additional combinatorial optimization problems such as feature selection, along with several heuristic search algorithms (e.g., genetic algorithms) for solving these problems.

2.1 Constructive Multi-Start Search Algorithms (CMSSAs)

CMSSAs, such as the IHC, are a class of algorithms commonly applied to combinatorial optimization problems (COPs) [1,6]. Generally, these algorithms generate a valid initial solution to the COP, and then attempt to improve the solution using heuristic techniques. Eventually, a solution that cannot be improved, referred to as a *local optimum*, is reached. In order to avoid getting trapped in a local optimum, the search algorithm restarts with a new valid initial solution. The process of local improvements and reconstruction of initial solutions (restarts) is repeated until a

solution that is sufficiently good is achieved, or some limit on computing resources is reached [6]. Multiple aspects of constructive multi-start search algorithms, including restart procedures, effective heuristics, and mechanisms that traverse a larger portion of the solution search space in a constant amount of time have been explored [4, 10]

2.2 Search Space Redundancy

Depending on the mechanism used for local improvements, redundancies in CMSSA might occur when a starting solution or an intermediate solution (referred to as a configuration) is encountered more than one time. This problem is prevalent in IHC and other heuristics that employ greedy local improvements since multiple explorations of a given configuration leads to the same solution. By recording previously encountered solutions, redundant searches can be reduced or eliminated. Nevertheless, many combinatorial optimization problems including the TSP are NP complete [11]. Hence, recording all encountered possible solutions is impractical. Consequently, record keeping mechanisms that efficiently use a fixed amount of memory are of interest [10]. Formally, redundancy in a heuristic search routine is defined to be the ratio of the total number of states explored to the number of distinct states explored. That is: $Redundancy = \frac{number\ of\ \ states\ explored}{number\ of\ \ distinct\ states}$. The redundancy of a problem can be used to infer an upper bound on the speedup that can be obtained through record keeping. The problem of redundancy in the search space is related to Kolmogorov program complexity [12], and lossless data compression methods such as the Lempel Ziv (LZ) family of algorithms [13].

2.3 Construction Algorithms

IHC can be implemented as a *constructive multi-start search*, in which multiple solutions are computed by constructing new solutions from an initial solution. At each step, the concept of a *neighborhood* is used in which a neighborhood of a solution s is defined as the set of solutions S that are generated by making a minor modification, denoted a *move,* to s [6, 8]. For example, a TSP tour can be adjusted by an operation such as 2-opt which results in exchanging the visitation order of two cities [8]. The IHC algorithm proceeds by choosing the best $s' \in S$ as the next step. The move s is then assigned s' and the process is repeated until no improvements to s can be made. This solution is referred to as a local optimum. In general, the algorithm repeatedly restarts with a new starting configuration until a sufficiently good solution is reached, or a set running time has expired.

In [5] we report on a set of experiments used to select the optimal restart mechanism including Random, Greedy Enumeration (GE) [4], Greedy Jump [4], Greedy Randomized Adaptive Search Procedure (GRASP) [3], Nearest Neighbor [8], and Clark Wright [8]. They are briefly described in the following paragraphs. The experiments show that GRASP and GE are the most suitable restart algorithms. The mean, variance and approximate distribution of the results are compared to optimal results reported in [10], and are very close (within a confidence level of 99%) to the mean variance and distribution of the optimal results. Since GRASP has slightly better redundancy we used it as the subject for the set of experiments reported in the current paper.

Random: This algorithm randomly constructs a starting tour. The starting tours produced may be inferior to those of other construction algorithms, but high-quality solutions may still result after climbing. Moreover, this algorithm is likely to explore portions of the search space left unconsidered by other construction algorithms.

Greedy Enumeration: The greedy enumeration (GE) algorithm selects (enumerates) subsets of the entire set of the edges. The enumeration is done in a consistent and greedy order and in a way that guarantees that the graph induced by the selected subset is Hamiltonian. Given a subset of the edges, the GE algorithm attempts to construct a minimal weight Hamiltonian path by repeatedly adding the shortest available edge, provided it does not introduce a cycle or increase the degree of any vertex to more than two [8]. This continues until a Hamiltonian path is obtained. Finally, the first and last vertices of the path are connected to obtain a Hamiltonian cycle. The GE restarts by considering the next subset in the enumeration.

Greedy Jump: GE constructs tours in a decreasing order of quality and with relatively small difference between consecutive restarts; thus, it is likely that many tours will cluster in the same *basin of attraction*, meaning the local search ultimately arrives at the same local optimum, effectively leaving much of the search space unexplored. The greedy jump algorithm addresses this issue by performing periodic jumps through the search space. The effect of this jump forward in the solution space is to discard preferred solutions in order to escape from a basin of attraction.

GRASP: In contrast to GE, which always makes the greedy choice, GRASP (Greedy Randomized Adaptive Search Procedure) makes a semi-greedy choice by choosing randomly from the shortest n edges [3]. Feo and Resende state that GRASP solutions are significantly better than solutions produced by random construction [3].

Nearest Neighbor: The algorithm begins at an arbitrary vertex and proceeds to the nearest unvisited vertex [8]. The process repeats until the tour is complete.

Clarke-Wright: The Clark-Wright TSP construction algorithm begins by designating one vertex as the central hub and creating a cycle consisting of an "out and back" trip to each other vertex, similar to a delivery truck making multiple deliveries and returning home after each delivery. This cycle is not Hamiltonian, as it visits the hub many times. The algorithm then begins improving the cycle by combining deliveries into fewer trips, based on the *savings* generated. The algorithm enumerates all valid circuit adjustments, chooses the one resulting in the largest savings, and repeats this process until no further improvement is possible.

2.4 Record Keeping in Heuristic Search

Record keeping and construction algorithms have been studied in previous work [4]. Our work differs from the work cited above due to the fact that we provide an in-depth study of the effects of cache configurations, Bloom filter parameters, as well as dedicated and unbounded memory models. It should be noted that record keeping resembles Tabu search where the Tabu list includes a subset of the states encountered so far [14], It differs from Tabu in that the objective of short term Tabu memory (e.g., a Bloom filter) is to explore the solution space more intelligently rather than to

speed the search. The goal of long-term Tabu memory is to enable revisiting explored search states in order to attempt to find a better local optimum [14].

Hertel and Pitassi [15] as well as Allen and Darwiche [16] have studied time/space trade-offs in the context of heuristic search. They found that time requirements could be significantly reduced through record keeping. They referred to their method as caching. Nevertheless, their cache is static; they do not consider cache replacement policies, or cache organization issues. Moreover, their findings do not pertain to CMSSA. Aggarwal investigates the technique of software caching for memory intensive applications that perform searches or sorted insertions [17]. He obtains reductions of up to 30% in computational time due to caching. However, Aggarwal's implementation is not extended to heuristic optimization. Karhi and Tamir demonstrate that iterative hill climbing (IHC) can get better performance by exploiting time/space tradeoffs emerging from saving intermediate results of the search [4]. They did not consider, however, other record keeping mechanisms such as Bloom filter [18]. Santos et al., have investigated cache diversity in genetic algorithms (GA). The cache is used to store partial results of the chromosome evaluation function [19]. Despite referring to their record keeping as cache, the record keeping mechanism does not include provisions for replacement policies and can actually be considered as infinite dedicated memory (unbounded memory). This limits their approach to small problems and to problems where a function computation that is associated with a state can be decomposed and recomposed.

3 Record Keeping Mechanisms

Four memory models are used to analyze redundancy and the potential for minimizing the redundancy in the search space: *Dedicated Memory, Unbounded Memory, Cache Memory,* and *Bloom filter.*

Dedicated Memory - The dedicated memory model assumes that a fixed amount of memory is used as a table that holds all the states encountered so far. The table is flashed and reused, or frozen when it gets full.

Unbounded Memory - The unbounded memory model assumes that there is sufficient memory to store all the distinct tours encountered during the search. Since IHC is performed with a limited number of restarts, only a finite number of tours are generated. Hence, dedicated memory of sufficient size to store every tour generated can serve as a model of "infinite" or unbounded memory. Unbounded memory and dedicated memory are used to establish the upper bound on complexity, attainable speedup, and solution quality improvement when using record keeping. Nevertheless, these mechanisms are not practical for large TSP instances.

Cache - Computer memory systems exploit two empirical principles: "locality of reference" and "small is fast," to implement a memory hierarchy [20]. Generally, under this scheme, the locality of the current central processing unit (CPU) reference word is stored in a small and fast cache. Related to the concepts of locality of reference are the terms of hit and miss and hit/miss rate. Other important concepts of caching relate to cache replacement policies and cache organization [20].

A cache cannot produce a false positive record. That is, if a state is stored (recorded) in the cache; it has already been encountered in a previous stage of the search. Nevertheless, since the cache is finite and due to the use of replacement policies, a cache might produce a false negative result which leads to re-evaluating a previously encountered state. In this sense, the addition of a cache does not result in disregarding any state that would have been explored in a non-cached algorithm.

Bloom Filters - A Bloom filter is a probabilistic data structure for tracking set membership [21]. Given a set U and a set $S \subseteq U$ of elements produced through an operation on U, a Bloom filter can be used to record the membership of an element s in S. The Bloom filter is implemented using a vector of $\boldsymbol{m}$ bits, initialized to 0, and $\boldsymbol{k}$ independent hash functions $(h_1, h_2, \ldots, h_k)$, each with a range of 1-$\boldsymbol{m}$.

To denote (record) that an item $s \in S$, each hash function is applied to the item, produces an index, and sets the vector bit addressed by the index to 1. Thus, recording the fact that an item exists in the set requires $\boldsymbol{k}$ array bits to be modified. Determining if an item $s \in S$ is similar; each hash function is applied to the element s, produces an index, and checks that the corresponding bit is set to 1. If all the bits addressed by all the indexes are set to 1, the item is considered to exist in the set. Nevertheless, two distinct items can set the same bits, allowing the possibility of false positives. Hence, the addition of a Bloom filter to IHC might result in skipping states explored by the traditional IHC. This is not necessarily a problem since it adds a minute amount of randomness which can result in escaping from a basin.

As new elements are recorded, it becomes more probable that each bit in the vector has already been set by a previous recording. These occurrences are harmless up to a point, since a single 0 among the $\boldsymbol{k}$ index bits is sufficient to conclude with certainty that an element is not a member of the set. But an unbounded number of set recordings will inevitably saturate the vector and lead to a *false positive*, where an element is incorrectly identified as a member of the set due to the fact that all $\boldsymbol{k}$ index bits for that element have been set by previous recordings. Indeed, this is a salient characteristic of Bloom filters, which must be managed by careful choices of $\boldsymbol{m}$ and $\boldsymbol{k}$. A Bloom filter, however, would not make a *False Negative* decision.

The probability of false positive increases with n, and decreases with m. Hence, the m/n ratio is the most important tuning Bloom filter parameter [21]. This ratio, combined with an estimate of the worst-case n, yields the vector size m required for a Bloom filter of the desired accuracy. Once the m/n ratio is established, k is typically optimized as $\left\lceil \lg\left(\frac{m}{n}\right) \right\rceil$.

4 Experiments and Results

This paper reports on three types of experiments: 1) estimation of the theoretical limits of record keeping, 2) redundancy minimization using cache, and 3) redundancy minimization using Bloom filters. The experiments explore the utility of fixed number of restarts and the utility of fixed number of steps. Speed (and speedup) is measured based on the count of characteristic instructions and the number of 2-opt operations.

Estimation of the Theoretical Limits of Record Keeping - Dedicated memory of different sizes using a "freezing policy" as well as a simulation of Unbounded

memory record keeping using 50 TSP random graphs with 100 cities is used to determine an experimental upper bound on the IHC speedup using record keeping.

Minimizing Redundancy Using Cache - Following numerous experiments with different parameters, a least recently used (LRU) replacement policy with set associative cache is implemented [20]. Three cache configurations used in hardware caches: 4-way, 8-way, and 16-way set associative cache have been simulated by software. These configurations are evaluated and compared for speedup. Furthermore, each configuration is tested using between 1,000 and 64,000 cache blocks (a total cache size from 15KB to 896KB).

Minimizing Redundancy Using Bloom Filters - Bloom filters differ from cache techniques for record keeping in that false positives can occur with Bloom filters, whereas false positives do not occur with caching techniques. Therefore, the evaluation of Bloom filters must address two key points: 1) the amount of memory that is required to reduce false positive rates to acceptable levels in order to compete with cache-based record keeping in performance, and 2) determining whether the same performance improvements that are achieved by caching techniques are achievable with Bloom filters.

When using record keeping with CMSSAs and IHC, each generated tour is encoded and tested for equality against previously encoded generated tours; if an equal tour is found, the algorithm restarts with a new tour. Because Bloom filters allow false positives, it is possible that a tour that has not been previously considered will be recognized by the Bloom filter as previously generated. Therefore, a portion of the solution search space is excluded from exploration, possibly hiding high quality solutions. Furthermore, a low false positive rate is necessary to compare the speedup achieved by Bloom filters as opposed to cache configurations since skipping unexplored areas of the search space will artificially inflate speedup numbers.

Statistical Analysis – Each experiment is repeated numerous times (up to 400 times in some cases). This enables reliable statistical analysis. The mean value for the entire set of results reported lies inside the 95% confidence interval. Furthermore, when applicable, the significance level of reported conclusions has been tested and verified using the T-Test statistics.

4.1 Experiment 1: Fixed Number of Restarts

Unbounded Memory - For runs of 100,000 hill climbing iterations, the average speedup using unbounded memory record keeping is 10.9.

Dedicated Memory - Table I shows the results of compressing the trace of states explored using exponentially increasing dedicated memory of different sizes. The 0 entry stands for no record keeping and the ∞ entry represents unbounded memory. The 128 KB option is equivalent to the unbounded (∞) option. The table shows that the compression (and complexity) decreases as the dedicated memory size increases.

Table 1. Compression rates with several dedicated memory sizes (in K-bytes)

0	1K	2K	4K	8K	16K	32K	64K	128K	∞
8.3	6.3	5.9	5.9	5.6	4.8	4.3	2.8	2.5	2.5

Cache Memory - Figure 1 shows the speedup obtained using different cache configurations and sizes as well as dedicated memory. As the cache size grows, more tours are held in cache, and consequently, the execution speedup approaches the maximum achievable speedup. It is apparent that cache performs better than dedicated memory at lower memory levels. In addition, the figure shows that even a small cache provides significant speedup. For these experiments, roughly 1.25MB is enough memory to hold all duplicate tours. Even with a cache size of 1KB, we still achieve a speedup of 6.7. This result is especially useful in cases where TSP configurations generate many more duplicates tours than can fit in memory. It should be noted that the due to the assumption that the caching is implemented by hardware the speedup analysis does not take into account the overhead associated with caching.

Bloom Filter - Table 2 shows the memory size of the Bloom filter used for each configuration, as well as the number of errors resulting from false positives, the speed up achieved over non-record keeping, and the quality loss of the solution resulting from errors. As expected, larger memory usage results in fewer errors; however, a memory usage of 77KB proves to be sufficient to achieve optimal speedup with no quality loss for this experiment, which is considerably less memory than is required by cache configurations. Thus, Bloom filters are promising as a much more scalable means of record keeping as opposed to caching. A Bloom filter ceases to be useful when it becomes saturated. However, this situation can be easily detected, and versions of Bloom filters that have the ability to grow if the need arises exist [18].

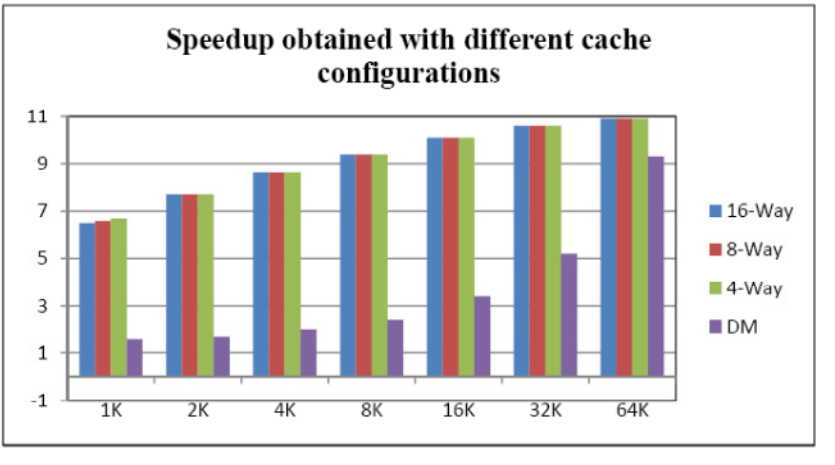


Fig. 1. Speedup obtained with different cache configurations

In summary both the cache and the Bloom filter achieve the theoretical limit applicable for this set of experiments (10.9). Nevertheless, the Bloom filter requires less memory and is expected to be less "expensive" in terms of hardware / software implementation. Note that due to false positives, the Bloom filter-based record keeping mistakenly identifies some configurations as redundant and skips their evaluation; resulting in artificially high speedup.

Table 2. Speedup and quality loss with Bloom filters

Bloom Filter				Quality Loss
m/n	K	Memory	Speedup	
6	4	77KB	11.0	0
5	3	64KB	11.0	.1 %
4	2	51KB	11.2	.1%
3	2	38KB	11.3	.1%
2	1	26KB	12.0	.2%
1	1	13KB	12.9	.3%

4.2 Experiment 2: Fixed Number of Steps

Time sensitive applications might employ an IHC algorithm with a fixed time limit, meaning that the best solution generated during a fixed amount of time is considered to be the best ("optimal") solution. Assuming that the time to generate a climbing step is constant, then a fixed time limit implies that only a constant number of steps are considered. Therefore, any redundancies that are eliminated through record keeping allow for exploration of a larger portion of the search space. This experiment measures the solution quality of record keeping mechanisms when limiting the number of steps generated to 50,000. Again, an unbounded memory record keeping mechanism is used to determine the optimal tour quality for the input graphs in the given number of steps. Figure 2 shows the results for unbounded memory record keeping, and two configurations of a Bloom filter using: 1) 1.25KB of memory and 2) 10KB of memory. The 10KB Bloom filter effectively achieves the same performance as unbounded memory with 6.8% of the memory usage. Similar experiments with cache show that a cache of roughly 118KB is sufficient to achieve a nearly identical tour quality graph to unbounded memory. The 10KB version of the Bloom filter displays interesting behavior at around the 10,300th step in which the tour

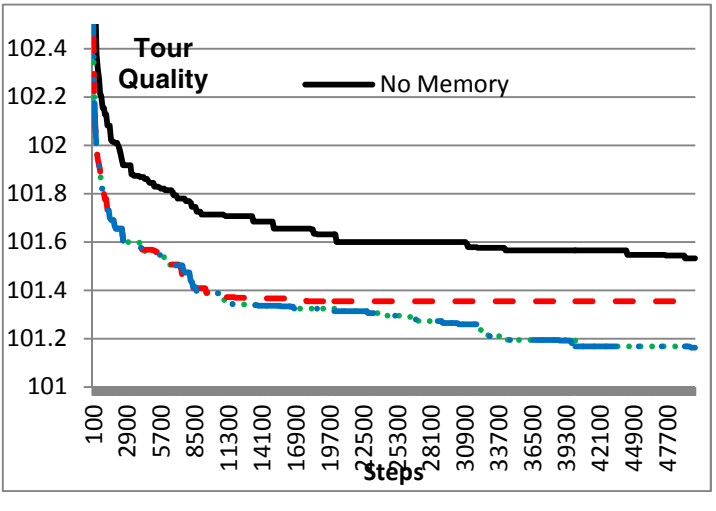


Fig. 2. Quality improvement with Bloom filter

quality remains relatively constant. At that point, the Bloom filter becomes saturated and returns a false positive for nearly every tour generated. This situation exhibits the drawback of Bloom filters for record keeping in which a Bloom filter ceases to be useful when it becomes saturated. However, as stated, this situation can be detected, and versions of Bloom filters that possess the ability to grow dynamically exist [18].

5 Conclusions

Record keeping mechanisms considerably improve the quality of tours generated by CMSSAs approach for solving the traveling salesman problem by allowing a larger portion of the search space to be explored in time period. Furthermore, we have shown that caching mechanisms can be highly effective even for small cache sizes. A significant finding is that Bloom filters are able to achieve optimal speedup for a constant number of restarts using only a fraction of the memory of caching mechanisms. Given that Bloom filters are more economic than comparable caches this finding has an important implication on the reducing the cost of record keeping. Therefore, properly configured Bloom filters show promise in limited memory applications, and scale well to take advantage of larger memories for increased performance. Another Unique contribution of the research is the empirical analysis of complexity and redundancy.

Future work plans includes further analysis of the theoretical aspects of record keeping, applying the findings to the MPPT and other COP problems, and using additional heuristic / record keeping methods.

References

[1] Pearl, J.: Heuristics; Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley Publishing Company (1984)
[2] Kolaczkowski, B., Thornton, J.W.: Performance of maximum parsimony and likelihood phylogenetics when evolution is heterogeneous. Nature, 980–984 (2004)
[3] Feo, T.A., Resende, M.G.C.: Greedy randomized adaptive search procedures. J. of Global Optimization 6, 109–133 (1995)
[4] Karhi, D., Tamir, D.E.: Caching in the TSP Search Space. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 221–230. Springer, Heidelberg (2009)
[5] King, C.R.: Improving the Performance of Constructive Multi-start Search using Recordkeeping. MS thesis, Texas State University: San Marcos (2010)
[6] Rego, C., Glover, F.: Local Search and Metaheuristics. In: Gutin, G., Punnen, A. (eds.) The Traveling Salesman Problem and its Variations, pp. 309–368. Kluwer Academic Publishers, Dordrecht (2002)
[7] Garey, M.R., Johnson, D.S.: Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman (1979)
[8] Johnson, D., McGeoch, L.: The Traveling Salesman Problem: A Case Study in Local Optimization. In: Aarts, E., Lenstra, J. (eds.) Local Search in Combinatorial Optimization, pp. 215–310. John Wiley and Sons, London (1977)

[9] Applegate, D.L.: The traveling Salesman Problem, A Computational Study. Princeton University Press, Princeton (2006)
[10] Tamir, D.E., Novoa, C., Lowell, D.: Time space Tradeoffs in GA Based Feature Selection for Workload Characterization. In: Proceedings of the 23rd International Conf. on Industrial, Engineering & Other Applications of Applied Intelligent Systems, Cordoba, Spain (2010)
[11] Cormen, T., Leiserson, C., Rivest, R., Stein, C.: Introduction to Algorithms. The MIT Press (2001)
[12] Kolmogorov, A.: Three Approaches to the Quantitative Definition of Information. IEEE Transactions on Information Theory, 1–7 (1965)
[13] Ziv, J., Lempel, A.: Compression of Individual Sequences via Variable-Rate Coding. IEEE Transactions on Information Theory, 530–536 (1978)
[14] Glover, F., Laguna, M.: Tabu Search. Kluwer Academic Publishers (1997)
[15] Hertel, P., Pitassi, T.: An Exponential Time/Space Speedup For Resolution. In: Electronic Colloquium on Computational Complexity (ECCC) (2007)
[16] Allen, D., Darwiche, A.: Optimal Time-Space Tradeoff in Probabilistic Inference. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence, pp. 969–975 (2003)
[17] Aggarwal, A.: Software Caching Vs. Prefetching. In: ISMM 2002: Proceedings of the 3rd International Symposium on Memory Management, pp. 157–162. ACM, Berlin (2002)
[18] Almeida, P., Baquero, C., Preguica, N., Hutchison, D.: Scalable Bloom Filters. Information Processing Letters, 255–261 (2007)
[19] Santos, E.E., Santos Jr., E.: Cache Diversity in Genetic Algorithm Design. In: FLAIRS Conference, pp. 107–111 (2000)
[20] Hennessy, J.L., Patterson, D.A.: Computer Architecture, A Quantative Approach, 4th edn. Morgan Kaufmann Publishers, Inc., San Francisco (2007)
[21] Dillinger, P.C., Manolios, P.: Bloom Filters in Probabilistic Verification. In: Hu, A.J., Martin, A.K. (eds.) FMCAD 2004. LNCS, vol. 3312, pp. 367–381. Springer, Heidelberg (2004)

A Modified Particle Swarm Optimizer for Engineering Design

Li Ma and Babak Forouraghi

Computer Science Department, Saint Joseph's University,
Philadelphia, PA, U.S.A.
{Li.Ma,Babak.Forouraghi}@sju.edu

Abstract. Particle swarm optimization (PSO) has been widely used in multi-objective engineering design optimization where parameter selection is of prime importance. This paper proposes a multi-objective particle swarm optimizer (MOPSO) with a modified crowding factor and enhanced local search ability. A new parameter-less sharing method is introduced to estimate the density of particles' neighborhood in the search space. Initially, the proposed method determines the crowding factor of the solutions; in later stages, it effectively guides the entire swarm to converge closely to the true Pareto front. In addition, the gradient descent search method is applied. The algorithm's performance on two engineering design problems is highlighted and compared with other approaches. The obtained results demonstrate that the proposed algorithm is capable of effectively searching along the Pareto optimal front and successfully obtaining trade-off solutions for the engineering design problems.

Keywords: Multi-objective Optimization, Particle Swarm Optimization, Pareto Front, Exploration and Exploitation.

1 Introduction

During the last decade, particle swarm optimization (PSO) has attracted a great deal of attention in the field of multi-objective optimization due to its simplicity, efficiency, and quick convergence. Multi-objective problems, which typically involve conflicting objectives, play a leading role in various engineering [2-4,13,14,20,21,23]. According to a distinguishing characteristic of multi-objective problems, there is no single solution considered as the optimal solution. Instead, a set of solution alternatives, known as Pareto-optimal solutions, is provided to the decision maker. So far, numerous variations of particle swarm optimization for multi-objective problems (MOPSO) have been proposed and proven to be effective. These methods achieve the goal of finding as many of the Pareto-optimal solutions as possible in a search space which is well-distributed along the Pareto front.

A MOPSO approach has been proposed which relies on the concept of Pareto dominance to determine the flight direction of a particle and maintains the discovered Pareto-optimal solutions in an external repository; in addition, a technique called hypercube formation is used to calculate fitness sharing [3]. A MOPSO was proposed that puts forward an efficient mutation strategy called elitist-mutation to effectively

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 187–196, 2012.

explore the feasible search space and speed up the search towards the Pareto front in conjunction with the crowding distance metric and a variable-sized external repository [15,21]. To deal with multi-objective problems in the field of manufacturing robust design, a PSO algorithm was proposed to achieve off-line quality control [18]. To minimize the mean value of the objectives and the standard deviation, a combination of MOPSO and the quasi-Newton method was introduced to find robust solutions against small perturbations of design variables [19].

In the proposed MOPSO presented in this paper the density of the search space is defined by a modified crowding factor which identifies social leaders by randomly selecting them from sparsely-populated areas. This crowding factor is an improvement over previous methods in that not only it provides the estimation of the density in the neighborhood but it also provides a fitness sharing mechanism which degrades fitness values of non-promising solutions. Further, to enhance the local search ability of the algorithm a gradient descent method is applied to a small proportion of the global Pareto-optimal solutions when the size of the repository exceeds a user-defined threshold.

The remainder of this paper is organized as follows. Section 2 provides the background information on particle swarm optimization. Section 3 introduces the proposed MOPSO algorithm. Sections 4 and 5 include the simulation results for two engineering design problems. And finally, Section 6 is the summary and conclusions.

2 Particle Swarm Optimization

Particle swarm optimization (PSO) is a stochastic global optimization approach, and its main strength is in its simplicity and fast convergence rates. The following is a brief introduction to PSO [1,7,22]:

A total of p particles are randomly distributed throughout the feasible design region, where X_i^t is the position of a particle i representing a design scenario at time t. The position of the particle can be updated using $X_i^{t+1} = X_i^t + V_i^{t+1}$, where:

$$V_i^{t+1} = \omega V_i^t + c_1 r_1 (P_i^t - X_i^t) + c_2 r_2 (P_g^t - X_i^t) \quad (1)$$

The point P_i^t is the best local solution found up to time t and represents the cognitive contribution to the search vector V_i^{t+1}. The point P_g^t is the best current global solution and forms the social contribution to the velocity vector. Random numbers r_1 and r_2 are uniformly distributed in the interval [0, 1]. The cognitive and social scaling factors c_1 and c_2 are typically selected such that $c_1 \cdot c_1$ and $c_2 \cdot c_2$ have a mean of 1 so that the particles overshoot the attraction points P_i^t and P_g^t half the time, thereby allowing wider search fronts [15]. The variable ω is the inertia weight and is typically chosen in the range of [0, 1]. A larger inertia weight facilitates global exploration and a smaller inertia tends to facilitate local exploration. Therefore, ω is a critical factor for the convergence behavior of PSO and is used to promote global exploration of the search space [22].

The cognitive learning factor is computed by the term $c_1 r_1 (P_i^t - X_i^t)$ in Eq. 1, and it is the short term memory of a particle representing the particle's inclination to repeat past behavior that has proven to be successful for that particular particle.

The social learning factor, on the other hand, is computed by the term $c_2 r_2 (P_g^t - X_i^t)$ in Eq. 1, and it is the peer pressure of a particle representing the particle's inclination to

imitate or emulate the behavior of other particles that are successful; it is the influence of a particle's neighbors.

3 The Proposed MOPSO Methodology

In order to solve multi-objective problems, a combination of particle swarm optimization and the Pareto-dominance strategy 2,4,13,14,20,21,23] can be used to find a set of global optimal solutions or Pareto-optimal solutions. Moreover, an external repository is requisite to store the Pareto-optimal solutions discovered so far. When a particle violates the constraints, the fly-back mechanism is applied to force the particle to revisit its previous position [12]. Further, to promote diversity several randomly-generated (mutated) particles are added to the repository.

The maintenance of the global repository is a crucial issue. The size of the repository is defined as a free system parameter. Particles in the densely-populated areas have the priority to be removed when the repository is oversized. The density of the search space is defined by a novel crowding factor. According to this crowding factor, social leaders are easily determined by randomly selecting one among those in the sparsely-populated areas. Finally, to enhance the local search ability, a gradient-based search method is applied to a small proportion of the global Pareto-optimal solutions when the size of the repository exceeds a user-defined threshold. The contributions of the proposed MOPSO in this paper are a modified social leader selection mechanism called the *Modified Crowding Factor* (MCF), and an enhanced local search using the steepest gradient-decent search.

3.1 A Modified Crowding Factor

Several social leader selection strategies based on the density measure of the population have been previously proposed. The two most widely-used measures are the nearest-neighbor density estimator [6] and the kernel density estimator [8].

The nearest neighbor density estimator quantifies how crowded the closest neighbors of a given particle are in the objective space [6]. This measure is estimated by the area of the largest cuboid formed by using the two nearest neighbors of particle *i* as the vertices [22].

The kernel density estimator provides a fitness-sharing mechanism achieved by degrading fitness values which are obtained by dividing the scaled fitness value of an individual by a quantity proportional to the number of individuals in the neighborhood [8].

A modified social leader selection method (*MDF*) was devised and implemented in this work. This proposed parameter-less sharing is an improvement over the two above-mentioned methods in that not only it provides the estimation of the density in the neighborhood, but it also provides a fitness sharing mechanism that degrades the fitness value of an individual solution with respect to a set of solutions in a similar circumstance. Before calculating the MDF value, a new sharing area in the objective space is calculated for the current generation. Each swarm particle is viewed as a hyper-circle whose center is a particle's objective vector and whose radius is the vector:

$$V = \frac{\max(f(i,j)) - \min(f(i,j))}{N} \tag{2}$$

where $f(i, j)$ is the jth dimension of the ith objective function, and N is the size of the global repository.

The MDF value for a particle is defined as the number of solutions in that particle's sharing area. Thus, the minimum crowding value of a particle is 1 because it only appears in its own sharing area. The Pareto-optimal solutions in the global repository can be divided into groups where members of each group share the same crowding value.

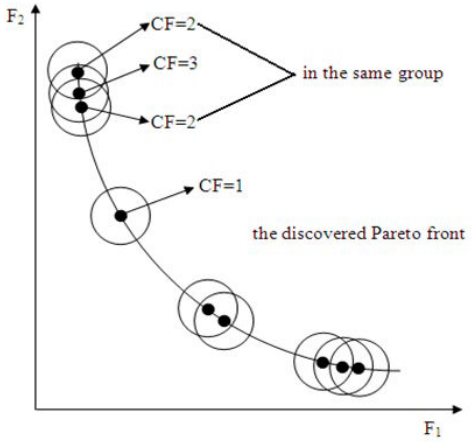


Fig. 1. A Modified Crowding Factor (MDF)

During the initial stage of social leader selection, 10% of the repository corresponding to the less-populated groups is subjectively identified and a leader among that group is randomly selected.

3.2 Enhanced Local Search

The steepest gradient-descent method enhances local search, and the PSO's linearly-decreasing inertia weight is capable of partial local search. This kind of search is non-deterministic and difficult to measure the performance. In addition, it reduces the optimizer's ability to efficiently explore the solution space. To force MOPSO to focus on the global search front, the steepest gradient-descent method [9,10] is used to exploit the local area of each Pareto-optimal solution. Searching along the gradient-descent direction is a guarantee of obtaining local Pareto-optimal solutions. Therefore, the gradient-descent search has a high probability of obtaining the global Pareto-optimal solutions among the local solutions.

This work's enhanced local search can be described as follows: first, 5% of the global Pareto-optimal solutions in the less-populated areas are selected; and second, the steepest gradient-descent algorithm is applied to the selected solutions using:

$$X' = X - (s.\nabla F(X)) \tag{3}$$

where X is the design vector, s is the step size of gradient search for each particle, and $\nabla F(X)$ is the first order derivative of the objective function. Fig. 2 presents the proposed algorithm in more detail.

Step 1	Initialize the particles and the global repository
Step 2	Update the inertia weight for the current generation Evaluate the particles in the current generation
Step 3	For each particle in the current generation: - update particle's own memory - choose the current social leader - update the velocity and position - check the feasibility
Step 4	Update the global repository
Step 5	If the size of the repository is greater than the threshold: - select 5% of the global Pareto optimal solutions in the less-populated area - apply the steepest gradient descent search to the selected solutions - add the non-dominated solutions from the obtained local Pareto optimal solutions to the global repository - update the global repository to maintain the non-dominance property
Step 6	Go to step 2 until the stopping criterion is met

Fig. 2. The proposed MOPSO algorithm

4 Design of an I-beam

This section presents the problem of multiobjective design of an I-beam which has previously been approached using classical vector optimization techniques [11,17,18].

Assuming that the I-beam is subject to maximal bending forces of P=600 kN and Q=50 kN at the midspan, the objective of the design is to find the optimum dimensions of the beam ($X^*=[x_1, x_2, x_3, x_4]^T$) such that the cross section area (f_1 in cm^2) and static deflection of the beam (f_2 in cm) are both minimized subject to the constraint that the beam's bending stress (f_3) does not exceed 16 kN/cm^2.

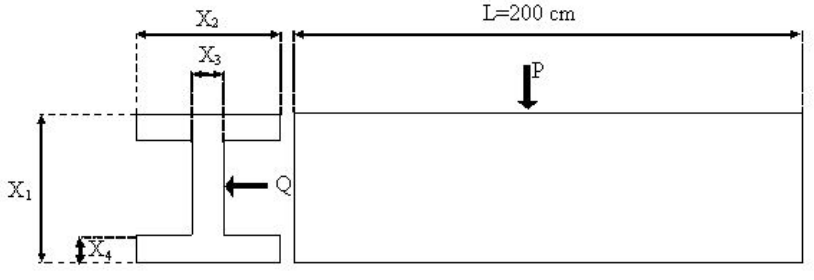


Fig. 3. The frontal and side views of an I-beam

The optimization of the beam can mathematically be stated as follows. Find X^* which minimizes $F(X) = [f_1(X), f_2(X)]^T$ where:
where,

$$f_1(X) = 2x_2x_4 + x_3(x_1 - 2x_4) \tag{4}$$

$$f_2(X) = \frac{60{,}000}{x_3(x_1 - 2x_4)^3 + 2x_2x_4\left[4x_4^2 + 3x_1(x_1 - 2x_4)\right]} \tag{5}$$

subject to the bending stress constraint:

$$f_3(X) = \frac{180{,}000x_1}{x_3(x_1 - 2x_4)^3 + 2x_2x_4\left[4x_4^2 + 3x_1(x_1 - 2x_4)\right]} + \frac{15{,}000x_2}{(x_1 - 2x_4)x_3^3 + 2x_4x_2^3} \leq 16 \tag{6}$$

and the geometric side constraints: $10 \leq x_1 \leq 80$, $10 \leq x_2 \leq 50$, and $0.9 \leq x_3$, $x_4 \leq 5.0$. Given the boundaries of the feasible design region, the computed ranges of responses for the two objective functions reveal that f_1 is in conflict with f_2 and that the ideal solution $f^{id} = [25.38, 0.0059]^T$, where the two objectives are simultaneously minimized, can never be attained.

To account for statistical fluctuations that can potentially produce misleading results, the proposed MOPSO algorithm was tested over 3 statistically independent runs, each time with a population of 50 particles over 60 generations. The inertia weight was fixed at 0.9 during the evolution. The cognitive and social learning rates were both set to 0.5.

The rapid convergence behavior is illustrated in Fig. 4.

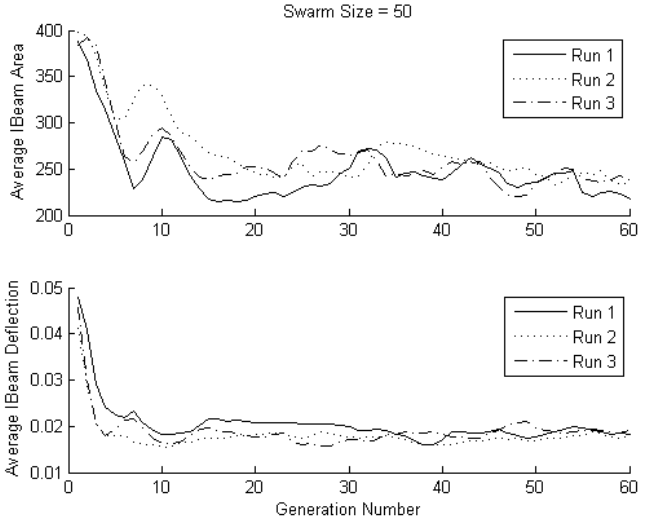


Fig. 4. Mean objective values for the I-Beam

Clearly, this demonstrates that the proposed MOPSO algorithm is capable of guiding the whole swarm to approach the optimal design region very early, due to its enhanced evolutionary process, which uses particles in less-populated area as reference to let other particles imitate and fly toward.

Table 1 presents five sets of trade-off solutions for the I-beam problem reported in several studies [11,16-18] as well as the ones discovered by the proposed MOPSO algorithm. The objective vector in the table contains the cross-sectional area, the volume of the beam and the bending stress. Compared to other approaches, the proposed algorithm successfully obtained competitive trade-off solutions.

Table 1. Comparison of the trade-off solutions

Method	$X = (x_1, x_2, x_3, x_4)$	$f(X) = (f_1(X), f_2(X), f_3(X))$
Min-max (Osyzcka)	1- (79.63, 48.41, 0.89, 3.66) 2- (79.87, 48.82, 0.89, 2.66) 3- (79.94, 48.88, 0.89, 1.99)	(419.9, 0.0093, 3.087) (326.5, 0.0119, 4.049) (263.3, 0.0151, 5.231)
Min-max (Hajela & Shih)	1- (79.99, 49.99, 0.90, 2.39) 2- (80.00, 50.00, 0.90, 2.08) 3- (79.99, 50.00, 0.90, 1.79)	(307.5, 0.0127, 4.318) (276.5, 0.0143, 4.879) (247.8, 0.0163, 5.579)
Taguchi (Kunjur & Krishnamurty)	1- (80.00, 50.00, 0.90, 2.30) 2- (80.00, 50.00, 0.90, 3.20) 3- (80.00, 50.00, 0.90, 4.10)	(297.8, 0.0132, 4.467) (386.2, 0.0099, 3.327) (474.6, 0.0081, 2.671)
Hyperparticle Swarm Optimization (Ochlak & Forouraghi)	1- ([79.17, 80.00], [45.30, 50.00], [0.900, 0.903], [0.9, 1.64]) 2- ([78.98, 80.00], [46.38, 50.00], [0.900, 0.904], [1.03, 1.71]) 3- ([78.42, 80.00], [47.63, 50.00], [0.900, 0.902], [0.94, 1.88])	(176.31±0.265, 0.025±4.02E-5, 8.94±0.17) (202.65±0.46, 0.021±6.79E-5, 7.39±0.0244) (213.45±0.14, 0.020±1.99E-5, 6.82±0.007)
The proposed MOPSO algorithm	1- (79.938, 49.137, 0.900, 1.075) 2- (80.000, 49.114, 0.900, 1.353) 3- (79.848, 50.000, 0.906, 1.434)	(175.6667, 0.0251, 8.8961) (202.4238, 0.0208, 7.2962) (213.1363, 0.0196, 6.7911)

5 Design of a Two-Bar Truss

A two-bar truss [5] is loaded by a vertical load of 100 *kN*. The lengths of bars AC and BC depend on the location of the joint C. The overall design goal is to minimize the total weight (volume) of the truss such that yielding and buckling are prevented as much as possible. The three design parameters are x_1 (cross-sectional are of AC), x_2 (cross-sectional area of BC) and y (vertical distance between the joints B and C). Thus, the design vector can be denoted as $X = (x_1, x_2, y)$.

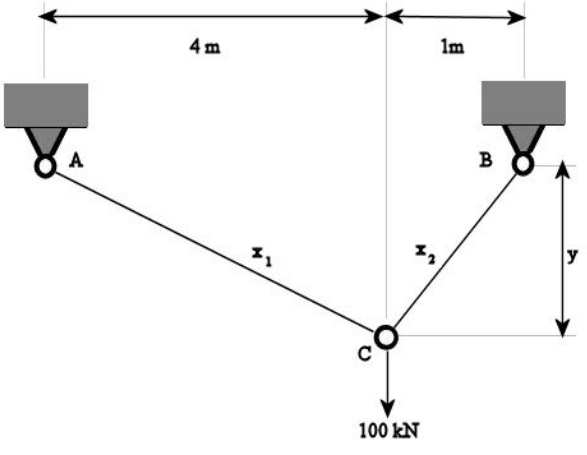


Fig. 5. A two-bar truss

The multi-objective optimization of the truss can mathematically be stated as follows. Find X^* which minimizes $F(X) = [f_1(X), f_2(X)]^T$ where,

$$f_1(X) = x_1\sqrt{16+y^2} + x_2\sqrt{1+y^2} \tag{7}$$

$$f_2(X) = \max(\sigma_{AC}, \sigma_{BC}) \tag{8}$$

subject to $0 \le x_1, x_2 \le 0.01$, $1 \le y \le 3$, and the following constraint:

$$\max(\sigma_{AC}, \sigma_{BC}) \le 1 \times 10^5 \tag{9}$$

The stresses for the two bars can be calculated as:

$$\sigma_{AC} = \frac{20\sqrt{16+y^2}}{yx_1} \tag{10}$$

$$\sigma_{BC} = \frac{80\sqrt{1+y^2}}{yx_2} \tag{11}$$

The proposed MOPSO algorithm was tested over 3 independent runs to avoid the statistical fluctuation. The swarm evolved a population of 100 particles over 100 generations.

Fig. 6 depicts how the design objectives converged during the process. Due to the utilized feasibility strategy, the swarm was initialized in the lower stress region. The entire swarm was then guided to the higher stress region (a less-populated area).

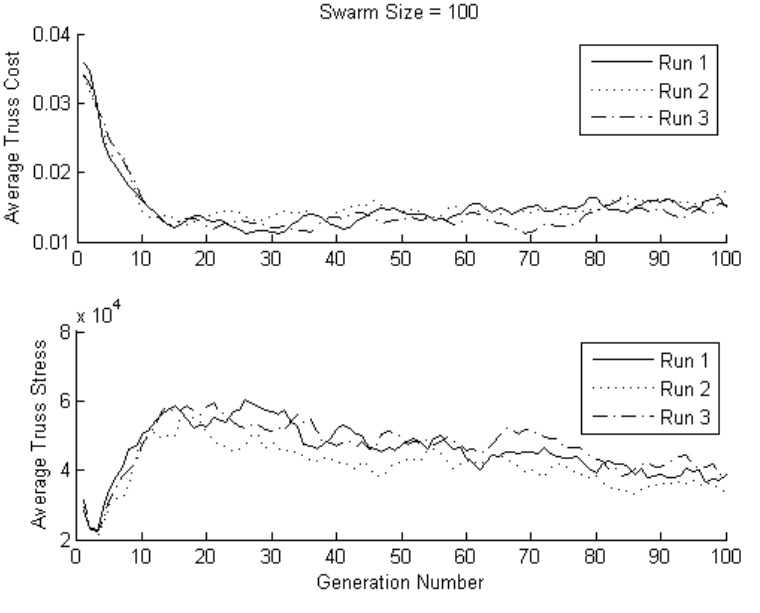


Fig. 6. Mean objective values for the two-bar truss

The trade-off design solutions are shown in Table 2, compared to the optimal solutions presented by the ∈-inequality constraint method. Considering the minimization of the total weight, the minimization of the maximum stress between AC and BC, and the balance of the stress between AC and BC, the trade-off solutions produced by the proposed MOPSO algorithm are superior to the optimal solutions found by the ∈-inequality constraint method.

Table 2. Comparison of the trade-off solutions

Method	$X = (x_1, x_2, y)$	$f(X) = (f_1(X), f_2(X), f_3(X))$
∈-inequality constraint method	1- (0.000497, 0.00099, 1.999)	(0.004445, 89983, 7)
	2- (0.000496, 0.00094, 2.034)	(0.004541, 88296, 428)
	3- (0.000519, 0.00110, 1.954)	(0.004740, 87697, 6449)
	4- (0.000517, 0.00103, 2.003)	(0.004621, 86786, 462)
	5- (0.000542, 0.00107, 2.007)	(0.004833, 83268, 1025)
The proposed MOPSO algorithm	1- (4.97011E-4, 9.94010E-4, 2.00008)	(0.004445, 89981, 3)
	2- (4.97526E-4, 1.00953E-3, 2.05211)	(0.004541, 88153, 87)
	3- (5.08185E-4, 1.02240E-3, 2.01471)	(0.004576, 87488, 133)
	4- (5.10357E-4, 1.02857E-3, 2.02667)	(0.004613, 86730, 24)
	5- (5.41262E-4, 1.08186E-3, 1.99288)	(0.004831, 82860, 126)

6 Conclusions

In this paper a new PSO-based approach was presented which combines stochastic optimization with Pareto optimal design. In the proposed algorithm the density of the search space is defined by a modified crowding factor which determines social leaders by randomly selecting them from sparsely-populated areas. It was demonstrated that performing local search using the steepest gradient-descent method helps MOPSO to focus on global search in order to efficiently solve multi-objective engineering design problems. In addition, the enhanced crowding factor was shown to guide an entire swarm toward the optimal design region. The performance on two well-studied engineering design problems highlighted the advantages of the proposed MOPSO algorithm using a modified crowding factor and the gradient-descent search.

References

1. Clerc, M.: Particle Swarm Optimization. ISTE Ltd., California (2006)
2. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A. (eds.): Evolutionary Algorithms for Solving Multi-Objective Problems. Springer, New York (2007)
3. Coello Coello, C.A., Lechuga, M.S.: MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. In: Proceedings of the Congress on Evolutionary Computation, vol. 2, pp. 1051–1056 (2002)
4. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, Chichester (2001)
5. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, New York (2004)

6. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. KanGAL Report 200001, Institute of Technology, Kanpur, India (2000)
7. Engelbrecht, A.P.: Fundamentals of Computational Swarm Intelligence. Wiley, Chichester (2005)
8. Goldberg, D.E., Richardson, J.: Genetic Algorithms with Sharing for Multimodal Function Optimization. In: Proceedings of the Second International Conference on Genetic Algorithms, pp. 41–49 (1987)
9. Gosavi, A.: Simulation-based Optimization: Parametric Optimization Techniques and Reinforcement Learning. Springer, New York (2003)
10. Haftka, R.T., Gürdal, Z.: Elements of Structural Optimization. Springer, New York (1992)
11. Hajela, P., Shih, C.J.: Multiobjective Optimum Design in Mixed Integer and Discrete Design Variable Problems. AIAA Journal 28(4), 670–675 (1990)
12. He, S., Prempain, E., Wu, Q.H.: An Improved Particle Swarm Optimizer for Mechanical Design Optimization Problems. Engineering Optimization 36, 585–605 (2004)
13. Ho, S.L., Yang, S., Ni, G., Lo, E.W., Wong, H.C.: A Particle Swarm Optimization-Based Method for Multiobjective Design Optimizations. IEEE Transactions on Magnetics 41, 1756–1759 (2005)
14. Liu, D., Tan, K., Goh, C., Ho, W.: A Multiobjective Memetic Algorithm based on Particle Swarm Optimization. IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics 37, 585–605 (2007)
15. Li, X.: Better Spread and Convergence: Particle Swarm Multiobjective Optimization Using the Maximin Fitness Function. In: Deb, K., et al. (eds.) GECCO 2004, Part I. LNCS, vol. 3102, pp. 117–128. Springer, Heidelberg (2004)
16. Kunjur, A., Krishnamurty, S.: A Robust Multi-Criteria Optimization Approach. Mechanism and Machine Theory 32(7), 797–805 (1997)
17. Osyczka, A.: Multicriteria Optimization for Engineering Design in Design Optimization. Academic Press (1985)
18. Ochlak, E., Forouraghi, B.: A Particle Swarm Algorithm for Multiobjective Design Optimization. In: Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2006), pp. 765–772 (2006)
19. Ono, S., Nakayama, S.: Multi-objective Particle Swarm Optimization for Robust Optimization And Its Hybridization With Gradient Search. In: IEEE Congress on Evolutionary Computation, pp. 1629–1636 (2009)
20. Ray, T., Liew, K.M.: A Swarm Metaphor for Multiobjective Design Optimization. Engineering Optimization 34, 141–153 (2002)
21. Reddy, M.J., Kumar, D.N.: An Efficient Multi-objective Optimization Algorithm based on Swarm Intelligence for Engineering Design. Engineering Optimization 39, 49–68 (2007)
22. Reyes-Sierra, M., Coello Coello, C.A.: A Survey of the State-of-the-Art Multi-Objective Particle Swarm Optimizers. International Journal of Computational Intelligence Research 2, 287–308 (2006)
23. Shim, M., Suh, M.: Pareto-based Continuous Evolutionary Algorithms for Multiobjective Optimization. Engineering Computation 19, 22–48 (2002)

Homogeneous RF Coil Design Using a GA

Karthik Nadig[1], William M. Potter[2], and Walter D. Potter[3]

[1] Artificial Intelligence, The University of Georgia
kvnadig@uga.edu
[2] Physics & Astronomy, The University of Georgia
wmpotter@uga.edu
[3] Computer Science & Artificial Intelligence, The University of Georgia
potter@uga.edu

Abstract. Optimizing the magnetic field homogeneity of single/dual tuned birdcage coils used in Magnetic Resonance Spectroscopy improves measurement accuracy of metabolite concentrations. This work in progress article focuses on the novel domain development of a heuristic technique using nature inspired optimization to find the best possible parameters for a birdcage coil design.

1 Introduction

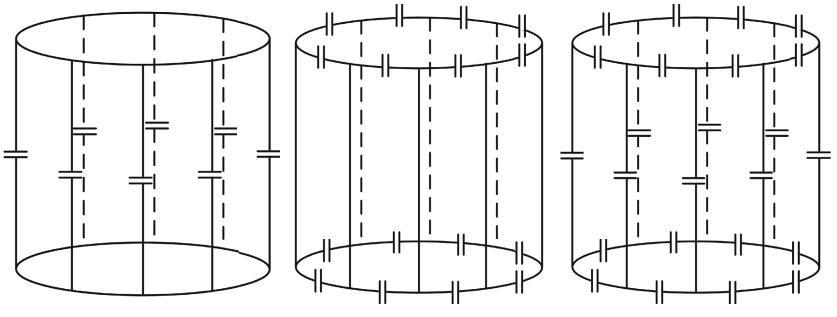
Magnetic Resonance Spectroscopy (MRS) is used as a diagnostic tool to analyze biological functions of the metabolites in the human brain. Optimizing MRS improves the chances of accurate analysis. The problem of optimizing MRS requires improving the design of Radio Frequency (RF) head coils[1], called birdcage coils, used to encircle a human patient's head. There are two parts to this problem; first Multi-Nuclear Spectroscopy (MNS) requires the use of a dual-tuned coil or multiple coils that will provide homogeneous B_1 fields at the resonant frequency for both the nucleus to be analyzed and Hydrogen. Second, the coil is required to be highly efficient under strong static magnetic field B_0 in order to maximize the signal-to-noise ratio (SNR). Birdcage coils produce a very homogeneous rotating magnetic field, inside the coil, that is able to both excite (or transmit) atomic spins and acquire (or receive) signal from the perturbed spins as they realign themselves with the static B_0 field. The homogeneity of the generated magnetic field is proportional to the signal-to-noise ratio (SNR) obtained from the coil. A better SNR will allow for more accurate analysis of metabolites in the human body[8]. The number of variables (or parameters) that a birdcage coil has makes it a difficult structure to model. However, one approach to satisfy the requirement for homogeneity of the magnetic field would be to optimize the physical dimensions of the coil. A novel heuristic based optimization technique is presented in this article that combines the features of the Genetic Algorithm and Finite Difference Time Domain (FDTD) simulations to optimize the birdcage coil configuration. Also discussed in this article are novel techniques to determine various geometric parameters for constructing a coil using the Control Language (CTL) and simulations using the MEEP FDTD library [7]. In

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 197–205, 2012.

addition, a custom in house simulator was built based on FDTD for the specific needs of the experiment. The goal of the research is to identify an automated optimization methodology that enhances the performance of customized patient RF coils over existing trial and error methods.

2 Birdcage Coils

The RF coil serves one of two purposes, to generate RF pulses at the Larmor frequency in order to excite the nuclei in the object to be scanned and to pickup RF signals emitted by the nuclei at the same frequency. When generating RF pulses the coil is called an RF transmit coil, and when receiving, the coil is called an RF receive coil. The magnetic field of the RF pulse generated by an RF transmit coil is the B_1 field, whose direction is perpendicular to the direction of the main magnetic field B_0. Note, the B_0 field is along the coil axis while B_1 field is perpendicular to the B_0 field. To obtain high quality MRI images the coil must generate a homogeneous B_1 field when transmitting and, when receiving, the coil must have a high Signal-to-Noise Ratio (SNR) and have the same signal gain in the region of interest. The birdcage coil is especially known for its ability to produce homogeneous B_1 fields over the large volume in the coil. The Fig. 1 shows various types of birdcage coil configurations[5].



(a) Lowpass Birdcage (b) Highpass Birdcage (c) Bandpass Birdcage

Fig. 1. Types of Birdcage coils

The coils have two basic parts; the rings and the rungs (or legs). The rings form the cylindrical shape of the birdcage coils, and the legs are usually equally spaced strips or wires. Capacitors may be placed on either the rings or the legs of the coil. When the capacitors are on the legs, the coil is said to be configured as a lowpass birdcage coil, as shown in Fig. 1a. If the capacitors are on the rings then the coil is a highpass birdcage coil, as shown in Fig. 1b. Finally if the capacitors are on both the rings and the legs, as in Fig. 1c, the coil is called a bandpass or hybrid birdcage coil.

The configuration of the coil determines how the current flows through the coil, which plays a major role in the mode in which the coil can operate. The resonant frequency of the lowpass coil can be calculated using (1). Where, m is the resonant mode, N is the number of legs, C is the capacitance, M is the self-inductance of the legs and L is the self-inductance of the connector for the capacitors. A lowpass coil can operate in the modes defined by $m = \left(0, 1, 2, \ldots, \frac{N}{2}\right)$. The lowpass birdcage coil is operated usually with $m = 1$, the second lowest resonant, since in resonant mode $m = 0$ the resonant frequency is zero. The $m = 1$ mode is called the primary mode and is the mode which gives the most uniform magnetic field in the center of the coil. The current distribution around the legs of the coil has the pattern of a sinusoidal wave.

$$\omega_m = \frac{1}{\sqrt{C\left(M + \frac{L}{2sin^2\left(\frac{\pi m}{N}\right)}\right)}} \tag{1}$$

Similarly, for a highpass birdcage coil, the resonant frequency is given by (2) and the resonant mode of a highpass coil is defined by $m = \left(0, 1, 2, \ldots, \frac{N}{2}\right)$. However, unlike the lowpass coil, at $m = 0$ the highpass coil has the highest resonant frequency and for $m = 1$ it has the second highest resonant frequency. Finally, for a bandpass birdcage coil the resonant frequency is given by (3). The resonant mode of the bandpass coil is also defined by $m = \left(0, 1, 2, \ldots, \frac{N}{2}\right)$. The end-ring resonant mode of the bandpass coil is given by $m = 0$. Unlike the lowpass or highpass birdcage coils, the $m = 1$ frequency are not the second lowest or the second highest resonant frequencies.

$$\omega_m = \frac{1}{\sqrt{C\left(L + 2Msin^2\left(\frac{\pi m}{N}\right)\right)}} \tag{2}$$

$$\omega_m = \sqrt{\frac{L + 2Msin^2\left(\frac{\pi m}{N}\right)}{\frac{1}{C} + \frac{2}{C'}sin^2\left(\frac{\pi m}{N}\right)}} \tag{3}$$

A more advanced coil structure can be developed using the basic coils shown in Fig. 1. An example of such a coil is the dual-tuned birdcage coil Fig. 2, which has three lowpass birdcage coils connected along the axis and the end-rings are fused together. The dual-tuned coils are commonly used types of coils in MRI/MRS examinations. A dual-tuned birdcage coil has the following geometric parameters that could be optimized such as the diameter (d), the length of the inner birdcage (l_i), the length of the outer birdcage (l_o), the number of legs or rungs of the birdcage (n), and the width of the copper strip (w). Initially, to simplify this problem, l_o is assumed to be 0, turning the dual-tuned coil into a single-tuned coil, Fig. 1a. This constraint was added to reduce the number of dimensions to be searched in the search space. Also, a single-tuned coil has fewer design parameters over other coil types such as concentric, 4-ring, and alternate rung designs. Another design choice made is that the coil is tuned for Hydrogen

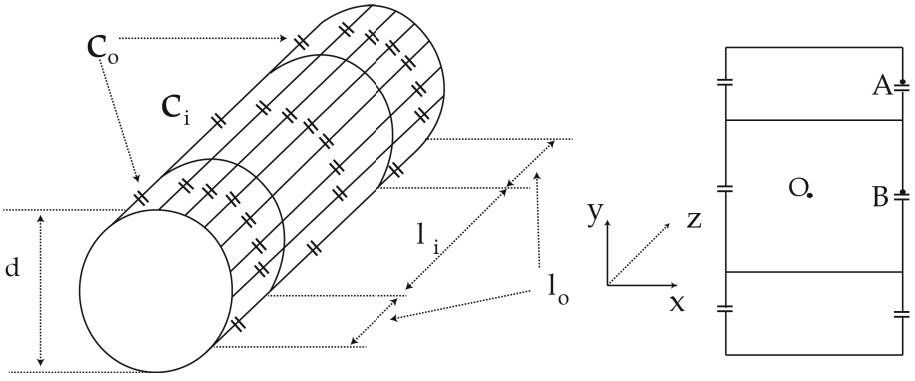


Fig. 2. Structure of a four-ring dual tuned birdcage coil. d: diameter of the coil; l_i: length of the inner birdcage; l_o: length of the outer birdcage; C_i: capacitors on the inner birdcage; C_o: capacitors on the outer birdcage. The figure on the right is the longitudinal plane view. Point A: feeding and sampling point for the outer birdcage; Point B: feeding and sampling point for the inner birdcage; Point O: sampling point in the center of the coil.[1]

(^1H) having a 127.7 MHz frequency at $3T$. A 1 volt signal at this frequency is applied to force resonance in the coil. The resonance was achieved by phase shifting the signal for each of the legs of the coil. The phase shift is equal to the angular distance between the various legs of the coil.

The homogeneity of the field within the coil is determined by calculating the mean ($\overline{f}$) and the standard deviation (f_{sd}) of the field inside the coil. That is, the total volume within the coil is sectioned into cubic segments (e.g., cubic centimeters), and the individual segment homogeneity values are calculated and then combined to form the coil's overall homogeneity. The magnetic field is considered homogeneous if the standard deviation of the field throughout the birdcage coil is less than 10% of the mean field strength[6]. The optimization problem here involves maximization of the volume within the coil where the field is homogeneous.

3 Experiment Setup

The problem representation used for the genetic algorithm is an array of real-valued numbers. Table 1 describes the range of values each gene (or parameter) can take. The two genes d and l represent the diameter and length of the inner coil as in l_i in Fig. 2. The outer coil length for this article is treated as 0. The limits for the diameter and length of the coil were based on the anatomy of the patient the coil is designed to target, that is, the patient's head (note, other medical diagnostic procedures may call for targeting the patient's knee or ankle, for example). The number of legs in the coil is represented by n, although in the representation it is treated as a real valued number, the fractional part is ignored when the coil is built within the simulation.

The range for the strip width s_w was chosen based on the largest possible multiple of $5mm$ that can fit in the coil with the smallest radius and most number of legs. This is given by $\frac{\pi.d_{max}}{n_{max}} = \frac{\pi.180mm}{20} = 28.27mm \approx 25mm$, which is the maximum width of the copper strip that can be used in a coil with the smallest diameter. This limit was imposed to avoid situations where the copper strip would leave no space between the legs due to overlap. The limit on the multiple of 5mm, was for ease of physical design. This may be later tweaked to handle more and finer divisions. Also, another assumption is that the copper strip width defines the strip width for every leg and ring in the coil. This means when the coil is constructed only one type of copper strip can be used to make the physical coil. This parameter could be redefined for each of the legs and end rings of the coil, but, here we consider a single value for simplicity.

The capacitance that would go on the coil for either the highpass or the lowpass design is indicated by C in Table 1. The capacitance is assumed to be same on every leg of the coil to keep the coil simple. However, band-pass or hybrid coils require two capacitance values, one for each of the lowpass and highpass parts of the coil circuit. The range of the capacitor is set between 50pF and 500pF, so that the tolerance level on the physical model does not impact the field. If the capacitance is very small then it becomes harder to tune the capacitor to the exact value. The coils are very sensitive to capacitance and a slight change in capacitance can cause distortions in the magnetic field.

The homogeneity of the coil was determined by computing the standard deviation and average of the magnetic field magnitude within a sphere of diameter $0.8d$, where d is the diameter of the coil being tested. This was done to ensure that at least the region in the center of the coil was homogeneous, since the center of the coil was the point of focus in this experiment. Any coil that did not fit the criteria was marked as an infeasible design. This was done to ensure that coils which fit the criteria had higher fitness and priority as the search algorithm progressed.

Table 1. Coil representation and gene description

Index	**Symbol**	**Minimum**	**Maximum**	**Description**
1	d	180mm	250mm	Diameter
2	l	100mm	450mm	Coil Length
3	n	3	20	Legs or Rungs on the coil
4	s_w	5mm	25mm	Width of Copper Strip
5	C	50pF	500pF	Capacitance
6	Δd	0mm	20mm	Change in diameter
7	Δl	0mm	25mm	Change in length
8	ΔC	0pF	25pF	Change in capacitance

The mutation operation in the GA was controlled using Δ parameters, which were also evolved along with the other parameters for the geometry of the coil. The Δ parameters define the range that the corresponding coil parameter can take during the mutation process. For example, suppose the coil length is 250mm

and Δl is 20mm, then after the mutation step the coil can have a length in the range [250mm, 270mm]. Further, the mutation was also forced to choose values which are multiples of 5mm (for length measures) or 5pF (for capacitance), this was done for ease of physical design. So, from the previous example, the coil can take either one of the following lengths, 250mm, 255mm, 260mm, 265mm, and 270mm.

The birdcage coil design objective here is to maximize the fitness function. There are two parts to the fitness function, the volume part f_v and the homogeneity part f_h. Maximizing the fitness function causes the volume of the homogeneous region to expand. Similarly, the homogeneity of a volume is obtained by calculating the mean and standard deviation of the field strength within the chosen volume [3] [4]. The homogeneity factor f_h increases as the region becomes more and more homogeneous. The upper limit to the fitness function is the volume of the coil. This is because volume f_v can vary between 0 to the volume of the coil, and f_h varies between 0 and 1. f_v contains the volume of the region inside the coil where the standard deviation of the magnetic field is less than 10% of the average field strength.

$$f_h = \frac{f_{sd}}{\overline{f}} \tag{4}$$

In (4), f_{sd} is the standard deviation, $\overline{f}$ is the mean of the magnetic field homogeneity. The fitness value is calculated by taking the product of f_v and f_h, (5) represents the fitness function. The feasibility of an individual coil is determined by values of f_{sd} and $\overline{f}$, a feasible birdcage coil will have f_{sd} less than 10% and $\overline{f}$ is greater than some pre-specified lower bound based on the source voltage.

$$f = \frac{f_h}{f_v} \tag{5}$$

The time domain magnetic field characteristics of a coil were simulated by a software package called MEEP (MIT Electromagnetic Equation Propagation)[7]. The package uses CTL (Control Language) input for design specifications to simulate coils. To calculate the fitness of the coil the field strength throughout the overall volume of the coil has to be determined. This is done by converting the individual design of a coil into a CTL File. The thickness of the copper strip was set to a typical value of 2mm; this was done for the ease of simulation. A 1 Volt sinusoidal signal was applied to each leg of the coil. To induce forced resonance the frequency was set to the resonant frequency of Hydrogen (^1H) and the signal was phase shifted for each leg, the phase shift was equal to the angular distance between each leg within a 3T MRI machine. At the end of the simulation the magnetic field strength for the coil was stored in an HDF5 file, typically several megabytes for an individual coil. The contents of this file were extracted using the H5dump utility in order to provide a human readable format. For this experiment the following field values were extracted from the file, B_x, B_y and B_z. The extracted content is a vector, whose magnitude is determined and is used to calculate the mean and standard deviations of the magnetic field.

The material used for simulation had a dielectric number of 12. The space within the coil was filled with air. The boundaries of the simulation space were made with PML (Perfectly Matched Layer) with unit size. The coil design simulation is a very computational and data intensive procedure taking many hours for the evaluation of a single fitness function.

Since a simulator was used to compute a coil's magnetic field, the time required to compute the fitness function of each individual was fairly large (but less than the time required to build and test a physical birdcage coil, and much less expensive; not to mention the evaluation time in an MRI machine). This means that the population size, in the GA, had to be small in order to get timely results. So a population size of 20 was used in the experiments. The genetic algorithm was configured to be generational, and to use tournament selection with a tournament size of 4. One point, two point, uniform and arithmetic crossover were implemented, and the crossover technique was chosen at random when creating a child. The probability of crossover was set to 0.9. The mutation operator used the ranges defined in Table 1 to update various parameters by uniformly sampling for a value in the range. The operator was forced to choose values that were multiples of 5 for ease of physical design.

The termination criterion was met when the coil's homogeneous volume was greater than or equal to 70% of total coil volume. That is, when the homogeneity of the coil reaches at least 70% of the volume of the coil the genetic algorithm is stopped. This could be further tightened for future experiments. After termination, each individual was converted to a CTL file for simulation with MEEP. The limit on homogeneity was set to 10%, and the optimization rules required the coil to be homogeneous as much as possible.

4 Preliminary Experiment Results

The experiment was set up and run as a preliminary test, and each evaluation took between 30 minutes and six hours, depending on each coil simulated. The result files generated from the simulation were of HDF5 format and analyzed using HDF5 scientific data format analysis software. From the preliminary runs it was observed that the initial set of randomly generated coils had homogeneity of 15% to 30% and the homogeneity improved with each generation by a small degree. For the test run, the population size was set to five individuals only. The computing resources available were not sufficient to perform a complete run of the genetic algorithm process since the average duration of each complete simulation lasted several hours. However, it provided insight into the complexity of the problem. It was observed that there were two areas that needed attention, first limitations of the simulation environment and second infeasible coils. The simulator limitations were basically due to either digitization issues or resource issues. The connection between the signal source and the coil was designed as a point source. But due to digitization errors sometimes the contact would not form correctly. This was fixed by adding a spread area to the point contact source to ensure that contact was made. The resource issue with the simulator was

due to the resolution used to simulate the coils. Higher resolution yielded more accurate models but the simulation time was too long, lower resolution took less time but the results were not accurate. Also, the physical resources consumed when generating result files added another constraint to the resolution value. After several trails a reasonable size for the FDTD cells was finally obtained, which was $1\,mm \times 1\,mm \times 1\,mm$.

From the preliminary runs it appeared that the simulation time would be a factor impacting the duration of the heuristic search process. To address this issue, the simulations were done with a frequency domain solver because it reduced to computation needed to analyze the time domain data and extract the field strength. In the time domain, the changes in the field over a duration of time are captured, for this project the strength of the field for a particular frequency was needed. In the frequency domain, we can observe a particular frequency and avoid capturing snapshots over time. To reduce the overhead of creating and working with CTL files, an in-house simulator was developed for this experiment. The current version of the simulator is designed with a highpass birdcage coil as the base model[2]. Our simulator considerably improved the simulation time for each coil to be within about one to twenty minutes. Some of the coil configurations and their fitness values are shown in Table 2.

Table 2. Results from the in house simulator. The objective is to reduce the fitness value.

#	d **mm**	l **mm**	n	s_w **mm**	C **pF**	**Fitness**
1	240	435	2	20	160	0.00013776
2	245	420	14	10	150	0.00014870
3	240	450	8	10	440	0.00019616
4	250	440	8	25	430	0.00020716
5	160	245	9	20	160	0.00100174
6	175	240	19	10	325	0.00156397
7	190	210	18	25	385	0.00187419
8	190	255	19	15	445	0.00213201

Genetic algorithms may be able to generate coils that are more efficient than coils designed by iterative trial and error methods, and we fully expect this to be true. This is because of the ability of the genetic algorithm to explore a large search space relatively quickly. Further study is required to design genetic algorithm techniques that can converge faster with fewer numbers of generations. One of the methods is to create individuals across the diagonal of the hyperspace formed by the various variables. This diagonal is the principal diagonal that starts with all values at a minimum and extends to all values at their maximum. To further improve the chances, an approximation of the search space will be developed based on the results of randomized coils and the principal diagonal coils. This approximation will aid in identifying the region where the search would be more promising. The work in progress is to develop a baseline procedure for designing radio frequency birdcage coils for MRI diagnosis of problems within

the human head. After generating satisfactory results, the coil models will be extended to account for more advanced geometry, such as concentric, overlapped, axially spiral, and cross-sectional spiral type coil designs. In addition, we plan to investigate alternative search schemes such as particle swarm optimization for this important real-world problem.

References

1. Duan, Y., Peterson, B.S., Liu, F., Brown, T.R., Ibrahim, T.S., Kangarlu, A.: Computational and experimental optimization of a double-tuned 1h/31p four-ring birdcage head coil for mrs at 3t. Journal of Magnetic Resonance Imaging 29(1), 13–22 (2009), http://dx.doi.org/10.1002/jmri.21509
2. Giovannetti, G., Landini, L., Santarelli, M., Positano, V.: A fast and accurate simulator for the design of birdcage coils in mri. Magnetic Resonance Materials in Physics, Biology and Medicine 15, 36–44 (2002), http://dx.doi.org/10.1007/BF02693842 10.1007/BF02693842
3. Hugon, C., D'Amico, F., Aubert, G., Sakellariou, D.: Design of arbitrarily homogeneous permanent magnet systems for nmr and mri: Theory and experimental developments of a simple portable magnet. Journal of Magnetic Resonance 205(1), 75–85 (2010), http://www.sciencedirect.com/science/article/pii/S1090780710001059
4. Ibrahim, T.S., Tang, L.: Insight into rf power requirements and b1 field homogeneity for human mri via rigorous fdtd approach. Journal of Magnetic Resonance Imaging 25(6), 1235–1247 (2007), http://dx.doi.org/10.1002/jmri.20919
5. Jin, J.M.: Electromagnetic Analysis and Design in Magnetic Resonance Imaging. CRC Press (1999) No. ISBN-13: 9780849396939
6. Liu, W., Collins, C.M., Delp, P.J., Smith, M.B.: Effects of end-ring/shield configuration on homogeneity and signal-to-noise ratio in a birdcage-type coil loaded with a human head. Magnetic Resonance in Medicine 51(1), 217–221 (2004), http://dx.doi.org/10.1002/mrm.10683
7. Oskooi, A.F., Roundy, D., Ibanescu, M., Bermel, P., Joannopoulos, J.D., Johnson, S.G.: MEEP: A flexible free-software package for electromagnetic simulations by the FDTD method. Computer Physics Communications 181, 687–702 (2010)
8. Taflove, A.: Computational Electrodynamics: The Finite-Difference Time-Domain Method, 1st edn. Artech House Publishers, Norwood (1995)

A Comparative Study of Content Statistics of Coding Regions in an Evolutionary Computation Framework for Gene Prediction*

Javier Pérez-Rodríguez, Alexis G. Arroyo-Peña, and Nicolás García-Pedrajas

Department of Computing and Numerical Analysis, University of Córdoba, Spain
{javier.perez,i52arpea,npedrajas}@uco.es
http://www.cibrg.org/

Abstract. The determination of which parts of a DNA sequence are coding is an unsolved and relevant problem in the field of bioinformatics. This problem is called gene prediction or gene finding, and it consists of locating the most likely gene structure in a genomic sequence.

Taking into account some restrictions, gene structure prediction may be considered as a search problem. To address the problem, evolutionary computation approaches can be used, although their performance will depend on the discriminative power of the statistical measures employed to extract useful features from the sequence.

In this study, we test six different content statistics to determine which of them have higher relevance in an evolutionary search for coding and non-coding regions of human DNA. We conduct this comparative study on the human chromosomes 3, 19 and 21.

1 Introduction

A genomic sequence is a string composed of four nucleotides, A, T, G and C, which codify in groups of three, called codons, the amino acids that form necessary proteins for all organisms to live. The gene is the structure that codifies the proteins. In eukaryotes, the coding sequence is usually broken by non-coding sequences, called introns, that are excised during the transcription in a process called splicing. The coding pieces are called exons. In this way, the eukaryotic gene begins with a first exon, then any number of pairs of introns and exons, and ends with a last exon, which finishes with a stop codon. There are also eukaryotic genes composed by a single exon. This study is concerned only with eukaryotic gene recognition because of its importance and difficulty. The boundary between an exon and an intron is called a splice donor site, and the boundary between an intron and an exon is called a splice acceptor site. The actual gene also has sequences of nucleotides before the start codon and after the stop codon, which are known as the untranslated terminal regions (UTRs). However, it is usual in

* This work has been financed in part by the Excellence in Research Projects P07-TIC-2682.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 206–215, 2012.

gene recognition to use the term "gene" when referring only to the coding part of the gene because that part determines the protein structure.

The terms gene recognition, gene structure prediction or gene finding, in addition to others, are used when determining which parts of a sequence are coding and constructing the whole gene from its start site to its stop codon [1,11]. Gene recognition is one of the most important open problems in the field of bioinformatics. The subtle sources of evidence and the many pitfalls associated with the problem make gene recognition in eukaryotes one of the most challenging tasks in this field.

Gene recognition may be considered as a search problem, in which many sources of evidence are combined in a scoring function that must be maximized to obtain the structure of a probable gene. There are two basic approaches to gene structure prediction. Homology-based approaches search for similar sequences in databases of known genes. These methods are usually called extrinsic methods. It is clear that only genes that are somewhat similar to known genes can be identified in this way. The second set of methods are usually known as intrinsic methods and consist of two basic approaches: *ab initio* and *de novo* methods. Both are based on identifying the features that characterize a coding region and/or the functional sites and using them to find the correct structure of the unknown genes. *Ab initio* methods use only the information of the genome to be annotated (the target genome), whereas *de novo* methods add information from one or more related genomes (the informant genomes).

The methods for obtaining and using information from genomes are many, including neural networks, hidden Markov models and conditional random fields. These methods are used to recognize genomic sequence patterns that are characteristic of splice donor and acceptor sites, translation initiation sites (TIS) and termination sites, and/or features specific of coding regions. Some of these methods are also used to find other important parts of genes, such as promoters, untranslated terminal regions or polyadenylation sites.

Evolutionary computation approaches have achieved promising results [12]. However, their performance is highly dependent on the coding statistics used to characterize the sequence. Many such statistics have been proposed in the past. The use of a good set of coding statistics directly impacts the recognition accuracy. Therefore, the selection of a set of optimal features for classification of coding and non-coding sequences should be performed carefully. Although systematic comparisons of feature performance exist, a study using evolutionary approaches has yet to be conducted.

The statistics used can be classified into two groups: content statistics and site statistics. Content statistics are used to determine whether a certain region is coding, whereas site statistics are used to find the functional sites of the sequences. However, this last task is better accomplished using classifiers such as support vector machines [3]. Thus, we focus this study on content statistics.

The main objective of this study is to demonstrate which content statistics can be efficiently used for gene prediction using evolutionary computation. Six measures of different types were chosen: in-frame hexamers, bulk hexamers,

position asymmetry, length distributions, average mutual information and local compositional complexity, all of which are tested in an evolutionary framework for gene recognition.

The remainder of the paper is organized as follows: Section 2 describes the content measures addressed in the study, Section 3 presents the evolutionary algorithm that will be used as a gene finder system, Section 4 states the experimental setup and results, and Section 5 provides the conclusions of the study.

2 Content Statistics

This paper presents a study of several statistical features that are often used for discriminating between coding and non-coding DNA regions. The feature strength was compared inside an evolutionary algorithm framework. A total of six content statistics were studied: in-frame hexamers, bulk hexamers, position asymmetry, local compositional complexity, exon length distributions and average mutual information.

In-frame hexamers and bulk hexamers statistics are related to codon usage bias. Position asymmetry is related to the asymmetric feature of the distribution of nucleotides at the three codon positions. Local compositional complexity is based on richness of exon information. Length distribution is based on the different average lengths of exons and introns. Average mutual information statistics are related to the correlation between nucleotides at a certain distance. A description of these statistics follows:

1. In-frame hexamer frequency (IFH). It has long been known that synonymous codons are not used with equal frequencies and that different organisms differ in their patterns of codon usage. The in-frame hexamer score for the interval starting at nucleotide i and ending at j, $IF_6(i,j)$ is calculated as follows:

$$IF_6(i,j) = max \begin{cases} \sum_{k=0,3,6,\ldots,j-6} & ln(\frac{f_k}{F_k}) \\ \sum_{k=1,4,7,\ldots,j-6} & ln(\frac{f_k}{F_k}) \\ \sum_{k=2,5,8,\ldots,j-6} & ln(\frac{f_k}{F_k}) \end{cases} \quad (1)$$

   where f_k is the frequency, in the table of in-frame hexamers in human coding sequences, of the hexamer starting at position k in the interval. In the calculation, F_k is the frequency of the same hexamer in a random population based on the base composition of the sequence. Hexamers with occurrences equal to those expected by composition have $IF_6 = 0$, those preferred have a positive score and those avoided, a negative score. First exons, last exons, internal exons, unique exons and introns hexamers are evaluated from different frequency matrices.
2. Bulk hexamer (BHEX). It could be useful to consider a genomic sequence as a series of overlapping hexamers without considering its reading frame. This approach is useful because the log-likelihood statistic applied to in-frame hexamers in exons can be applied in this form to intron sequences as well. It has been determined [2] that the frequencies of hexamers can differ greatly

in functionally different classes of sequences and can be used to discriminate among them. The bulk hexamer score for the interval i to j belonging to sequence class X, BHEX(i, j) is defined as follows:

$$BHEX_x(i,j) = \sum_{k=i}^{j-6} ln(\frac{f_k^x}{F_k}) \tag{2}$$

where f_k is the frequency of hexamer k in sequence class x and F_k is the frequency of hexamer k in a random population based on the base composition of the locus in question. Thus, this statistic is similar to the IFH test except that the log-likelihood is summed over all reading frames and that each sequence class is compared with its own hexamer frequency table.

3. Position asymmetry (PA). Let $f(b,r)$ be the relative frequency of nucleotide b at codon position r. Let $f(b) = \sum_{r=1}^{3}(f(b,r))/3$ be the average frequency of nucleotide b at the three codon positions, and define the asymmetry in the distribution of nucleotide b as the variance of this frequency, i.e., $asym(b) = \sum_{r=1}^{3}(f(b,i) - f(b))^2$, and the PA of the sequence is defined as follows [5]:

$$PA = asym(A) + asym(C) + asym(G) + asym(T) \tag{3}$$

4. Local compositional complexity (LCC). In non-coding regions of the eukaryotic genomes is typical to find large amounts of repetitive DNA sequences. In contrast, coding regions hold information richness. This property, quantified by the Shannon information [13], is a measure of the local redundancy of the sequence. We can define a local compositional complexity of a segment as a statistical property to distinguish between coding and non-coding sequences. This local entropy measure, LCC, using a segment of nucleotides of length L, is defined as:

$$LCC = -\sum_{k=\{A,C,G,T\}} (\frac{N_k}{L}) log_2(\frac{N_k}{L}) \tag{4}$$

where N_k is the number of times base k occurs in the segment of nucleotides of length L [9].

5. Length distribution (LD). This statistic provides evidence that introns and exons have different extreme and average lengths [6]. Even within the class of exons, the length distributions of first, last and internal exons all differ significantly from one another. This information can be used as evidence that an interval is a member of a particular sequence type by looking up the frequency of the interval length in a table. A low score can be used as strong evidence that the interval is not part of the actual solution.
6. Average mutual information (AMI). The correlation $(\rho_{ij}(k))$ between nucleotide i and nucleotide j at a distance of k nucleotides can be calculated as $\rho_{ij}(k) = p_{ij}(k) - p_i p_j$, where p_i and p_j are the probabilities of nucleotides i and j in the sequence and $p_{ij}(k)$ is the probability in the sequence of the pair of nucleotides i and j at a distance of k nucleotides [7]. Thus, for each

distance k, 16 different individual correlations can be calculated. A measure that summarizes all individual correlations at a given distance k is the mutual information function,

$$I(k) = \sum P_{i,j}(k) log_2(\frac{P_{i,j}(k)}{P_i P_j}) \tag{5}$$

The mutual information $I(k)$ quantifies the amount of information that can be obtained from one nucleotide about another nucleotide at a distance k. In coding DNA, $I(k)$ oscillates between two values, whereas in non-coding DNA, $I(k)$ is rather flat. The two values between which $I(k)$ oscillates in coding DNA in the in-frame mutual information are called I_{in} at distances $k = 2, 5, 8, \ldots$, and the out-of-frame mutual information I_{out} at $k = 4, 5, 7, 8, \ldots$. To reduce the pair of numbers I_{in} and I_{out} to a single quantity, we compute the average mutual information (AMI) as follows:

$$AMI = \frac{I_{in} + 2I_{out}}{3} \tag{6}$$

3 Evolutionary Gene Recognition Framework

As previously mentioned, gene recognition can be considered as a search problem, and therefore an evolutionary approach can be taken. The evolutionary system developed to test each feature used a standard genetic algorithm the fitness function of which is a sum of different scores of each segment in the gene structure. The search is conducted considering only a source of evidence corresponding to the signals that identify coding regions. The application of evolutionary computation to gene structure prediction is based on a two-step procedure. The first step consists of reducing the search space. In a second step, we develop the evolutionary algorithm to find the most likely gene structure.

The first step is devoted to limiting the search space. At first sight, a gene appears to be a structure delimited by two sites, the start and stop codons. In a second approach, between these two boundaries are two different substructures, exons and introns. Exons and introns are bounded by a donor and an acceptor splice site, with the exception of the first exon, which begins with the start site, and the last exon, which finishes with the stop signal. If we consider no restrictions, the search space would be huge, and any method would be likely to fail. The common approach for reducing the search space is to limit the putative start, splice and stop sites to the most probable ones. In our system, we used support vector machines (SVMs) with a string kernel function [10] for site recognition. String kernels are appropriate and specific function kernels to handle character sequences.

The SVM model is obtained from a dataset of known genes using a supervised training method. Specifically, we consider every actual site and canonical but false site that was found in the set as positive and negative patterns for

training, respectively. Because we had a class-imbalance problem, a random undersampling algorithm was used.

Random undersampling consists of randomly removing instances from the majority class until a certain criterion is reached. In most studies, instances are removed until both classes have the same number of instances. Several studies comparing sophisticated undersampling methods with random undersampling [8] have failed to establish a clear advantage of the former.

With this method, setting a certain threshold, we obtain a set of possible start, stop and splice sites that will be the only ones considered. Table 1 shows sequence window lengths used for each site to train the SVM. These values were experimentally calculated as a compromise solution between accuracy and execution time. It must be noted that these windows are considerably longer that those used previously in other studies [4]. Our experiments found that the use of smaller windows has a large negative impact on the performance of the gene recognizer.

Table 1. SVM string kernel window lengths for the different sites

Site	Length	Upstream Offset	Downstream Offset
TIS	50 bps	10 bps	37 bps
Donor	52 bps	24 bps	25 bps
Acceptor	52 bps	24 bps	25 bps
Stop codon	103 bps	50 bps	50 bps

A second reduction of the search space is achieved by taking into account the constraints in the gene structure:

- The exons do not overlap.
- The gene starts and finishes with an exon.
- An intron must be flanked by two exons.
- A gene can be composed of only one exon.

The initial population of the genetic algorithm is randomly obtained from the possible exons that satisfy the above constraints. The codification of each individual is a string of integers that represents the sites of the gene. The initial population is divided into a number of subpopulations in which the individuals are placed depending on their number of exons. An individual may migrate to another population when a mutation modifies its length.

For each generation, several actions are performed:

- **Selection.** Selection is performed using binary tournament to avoid excess selective pressure and to handle the maintenance of the balance among the numbers of exons of the individuals. The subpopulations with different numbers of exons are kept with the same number of individuals. Elitism is applied to avoid losing the best solutions to that point.

- **Crossover.** Crossover is conducted randomly by recombining the exons of two parents to obtain two offspring. The offspring substitute their parents.
- **Mutation.** This operator consists of randomly modifying exon boundaries or removing, adding or exchanging an exon. After mutation, the individual is checked to ensure its viability.

The algorithm is a standard generational genetic algorithm [14] with the particularity that we have forced an even distribution of the lengths of the genes, in terms of number of exons, to avoid a premature convergence to a suboptimal solution.

3.1 Relationship among Features and Fitness Function

In our evolutionary approach, we have used a fitness function that is as simple as possible. It must be remembered that our main objective is to develop a system that proves the validity of different statistical features for evolutionary gene recognition. We are not creating a system that is competitive with current gene recognizers, which are very complex programs.

The fitness of the possible solution is calculated, evaluating each segment in terms of its type. In this way, first exons, last exons, internal exons, unique exons and introns are evaluated differently. For all of these types, there is a common measure, content statistics we are testing in turns. It has long been known that synonymous codons are not used with equal frequencies and that different organisms differ in their patterns of codon usage.

Additionally, for each site we calculated a score based on a trained SVM. Each segment (i, j) of the sequence will have a score that depends on its type, k: first exon, f; internal exon, e; internal intron, i; last exon, l; and unique exon, u. The score of a segment of type k, from j to l, $S_k(j, l)$, represents the probability of the segment of being of type k. The score is the value of the content statistic for a segment of nucleotides using the statistics methods explained. The fitness for an individual is the sum of the scores for all of its sequences:

$$F = S_f(h, j) + S_i(j + 1, n) + S_e(n + 1, l) + S_i(l + 1, m) + \ldots + S_l(p + 1, n) \quad (7)$$

3.2 Evaluation Measures

Accuracy is not a useful measure for imbalanced data. In the prediction of gene structure, the ratio of coding against no coding regions is heavily imbalanced, and therefore other measures must be used. Several measures have been developed that consider the imbalanced nature of the problems. Given the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN), we can define the following two basic measures: sensitivity $Sn = \frac{TP}{TP+FN}$ and specificity $Sp = \frac{TN}{TN+FP}$.

These are common measures in any class-imbalance problem. There are also measures specific to the gene recognition task. One of the most commonly used measures of this type is the correlation coefficient, CC:

$$CC = \frac{(TP)(TN)-(FP)(FN)}{\sqrt{(PP)(PN)(AP)(AN)}} \tag{8}$$

where PP are the predicted positives, AP the actual positives, PN the predicted negatives and AN the actual negatives. CC will be our main measure of the performance of the method.

The source code used for all methods, in C and licensed under the GNU General Public License, as well as the partitions of the datasets, are freely available on request from the authors.

4 Experimental Setup and Results

The system was tested for each feature on the chromosomes 3, 19 and 21 of the human genome. Chromosome 3 has 4 contigs where 1497 genes are distributed, chromosome 19 has 1767 genes distributed on 4 contigs and chromosome 21 has 312 genes on 8 contigs. The size of the whole dataset is more than 200 million nucleotides. For evaluating the performance of the system with each statistic feature, we used k-fold cross-validation, where k is the total number of contigs.

Four different SVMs were trained to predict translation initiation sites, donor splice sites, acceptor splice sites and stop codons. Then, we performed the described evolutionary processes. The evolution to obtain the gene structure of each test sequence was performed for 1000 generations. The populations consisted of 200 individuals.

The results obtained in the experimental process are shown in tables 2 and 3. G+C content determines the sequence of the statistical features. To optimize the performance of the system, the training set was divided into three groups, depending on the G+C content of the sequences. Table 2 shows the percentage of correctly predicted exons, overlapped predicted exons, completely wrong exons predicted and the percentage of actual exons predicted correctly, actual exons predicted overlapped and exons not found. Table 3 shows the correlation coefficient, sensitivity and specificity at the nucleotide level.

The first interesting finding is that the relative performance of the content statistics does not depend on the G+C content. Regarding the behavior of each statistic, both tables show a similar trend. IFH is clearly the most discriminant measure, achieving better accuracy than all of the remaining measures at the exon level and at the nucleotide level. BHEX is the second measure in discriminative power. AMI and LCC show a medium performance at both levels, whereas LD shows poor performance also at both levels. PA shows a medium performance at the exon level, but it is the worst-performing statistic at the nucleotide level.

Table 2. Comparative statistical feature results for chromosomes 3, 19 and 21 at the exon level

	G+C content	CC	Sn	Sp
AMI	Low	0,285	0,270	0,558
	Medium	0,289	0,273	0,578
	High	0,335	0,301	0,741
BHEX	Low	0,423	0,295	0,806
	Medium	0,366	0,274	0,670
	High	0,415	0,306	0,790
IFH	Low	0,475	0,355	0,865
	Medium	0,451	0,331	0,797
	High	0,453	0,341	0,835
LCC	Low	0,253	0,691	0,435
	Medium	0,227	0,669	0,269
	High	0,242	0,695	0,348
LD	Low	0,185	0,463	0,345
	Medium	0,193	0,490	0,298
	High	0,223	0,426	0,394
PA	Low	0,335	0,987	0,427
	Medium	0,320	0,936	0,335
	High	0,287	0,976	0,284

Table 3. Comparative statistical feature results for chromosomes 3, 19 and 21 at the nucleotide level

		Predicted			Actual		
	G+C content	Exact	Overlap	False	Exact	Overlap	False
AMI	Low	0,333	0,455	0,212	0,295	0,347	0,357
	Medium	0,322	0,472	0,205	0,177	0,221	0,600
	High	0,237	0,535	0,226	0,149	0,345	0,504
BHEX	Low	0,433	0,437	0,130	0,147	0,423	0,428
	Medium	0,463	0,383	0,152	0,191	0,159	0,648
	High	0,477	0,397	0,125	0,191	0,294	0,513
IFH	Low	0,530	0,352	0,118	0,247	0,223	0,528
	Medium	0,529	0,361	0,108	0,220	0,125	0,654
	High	0,543	0,361	0,095	0,234	0,169	0,595
LCC	Low	0,261	0,643	0,095	0,234	0,169	0,595
	Medium	0,205	0,614	0,180	0,142	0,412	0,445
	High	0,197	0,633	0,168	0,133	0,649	0,216
LD	Low	0,091	0,818	0,091	0,037	0,348	0,614
	Medium	0,085	0,848	0,065	0,027	0,258	0,714
	High	0,072	0,777	0,149	0,046	0,342	0,610
PA	Low	0,056	0,873	0,071	0,023	0,327	0,656
	Medium	0,024	0,925	0,050	0,015	0,320	0,663
	High	0,136	0,834	0,028	0,083	0,341	0,574

5 Conclusions

In this study, we have investigated various content statistics that are commonly used to obtain evidence of coding regions in DNA sequences. These measures have been studied inside an evolutionary computation framework for gene recognition. This study has concluded that IFH is the most relevant measure. The superior performance of this measure has been confirmed at both the nucleotide and the exon level. The study has been performed considering three levels of G+C content.

In future research, it would be interesting, although computationally quite expensive, to perform the same study by grouping the measures instead of testing each measure in isolation.

References

1. Brent, M.R., Guigó, R.: Recent advances in gene structure prediction. Current Opinion in Structural Biology 14, 264–272 (2004)
2. Claverie, J., Sauvaget, I., Bougueleret, L.: k-tuple frequency analysis from intron/exon discrimination to t-cell epitope mapping. Methods Enzymology 183, 237–252 (1990)
3. García-Pedrajas, N., Pérez-Rodríguez, J., García-Pedrajas, M.D., Ortiz-Boyer, D., Fyfe, C.: Class imbalance methods for translation initiation site recognition in dna sequences. Knowledge-Based Systems 25, 22–34 (2012)
4. Gross, S.S., Do, C.B., Sirota, M., Batzoglou, S.: CONTRAST: a discriminative, phylogeny-free approach to multiple informant de novo gene prediction. Genome Biology 8, R269.1–R269.16 (2007)
5. Guigó, R.: DNA composition, codon usage and exon prediction. In: Bishop, M. (ed.) Genetic Databases, pp. 53–80. Academic Press (1999)
6. Hawkins, J.D.: A survey of intron and exon lengths. Nucleic Acids Research 16, 9893–9908 (1988)
7. Herzel, H., Große, I.: Measuring correlations in symbolic sequences. Physica A 216, 518–542 (1995)
8. Japkowicz, N.: The class imbalance problem: significance and strategies. In: Proceedings of the 200 International Conference on Artificial Intelligence (IC-AI 2000): Special Track on Inductive Learning, Las Vegas, USA, vol. 1, pp. 111–117 (2000)
9. Konopka, A.K., Owens, J.: Complexity charts can be used to map functional domains in DNA. Genetic Analysis, Techniques and Applications 7(2), 35–38 (1990)
10. Lodhi, H., Saunders, C., Shawe-Taylor, J., Christiani, N., Watkins, C.: Text classification using string kernels. Journal of Machine Learning Research 2, 419–444 (2002)
11. Mathé, C., Sagot, M.F., Schiex, T., Rouzé, P.: Current methods of gene prediction, their strengths and weaknesses. Nucleic Acids Research 30(19), 4103–4117 (2002)
12. Pérez-Rodríguez, J., García-Pedrajas, N.: An evolutionary algorithm for gene structure prediction
13. Shannon, C.E., Weaver, W.: The Mathematical Theory of Communication. The University of Illinois Press, Urbana (1964)
14. Syswerda, G.: A Study of Reproduction in Generational and Steady-State Genetic Algorithms. In: Rawlins, G. (ed.) Foundations of Genetic Algorithms, pp. 94–101. Morgan Kaufmann (1991)

Enhancing Point Clouds Accuracy of Small Baseline Images Based on Convex Optimization

My-Ha Le, Andrey Vavilin, Sung-Min Yang, and Kang-Hyun Jo

Graduated School of Electrical Engineering,
University of Ulsan, Ulsan, Korea
{lemyha,andy,smyang}@islab.ulsan.ac.kr, acejo@ulsan.ac.kr

Abstract. This paper proposes a method for enhancing accuracy of point clouds which are generated from small baseline of sequence images. The main contributions are threefold: First, the constraints of image pair-wise are computed based on invariant feature. The correspondence problem is solved by iterative method which remove the outlier. To avoid the disadvantage of incremental structure from motion, the global rotation of cameras are estimated by a robust method in the second step. These global rotations are fed to the point clouds generation procedure in next (third) step. In contrast with bundle adjustment which can gain local minima of back-projection error in L_2-norm, the proposed method utilized error minimization in L_∞-norm to triangulate accurately 3D points recast in quasiconvex optimization form. The simulation results will demonstrate the accuracy of this method from large view scene images in outdoor environment.

Keywords: SIFT, correspondence, RANSAC, global rotation estimation, convex optimization.

1 Introduction

3D reconstruction of objects in large view scene is one of important process in various applications of virtual environment, scene planning and navigation of autonomous mobile robot. Some progress has been made in the 3D reconstruction which is obtained during the last few years but still there is no methods satisfy the requirement of high accurate structure. Also, some of them needed a large amount of work done by hand or apparatus, such as laser radar, and airborne light detection and ranging. They are usually expensive and require much more time for data acquisition.

In recent years many well known algorithms have been developed for 3D reconstruction and motion estimation, which can roughly be devised into several categories, namely methods using bundle adjustment [1], methods based on factorization [2-4] and hierarchical methods [5-6]. In the first group, multi-view structure from motion start by estimating the geometry of two views. These structure will be used to estimate the pose of the adjacent camera. The quality of the reconstruction depend heavily on the initial structure of first pair cameras [7], [8].

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 216–225, 2012.

Another disadvantage of this method is the drift problem [9]. The expensive computation and acummulate errors increase in iterative process of adding new camera of this sequence method. This is also the strong weak point of visual SLAM (Simultaneous Localization and Mapping) based on bundle adjustment. In the second group, the missing data and sensitivily to outliers is the main problem. It is well studied by some author groups, e.g. in [4]. In the third group, the input images will be arrange in the hierarchical tree and they are processed from root to the top.

Without using any addition device, e.g. laser sensor out of calibrated images from single camera, our proposed method overcomes some disadvantages mentioned above. It is much cheaper and compact. The flow chart of proposed method can be seen in figure 1. Using monocular camera, images are acquired from objects of the scene. SIFT algorithm [10], [11] is applied to find invariant feature and matching for each pair of views in combinatorial form. The estimation of fundamental matrix and intrinsic parameters of camera is computed base on 8-points RASAC algorithm [12] and calibration method [13] respectively. Essential matrix is derived from computed fundamental matrix and above calibration information. The initial pair-wise rotation will be obtained using the method from Horn [14]. The global camera rotation are computed based on graph-based sampling scheme [15]. After obtaining camera rotation matrix in global cordinate, we find the high accuracy point clouds by applying convex optimization [16]. The L_∞-norm is ultilized to minimized the back-projection error [17]. Finally, the structure of objects will be generated.

This paper is organized into 5 sections. The next section describes camera motion in pair-wise constraint and estimate global camera rotation method. Section 3 presents point clouds generation in convex optimization form. We also explain how to measure and minimize residual error in L_∞-norm. Experiments are showed in section 4. Finally, paper is finished with conclusions and point out future works discussion in section 5.

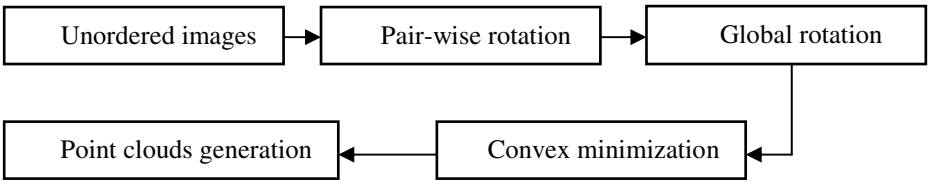


Fig. 1. General proposed scheme

2 Rotation Registration

In order to compute the global camera rotation, first, we find the essential matrix which describes the relative position and orientation of image pair-wise. This matrix is estimated using correspondences point with outlier removal and intrinsic parameters of camera. Second, the global rotation constraints will be computed based on

these local pair-wise constraints by averaging motion method. In this section, we explain what are camera model, how to extract and match invariant features robustly with outlier removal as well as how an essential matrix can be decomposed to recover the camera motion, and, thereby, camera projection matrices.

2.1 Camera Model Introduction

We use the projective geometry throughout this paper to describe the perspective projection of the 3D scene onto 2D images [12]. This projection is described as follows:

$$x = PX \tag{1}$$

where P is a 3×4 projection matrix that describes the perspective projection process, $X = [X, Y, Z, 1]^T$ and $x = [x, y, 1]^T$ are vectors containing the homogeneous coordinates of the 3D world coordinate, respectively, 2D image coordinate.

When the ambiguity on the geometry is metric, (i.e., Euclidean up to an unknown scale factor), the camera projection matrices can be put in the following form:

$$P = K[R \mid -Rt] \tag{2}$$

with t and R indicating the translation and rotation of the camera and K, an upper diagonal 3×3 matrix containing the intrinsic camera parameters.

$$K = \begin{bmatrix} f_x & s & u_x \\ & f_y & u_y \\ & & 1 \end{bmatrix} \tag{3}$$

where f_x and f_y represent the focal length divided by the horizontal and vertical pixel dimensions, s is a measure of the skew, and (u_x, u_y) is the principal point.

2.2 Features Extraction and Matching

There are many kind of features are considered in recent researches in feature extraction and matching problem including Harris [18], SIFT, PCA-SIFT, SURF [19-20], GHOL [21], etc. SIFT is first presented by David G Lowe in 1999 and it is completely applied in pattern recognition problem in 2004. As we know on paper experiments, his proposed algorithm is very invariant and robust for feature matching with scaling, rotation, or affine transformation. According to those conclusions, we utilize SIFT feature points to find correspondent points of image pairs. The SIFT algorithm are described through these main steps: scale-space extrema detection, accurate keypoint localization, orientation assignment and keypoint descriptor. The result of correspondence points will be used to compute fundamental matrix described in the next step.

2.3 Pair-Wise Camera Motion Constraint

The result of correspondence point in previous step will be used to compute fundamental matrix. The epipolar constraint represented by a 3x3 matrix is called the fundamental matrix, F. Also, this method based on two-view geometry theory which was studied completely in [12]. According to theory, once the intrinsic parameters of the cameras are known, the fundamental epipolar constraint above can be represented algebraically by a 3x3 matrix, called the essential matrix. We can form the matrix E:

$$E = K'^{T} F K \tag{4}$$

Here E is essential matrix, K' and K are intrinsic parameters of camera 1 and 2. In the case of using monocular camera, we have $K' = K$. The projection matrix of the first camera P is set follow this equation:

$$P = K[I \mid 0] \tag{5}$$

The second projection matrix is found from four possible choices: $P = (UWV^T \mid +u_3)$ or $P = (UWV^T \mid -u_3)$ or $P = (UW^TV^T \mid +u_3)$ or $P = (UW^TV^T \mid -u_3)$, where U and V are found from SVD decomposition of E. u_3 is the last column of U and W.

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

Only one of these four choices is possible for the second camera. We can find it by testing whether a reconstructed point lies in front of both cameras.

2.4 Global Camera Rotation Estimation

Some common approaches for camera registration were proposed by several author groups. As in [22], authors used cycles in the camera graph and Bayesian framework for incorrect pair-wise detection. Another linear solution based on least square method was presented in [23]. Whereas in [24] branch-and-bound search over rotation space was used to determine camera orientation. In this paper, we apply a robust rotation averaging method as proposed in [15]. The results proclaim that graph-based sampling scheme efficiently removes outliers in the individual relative motions based on RANSAC scheme. A short description for this method is presented as follow: given the relative rotation R_{ij}, we want to find a robust method to compute a set of all camera rotations R_k in the global coordinate, e. g,

$$R_i = R_{ij} R_j \tag{7}$$

According to the reference paper, the algorithms of this method might be summarized as below.

Algorithm1: RANSAC Algorithm for Robust Motion Averaging

Input: $\{R_{ij1}, R_{ij2},..., R_{ijn}\}$ (n relative motions)

Distance threshold D_0 and number of trials T

Output: $R_g = \{R_2, R_3,..., R_N\}$ (N image global motion)

- *Set G: view-graph of relative motions*
- *Generate minimum spanning trees* $MSTe = MST(G)$
- *Solve for global motion* R_{mst} *using* $MSTe$
- *Count the number of relative motions within distance* D_0 *of* R_{mst}
- *Repeat for* T *trials and select* MST *with maximal count*
- *Discard relative motions that are outliers for this* MST
- *Using the inliers solve for* R_g *using algorithm 2.*

Algorithm 2: Relative Motion Averaging

Input: $\{R_{ij1}, R_{ij2},..., R_{ijn}\}$ (n relative motions)

Output: $R_g = \{R_2, R_3,..., R_N\}$ (N image global motion)

Set R_g to an initial guess

Repeat

$$\Delta R_{ij} = R_j^{-1} R_{ij} R_i$$

$$\Delta r_{ij} = \log(\Delta R_{ij})$$

$$\Delta v_{ij} = vec(\Delta r_{ij})$$

$$\Delta \delta = D^* \Delta V_{ij}$$

$$\forall k \in [2, N], R_k = R_k \exp(\Delta v_k)$$

Until $\|\Delta \delta\| < e$

where D^* is the pseudo-inverse.

3 Point Clouds Triangulation

In this section, triangulation with known rotation consistency will be recast as quasi-convex optimization problem. Some author groups proposed methods using L_∞-norm [17], [25] or L_∞-norm combined L_1-norm [26] instead of L_2-norm in minimizing the residual error of measured feature and back-projection of 3D points. It is easy to figure out that solving the L_2-norm for more than two cameras is a hard non-convex problem. It can yield local minima instead of single global minimum if error is minimized in L_∞-norm. The next section will formulate and solve the triangulation problem by using bisection convex optimization method.

3.1 Problem Formulation

Let $P_i, i = 1,2,...,m$ are the m known cameras and u_i are the projection of point U in 3D space (both are expressed in homogeneous coordinates). The problem of finding U given the camera matrices and image points is triangulation. In the ideal case (absence of noise), the triangulation is ordinary. In the noise case, the back-projection of point U to image plane doesn't coincide with u_i. Thus, we must find point U that its projection is nearest u_i, i.e., minimizes the cost function:

$$\sum_{i=1}^{m} d(u_i, P_i U)^2 \tag{8}$$

here $d(\cdot,\cdot)$ represents the geometric distance between two points in the image. Argument in [23] point out that the L_2-norm error of this cost function in three view triangulation creates three local minima whereas the L_∞-norm create single minimum. In this paper, we use the procedure which is similar to [17]. The known rotation problem will be described in detail as follow: consider the camera matrix P we try to solve the minimization problem:

$$\min \max_i d(u_i, P_i U(x))$$

$$\text{subject to } \lambda_i(x) > 0, i = 1,2,...,m \tag{9}$$

here $\lambda_i(x)$ is the depth of the point in image i. It is easy to realize that the square image distance is a rational function of *x*:

$$d(u, PU(x))^2 = \frac{f_1(x)^2 + f_2(x)^2}{\lambda(x)^2} \tag{10}$$

Where $f_1(x)^2$, $f_2(x)^2$ and $\lambda(x)^2$ are affine functions in x and with coefficients determined by u and P.

Remark 1. *The problem* $\min \max_i d(u_i, P_i U(x))$ *has some convexity properties. Thus, this problem can be solved by quasiconvex optimization method.*

3.2 Bisection Based Quasiconvex Optimization Solver

Suppose that γ is an upper bound of the objective function in problem (9). According to theory in [26], this problem can be formulated again with form:

$$\min \gamma$$

$$\text{Subject to } \left\| f_{1i}(x), f_{2i}(x) \right\| \le \gamma \lambda_i(x)$$

$$\lambda_i(x) > 0, i = 1,2,...,m \tag{11}$$

if γ is consider unknown, equation (11) can be rewritten in second order cone programs (SOCP) feasibility problem form:

$$\begin{aligned} &\text{find } x \\ &\text{Subject to } \left\| f_{1i}(x), f_{2i}(x) \right\| \le \gamma \lambda_i(x) \\ &\lambda_i(x) > 0, i = 1,2,...,m \end{aligned} \tag{12}$$

Assume that the optimal γ^* is lower than some threshold of γ_u pixels, then evidently $\gamma^* \in [0, \gamma_u]$. Until now, the typical convex feasibility problem solving is applied. The detail algorithm is presented bellow.

Algorithm 3: Bisection based quadiconvex optimization solver

Given: *optimal value* $f_0^* \in [\gamma_l, \gamma_l]$ *and tolerance* $\varepsilon > 0$

Repeat

1. $\gamma := (\gamma_l + \gamma_u)/2$

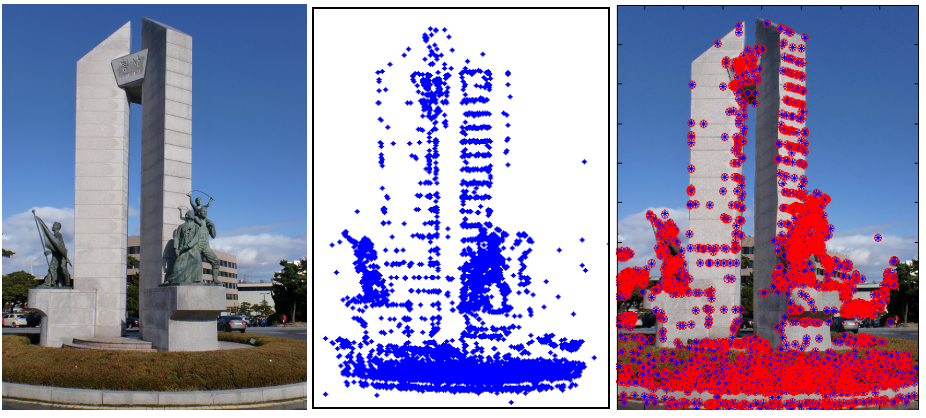
2. Solve the convex feasibility problem

3. ***if*** *feasible* $\gamma_u := \gamma$, ***else*** $\gamma_l := \gamma$

Until $\gamma_u - \gamma_l \le \varepsilon$

4 Experiments

In this section we describe some experiments to evaluate the proposed method. The main objects are large view scenes in outdoor environment. The dataset images are acquired by perspective camera. This proposed algorithm was simulated on Intel(R) Core(TM) i5 CPU 750@2.67 GHz with 3GB RAM under Matlab environment and MOSEK toolbox [27]. In the first experiment, we try to reconstruct a complicated structure which has three statues stand in front of three pillars. This object is named "Rotary statue" and 10 images of size [2048x1536] were used. Here are some main parameters of result: SIFT features in each image are about 8000 keypoints, the pairwise matching after RANSAC outlier removal are about 2000 points and the triangulated cloud of structure are 9005 points. Figure 2(a) and (b) are one of views of dataset image as well as point clouds of "Rotary statue". It is easy to recognize that the back-projection of 3D point into image quite near together as in figure 2(c), i.e., this proposed method was demonstrated high accuracy in point cloud triangulation. The zoomed view of this error can be seen clearly in figure 3. According to this figure, the error is less than one pixel. Similarly, we also checked our proposed method with two more datasets of buildings in large view scene. The point clouds result of 16 images of "International building" and 15 images of "Brown building" are presented in figure 4 and 5, respectively.



(a) (b) (c)

Fig. 2. Point clouds of "Rotary statue". (a) and (b) are one of views of dataset and point clouds. (c) is back-projection of point clouds into image (marked in red color "o") and original image points (marked in blue color "*").

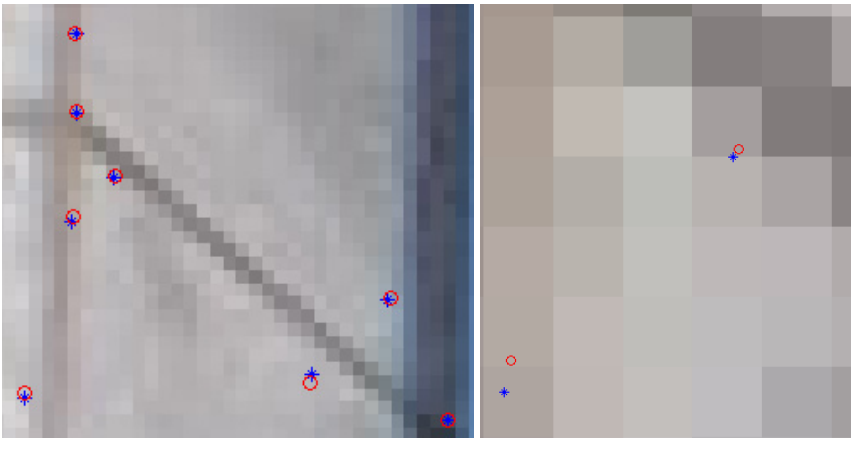
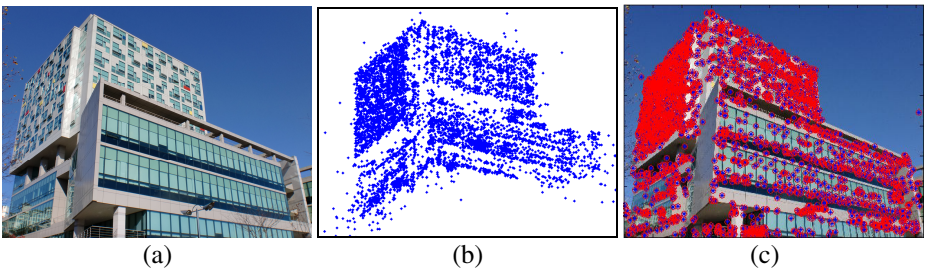
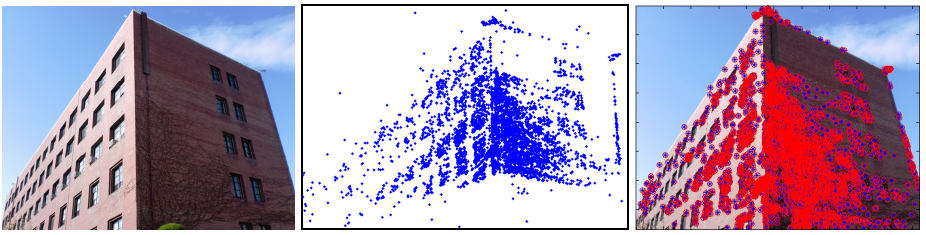


Fig. 3. The zoomed views of back-projection error



(a) (b) (c)

Fig. 4. Point clouds of "International building". (a) and (b) are one of views of dataset and point clouds. (c) is back-projection of point clouds into image (marked in red color "o") and original image points (marked in blue color "*").



(a) (b) (c)

Fig. 5. Point clouds of "Brown building". (a) and (b) are one of views of dataset and point clouds. (c) is back-projection of point clouds into image (marked in red color "o") and original image points (marked in blue color "*").

5 Conclusions

Outdoor scene reconstruction from multiple views based on convex optimization method is presented on this paper. Some advantage points can be realized through our explanation. First, we avoid using bundle adjustment which is used L_2-norm in back-projection error minimization. This method can lead to local minima. Instead, we apply convex optimization in our algorithm. By utilizing L_∞-norm for minimization back-projection error, we can triangulate the high accuracy structure of object. Second, in the global camera rotation estimation, graph-based sampling scheme according to RANSAC spirit result in robust results. Our future woks focus on the comparison of this method with L_1 combined L_∞-norm. Also, we will improve and develop this method by upgrading to dense point clouds using stereo rid or Omni-directional camera in outdoor scene. The last ambition is application of this method to real scene modeling systems.

Acknowledgment. This work was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the Human Resources Development Program for Convergence Robot Specialists support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2010-C7000-1001-0007).

References

1. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle Adjustment – A Modern Synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) ICCV-WS 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000)
2. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. Int. Journal of Computer Vision (1992)
3. Sturm, P., Triggs, B.: A Factorization Based Algorithm for Multi-Image Projective Structure and Motion. In: Buxton, B.F., Cipolla, R. (eds.) ECCV 1996. LNCS, vol. 1065, pp. 709–720. Springer, Heidelberg (1996)

4. Tardif, J.-P., Bartoli, A., Trudeau, M., Guilbert, N., Roy, S.: Algorithms for batch matrix factorization with application to structure from motion. In: Conf. Computer Vision and Pattern Recognition (2007)
5. Nistér, D.: Reconstruction from Uncalibrated Sequences with a Hierarchy of Trifocal Tensors. In: Vernon, D. (ed.) ECCV 2000, Part I. LNCS, vol. 1842, pp. 649–663. Springer, Heidelberg (2000)
6. Gherardi, R., Farenzena, M., Fusiello, A.: Improving the efficiency of hierarchical structure-and-motion. In: Conf. Computer Vision and Pattern Recognition (2010)
7. Thormählen, T., Broszio, H., Weissenfeld, A.: Keyframe Selection for Camera Motion and Structure Estimation from Multiple Views. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004, Part I. LNCS, vol. 3021, pp. 523–535. Springer, Heidelberg (2004)
8. Torr, P., Fitzgibbon, A., Zisserman, A.: The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. Int. Journal of Computer Vision (1999)
9. Cornelis, K., Verbiest, F., Van Gool, L.: Drift detection and removal for sequential structure from motion algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence (2004)
10. Lowe, D.: Object recognition from local scale-invariant features. In: International Conference on Computer Vision, pp. 1150–1157 (1999)
11. Lowe, D.: Distinctive Image Features from Scale-Invariant Interest Points. International Journal of Computer Vision 60, 91–110 (2004)
12. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2004)
13. Bouguet, J.-Y.: Camera Calibration Toolbox for Matlab, `http://www.vision.caltech.edu/bouguetj/calib_doc/index.html`
14. Horn: Recovering baseline and orientation from essential matrix (1990)
15. Govindu, V.M.: Robustness in Motion Averaging. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3852, pp. 457–466. Springer, Heidelberg (2006)
16. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
17. Kahl, F., Hartley, R.: Multiple view geometry under the L_∞-norm. IEEE Transations on Pattern Analysis and Machine Intelligence (2008)
18. Harris, M.S.: A combined corner and edge detector. In: Alvey Vision Conference, Manchester, UK, pp. 147–151 (1998)
19. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
20. Juan, L., Gwun, O.: A Comparison of SIFT, PCA-SIFT and SURF. International Journal of Image Processing 3(5) (2010)
21. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. Trans. Pattern Analysis and Machine Intelligence (2005)
22. Zach, C., Klopschitz, M., Pollefeys, M.: Disambiguating visual relations using loop constraints. In: Conf. Computer Vision and Pattern Recognition (2010)
23. Martinec, D., Pajdla, T.: Robust rotation and translation estimation in multiview reconstruction. In: CVPR, pp. 1–8 (2007)
24. Hartley, R., Kahl, F.: Global optimization through rotation space search. In: IJCV (2009)
25. Sim, K., Hartley, R.: Removing outliers using the L_∞-norm. In: Conf. Computer Vision and Pattern Recognition, New York City, USA, pp. 485–492 (2006)
26. Dalalyan, A., Keriven, R.: L_1-penalized robust estimation for a class of inverse problems arising in multiview geometry. In: Neural Information Processing Systems (2009)
27. The MOSEK optimization toolbox for MATLAB manual, `http://www.mosek.com`

Subset Quadratic Assignment Problem

He Jiang and Yan Hu

School of Software, Dalian University of Technology
Dalian, China
{jianghe,huyan}@dlut.edu.cn

Abstract. Variants of QAP have become the hot lines in research on NP-Hard combinatorial optimization problems. There exists a new kind of problem which can't be modeled as QAP or its existing variants, in applications such as hospital layout whose facility must be assigned to one location in some predefined subset. This new problem is modeled as the subset QAP (SQAP) in this paper. We show that SQAP is NP-Hard and no $\varepsilon-$approximation algorithm exists for it ($\varepsilon > 0$). Furthermore, we prove that it can be determined in polynomial time whether a feasible solution exists or not, by proving its equivalence to perfect matching problem on bipartite graph.

Keywords: Subset Quadratic Assignment Problem, NP-Hard, Heuristic, Perfect Matching.

1 Introduction

The quadratic assignment problem [1-2] (QAP) is a typical NP-hard combinatorial optimization problem with numerous real-life applications arising in hospital layout, keyboard layout, and VLSI design, etc. Under the theory of computational complexity [3], there exists no exact algorithm in polynomial running time for NP-hard problems unless P=NP. Therefore, many heuristic algorithms have been designed and applied to QAP, including ant colony optimization [4-5], evolutionary computation [6], genetic programming [7], simulated annealing [8], neural network [9], tabu search [10-11], GRASP [12], etc. In addition to QAP, there is also a hot line of research working on many QAP variants, driven by numerous application demands. Burkard et al. [13-14] proposed biquadratic assignment problem (BiQAP) in 1994; Knowles and Corne et al. [15-16] proposed multiobjective QAP (mQAP) in 2002; Chi-Guhn Lee et al. proposed generalized QAP problem (GQAP), in which multiple facilities are allowed be placed in one location; Billionnet et al. [18] studied quadratic semi-assignment problem (QSAP); Hahn et al. [19] worked on quadratic 3-dimensional assignment problem (Q3AP). More detailed survey on QAP and its variants can be found in [2].

However, there are many application scenarios which cannot be modeled by QAP and existing QAP variants. For example, in the hospital layout problem, the hospital plans to assign departments to appropriate units. The optimal plan should minimize the overall cost of time or distance regarding the traffic of humans amongst hospital

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 226–230, 2012.

departments. Moreover, the locations of departments should also meet some extra constraints, e.g. the emergency rooms should be close to the gate, while the sickrooms must be located at locations with abundant sunshine. This hospital layout problem is not a standard QAP problem, or an existing QAP variant (e.g., Q3AP or mQAP). Similar applications of this new problem can also be found in the wireless sensor deployment process. For instance, body sensor network has been widely used to detect the body functionalities of patients. Different sensor nodes should be placed at some fixed positions to detect certain body functionality status, and there is some information flow around deployed sensor nodes. The locations of sensor nodes should satisfy some constraints: sensors to detect heart functionalities can only be placed near the heart or the pulse, while sensors to detect body temperature should be placed under the armpit or inside the mouth [22-23].

In this paper, we issue the new problem as a QAP variant, namely the subset QAP (SQAP). It is easy to figure out that the new applications related to SQAP have one commonality: facilities could only be placed in a subset of all the locations. In this paper, we present the formal definition of SQAP. Then, we analyze the computational complexity of SQAP, and prove that SQAP is an NP-hard problem, for which no $\varepsilon-$ approximate polynomial time algorithm exists. In addition, we show that the existence of SQAP feasible solutions is equivalent to perfect matching on bipartite graph, and can be solved in polynomial time.

2 Preliminaries

In this section, we present some definitions of QAP and SQAP. In addition, some related definitions about graph matching problems are also given so as to investigate the computational complexity of the existence of feasible solutions to SQAP.

Definition 1. Given n facilities $F=\{f_1,f_2,...,f_n\}$ and n locations $L=\{l_1,l_2,...,l_n\}$, let $A=(a_{ij})_{nxn}$ be the flow matrix for facilities (a_{ij} stands for the flow cost between facility f_i,f_j), $B=(b_{ij})_{nxn}$ be the distance matrix for locations (b_{ij} stands for the distance between location l_i,l_j), the goal of QAP is to find a permutation $\pi:\{1,2,...,n\}\to\{1,2,...,n\}$ that minimizes the overall cost among facilities. More formally, the overall cost of a permutation π is calculated by $c_\pi=\sum_{j=1}^{n}\sum_{i=1}^{n}a_{ij}b_{\pi(i)\pi(j)}$.

Definition 2. For each facility $f_i\in F$ in a QAP problem, let L_i ($L_i\subseteq L$) be the set of locations that f_i can be placed, a feasible solution for SQAP is a permutation $\pi_S:\{1,2,...,n\}\to\{1,2,...,n\}$, such that $l_{\pi_S(i)}\in L_i$ for each facility $f_i\in F$. Let Π_S be the set of all feasible solutions to SQAP, then the goal of SQAP is to find a feasible solution π_S^* which minimizes the overall cost, i.e., $c_{\pi_S^*}=\min_{\pi_S\in\Pi_S}c_{\pi_S}$.

Definitions about graph matching and perfect matching, bipartite graph, associated graph are given below.

Definition 3. Given a graph $G=(V(G),E(G))$, for $M\subseteq E(G)$, $\forall e_i,e_j\in M$, if e_i,e_j is not adjacent, then M is a matching of graph G; the two vertices of an edge in M is called matching under M; each vertex in M is called M matched; when each vertex in G is M matched, then M is a perfect matching.

Definition 4. Given a graph $G=(V(G),E(G))$, if $V(G)=X\bigcup Y, X\bigcap Y=\varnothing$, and any two vertices in X are not adjacent, and any two vertices in Y are not adjacent, then G is a bipartite graph.

Definition 5. Given an instance I of SQAP problem, the associated graph for I is $G_I=(V(G_I),E(G_I))$, where $E(G_I)=\{(f_i,l_j)\mid f_i\in F,l_j\in L_i,1\le i,j\le n\}$ and $V(G_I)=F\bigcup L$.

Obviously, according to definition 4 and 5, let $X=F,Y=L$, we have:

Proposition 1. Given an instance I of SQAP, the associated graph $G_I=(V(G_I),E(G_I))$ is a bipartite graph.

3 Complexity Analysis of SQAP

The computational complexity of SQAP is investigated in this section. We also show that a feasible solution of SQAP can be determined in polynomial time using an existing algorithm for perfect matching problem on bipartite graphs.

Sahni et al. [1] proved that QAP problem is NP-hard in 1976, and no $\varepsilon-$approximate polynomial time algorithm exists for QAP ($\varepsilon>0$). Since QAP is a special case of SQAP(let $L_i=L, i\in\{1,2,\cdots,n\}$), according to the theory of computational complexity [3], we have:

Proposition 2. SQAP is an NP-hard problem, and no $\varepsilon-$approximate polynomial time algorithm exists for SQAP ($\varepsilon>0$).

Lemma 1. Given a SQAP instance I, the problem of finding its feasible solutions is equivalent to finding perfect matching in the associated graph G_I.

Proof: Necessary condition: Given a SQAP instance I and a feasible solution $\pi_S:\{1,2,...,n\}\to\{1,2,...,n\}$, i.e. for each $\forall f_i\in F$, we have $l_{\pi_S(i)}\in L_i$. Then we can construct edge set $M_I=\{(f_i,l_{\pi_S(i)})\big|1\le i\le n\}$, it is easy to verify that M_I is a perfect matching on G_I.

Sufficient condition: According to the definition of G_I, for any edge (f_i,l_j) in its perfect matching, we have $f_i\in F,l_j\in L_i$. Then a permutation π_S can be constructed, where $\pi_S(i)=j$. Then π_S is a feasible solution for instance I. □

Lemma 2. Given a SQAP instance I, the perfect matching problem on the associated graph G_I can be solved in polynomial time.

Proof: According to proposition 1, the associated graph of instance I is a bipartite graph. It has been proved that the perfect matching problem on bipartite graphs can be solved by polynomial time algorithms. Edmonds proposed Hungarian algorithm [24] with time complexity $O(|E||V|)$ in 1965. Hopcroft et al. [25] designed perfect matching algorithm with time complexity $O(|E|\sqrt{|V|})$; Alt et al. [26] proposed a fast matrix scanning algorithm with time complexity $O(|V|^{1.5}\sqrt{|E|/\log|V|})$ for perfect matching on dense bipartite graphs. In summary, the perfect matching on G_I is solvable in polynomial time. □

According to Lemma 1 and Lemma 2, we have the following theorem.

Theorem 1. The feasible solutions of SQAP can be determined in polynomial running time.

4 Conclusions and Future Work

A new variant of QAP, namely SQAP, is issued and studied in this paper. SQAP originates from real-life applications. It imposes extra constraints on standard QAP that each facility can only be placed at a specified subset of locations. Based on the formal definition of SQAP, we prove that SQAP is NP-hard and the feasible solutions to SQAP can be determined in polynomial time. In the future work, we plan to design an efficient heuristic algorithm for solving SQAP in polynomial time. The new algorithm can employ an existing algorithm for perfect matching to construct a feasible solution and then some optimization strategies can be applied to improve on this solution.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China under grants 61175062 and 60805024, and the "Software + X" funding of Dalian University of Technology.

References

1. Sahni, S., Gonzalez, T.: P-complete approximation problems. Journal of the ACM 23, 555–565 (1976)
2. Eliane, M.L., Nair, M.M.A., Paulo, O.B.N., Peter, H., Tania, Q.: An analytical survey for the quadratic assignment problem. European Journal Operational Research 176(2), 657–690 (2007)
3. Garey, M.R., Johnson, D.S.: Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman, San Francisco (1979)
4. Zou, P., Zhou, Z., Chen, G.L., Jiang, H., Gu, J.: Approximate-Backbone guided fast ant algorithms to QAP. Journal of Software 16(10), 1691–1698 (2005)
5. Gambardella, L., Taillard, E.D., Dorigo, M.: Ant colonies for the QAP. Journal of the Operations Research Society (1999)
6. Nissen, V.: Solving the quadratic assignment problem with clues from nature. IEEE Transactions on Neural Networks 5(1), 66–72 (1994)

7. Drezner, Z.: A new genetic algorithm for the quadratic assignment problem. Informs Journal on Computing 15(3), 320–330 (2003)
8. Tsuchiya, K., Nishiyama, T., Tsujita, K.: A deterministic annealing algorithm for a combinatorial optimization problem using replicator equations. Physica D: Nonlinear Phenomena 149(3), 161–173 (2002)
9. Ishii, S., Sato, M.: Doubly constrained network for combinatorial optimization. Neurocomputing 43(4), 239–257 (2002)
10. Misevicius, A.: A modification of tabu search and its applications to the quadratic assignment problem. Information Technology and Control 27, 12–20 (2003)
11. Drezner, Z.: The extended concentric tabu for the quadratic assignment problem. European Journal of Operational Research 160, 416–422 (2005)
12. Oliveira, C.A.S., Pardalos, P.M., Resende, M.G.C.: GRASP with path-relinking for the QAP. In: MIC 2003: The Fifth Metaheuristics International Conference (2003)
13. Burkard, R.E., Çela, E., Klinz, B.: On the biquadratic assignment problem. In: Pardalos, P.M., et al. (eds.) Quadratic Assignment and Related Problems. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 16, pp. 117–146. AMS, Rhode Island (1994)
14. Burkard, R.E.: Selected topics on assignment problems. Discrete Applied Mathematics 123(3), 257–302 (2002)
15. Knowles, J.D., Corne, D.W.: Towards landscape analyses to inform the design of a hybrid local search for the multiobjective quadratic assignment problem. In: Abraham, A., et al. (eds.) Soft Computing Systems: Design, Management and Applications, pp. 271–279. IOS Press, Amsterdam (2002)
16. Knowles, J.D., Corne, D.W.: Instance generators and test suites for the multiobjective quadratic assignment problem. IRIDIA technical report TR/IRIDIA/2002-25
17. Chi-Guhn, L., Zhong, M.: The generalized quadratic assignment problem. Department of Mechanical and Industrial Engineering, University of Toronto, working paper (2003)
18. Billionnet, A., Elloumi, S.: Best reduction of the quadratic semi-assignment problem. Discrete Applied Mathematics 109(3), 197–213 (2001)
19. Hahn, P.M., Kim, B.J., Hightower, W.L., Stützle, T., Kanthak, S., Samra, H., Ding, Z., Guignard, M.: The quadratic three-dimensional assignment problem: exact and heuristic solution methods. OPIM Working Report No. 04-08-02, The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania, USA (2004)
20. Eiselt, H.A., Laporte, G.: A combinatorial optimization problem arising in dartboard design. Journal of the Operational Research Society 42, 113–118 (1991)
21. Christian, S., Karl, D., Axel, F., Kurt, H.: Determining location and size of medical departments in a hospital network: a multiobjective decision support approach. Health Care Management Science 7, 63–71 (2004)
22. Potte, J.G., Kaiser, J.W.: Wireless integrated network sensors. In: Proc. of Communications of the ACM, New York, USA, pp. 551–558 (2000)
23. Akyildiz, I.F., Kasimoglu, I.H.: Wireless sensor and actor networks: research challenges. Ad Hoc Networks 2, 351–367 (2004)
24. Edmonds, J.: Matching and a polyhedron with 0-1 vertices. J. Res. NBS 69B, 125–130 (1965)
25. Hopcroft, J.E., Karp, R.M.: An $n^{5/2}$algorithm for maximum matchings in bipartite graphs. SIAM Journal on Computing 2, 225–231 (1973)
26. Alt, H., Blum, N., Mehlhorn, K., Paul, M.: Computing maximum cardinality matching in time $O(|V|^{1.5}\sqrt{|E|/\log|V|})$. Information Process Letters 37, 237–240 (1991)

Forward or Ignore: User Behavior Analysis and Prediction on Microblogging

Guanghua Song[1], Zhitang Li[1,2], and Hao Tu[1,2]

[1] College of Computer Science and Technology of HUST, Wuhan, China
[2] Network Center, Huazhong University of Science and Technology, Wuhan, China
ghsong2010@gmail.com

Abstract. There has been an enormous development in online social networks all over the world in current times. Represented by Twitter and Facebook, the wave of online social networking is bringing broad impact and changing people's lives increasingly. At the same time, the online social networks are experiencing a rapid development in china. Large numbers of Chinese Internet users are spending more and more time on online social networks. Represented by SINA Weibo, the online social networks are gradually occupying Chinese people's vision and causing widespread concern. At present, the study of online social networks has focused on Twitter and Facebook, the popular Chinese online social network SINA Weibo has not been deeply studied.

In this paper, we analyze the user's behavior on the SINA Weibo, pointing out the impact of user behavior in four key factors: the user's authority, the user's activity, the user's preferences and the user's social relations. By empirical methods, we give each factor the impact of user behavior through the likelihood. We find that the user's preferences and activity have greater impact on user behavior, while the authority of the user's social relations and values of the user's behavior also has some impact. On this basis, we present an idea with machine learning to predict the behavior of users, and use pattern classification methods to solve the prediction problem.

To the best of our knowledge this work is the first quantitative study on user behavior analysis. Changing the prediction problem into a pattern classification problem is the most important contribution of our work.

Keywords: Online social networks, Microblogging, Forwarding, Preference, User behavior analysis.

1 Introduction

Currently the most influential social network website in China is SINA Weibo. By June 30, 2011 there were 200 million users registered in the site, and the average release of daily information is more than 20 million[1]. It is an important choice of people's online social in China. SINA Weibo has become a major platform for the

[1] The SINA corporation second quarter report is available (in Chinese,2011) at http://tech.sina.com.cn/i/2011-08-18/05295944929.shtml

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 231–241, 2012.

youth to gather information and to make friends with others. Large numbers of people gather in the micro-blog every day. They make friends, post messages, share ideas and play games each other. It can be said that microblogging is becoming the online home of young people. There has been a lot of prior research done on the message diffusion and evolution of trends in Western online social networks. But, in contrast, Chinese social networks such as SINA Weibo have not been well-studied.

In this paper, we take SINA Weibo as data sources to analysis user behavior and to predict the user's behavior when they receive a new message. Our key findings are as follows.

Firstly, we analyze the user behavior on SINA Weibo, finding that there are basically between the flows of SINA Weibo and people's activity habits. We also found large number of users who are active would like to forward the view of someone else's on SINA Weibo. In other words, a small amount of users who often release information attract a large number of users' attention.

Secondly, we analyze the factors which impact user behavior on SINA Weibo, finding four main factors as follows: the user's authority, the user's activity, the user's preferences and the user's social relations. With the likelihood figure, we describe the each factor's impact on user. We find that the user's preferences and activity have greater impact on user behavior, while the authority of the user's social relations and values of the user's behavior also has some impact. Then, we solve problem of predicting user behavior with machine learning methods preliminary. Changing the prediction problem into a pattern classification problem is the most important contribution of our work.

2 Related Work

Currently there are lots of research on the online social network, and achieved some results. Ahn et al. [01] analyzed the structure of large-scale online social networks, and gave the topological properties of social networks. Ravi Kumar et al. [02] analyzed the structure and evolution of online social network, revealed that the simulation method can be used to quantitatively study on the social networks, M. Cha et al. [03] analyzed the spread of the message on Flickr, proposing a method based on measurement-driven analysis. Haewoon Kwak et al. [04] analyzed the twitter, pointing out that the twitter of has not only the general social networking features, but also has some of the characteristics of the media. Castellano C et al. [05] reviewed the various models of transmission dynamics of public opinion, and revealed how to apply these models. WuFang and Huberman [06] proposed the model of public opinion based on the analysis of social networks, the transmission network model. Researchers have also analyzed the structure of various Chinese online social networks [7] [8] [9] [10]. Louis Yu et al. [11] analyzed Chinese social media and evaluated how the trending topics in China related to the news media. Guo et al. [12] take measurement and analysis on SINA Weibo. They found that the following degree and follower degree in SINA Weibo followed the power-law distribution in certain range.

Information diffusion discusses the characteristics of the news spreading in the social networks and related issues. Liben-Nowell et al [13] take a more

comprehensive description of the problem. They summary the issues and the existing work, and describe some of the challenges faced. As information diffusion involving a wide range of content, many other fields of knowledge can also be used for reference. For example, in Kossinets et al. [14] work, author firstly cluster the social network data, then generate the tree of dissemination, eventually propose and describe a probabilistic model. Another work about information dissemination is literature [15], the author analysis the information dissemination in the blog space environment, and made a prediction model for information diffusion. As a broad perspective, SINA Weibo and other micro-blogging site can be regarded as an extension of traditional blog type sites and extension of some of the literature research ideas will be adopted and extended in this paper.

At present, the study of modeling information dissemination focuses on the idea to use random process to carry out such as Markov process. In the Song et al [16] work, they designed a diffusion process and propose an information flow model. Chakrabarti et al [17] invented the viral propagation method. They attempt to predict the infected population with the non-supervised learning methods. Other information diffusion model with unsupervised learning methods includes [18] and [19]. Most methods take invariably decay of spread the message as a basic assumption, and the attenuation factor is a crucial model parameters. For example, in Leskovec, etc. [13] and Parshani et al [19] work, they usually set the decay factor to a smaller value artificially, and subsequent model validation is based on extensive numerical simulation method. It makes these models difficult in other real scenarios.

3 Data Acquisition

SINA Weibo offers an Application Programming Interface (API) that is easy to crawl and collect data. We crawled and collected profiles of all users on SINA Weibo from July 1st to October 20th, 2011.We collected information from each user's profile, including: user ID, nickname, following, following's ID, follower, follower's ID, user cities and time lines. Specific data format is as shown in Table 1:

Table 1. The data structure of user PROFILE

Parameter	**uid**	**name**	**location**	**flw**
Description	user ID	nickname	user cities	following
Parameter	**flw_num**	**flwer**	**flwr_num**	**tp**
Description	count	follower	count	timestamp

We also collected the data of the information properties which users posting, including: the information publisher ID, information ID, forwarding number, comment number, information content and time lines. Specific data format is as shown in Table 2:

Table 2. The data structure of user PROFILE

Parameter	**id**	**sid**	**rts**	**rts_gr**
Description	user ID	info ID	forwarding	rate
Parameter	**cmts**	**cmts_gr**	**cets**	**dtme**
Description	comments	rate	contents	datetime

4 User Behavior Analysis

We first analyze the user's forward behavior. We start with the perspective of empirical analysis. All data are from our database.

We can observe a very intuitive phenomenon: some specific message post by the specific user more easily spread widely in the network, while other news is hardly to attract users' attention. Information dissemination on the microblogging determined by multiple factors, such as the user's activity, the message content, the user's preferences. Therefore, the study why some information is more able to spread on the microblogging will be very useful. We first analyze the characteristics of the individual user and message, which will help us understand why some users will forward the message, while others do not. We want to answer a major question: what factors determine the user's forwarding behavior? On this basis, can we predict the user's forwarding behavior based on available information?

4.1 The User's Authority

Up to now, SINA Weibo has about 200 million registered users, and the number of information released daily is about 20 million. Such huge data makes the analysis and research on information dissemination mechanism is particularly difficult. In order to describe the different user's position in SINA Weibo, We use the Hits algorithm to calculate the quantitative value of each person's authority.

According to our data collected, the user's authority values can be directly read out. So, through the EM algorithm, we found that the correlation between the value of the user's authority and user behavior is weak. The user's authority and end-user's forwarding attitude relevant is as shown below Figure 1:

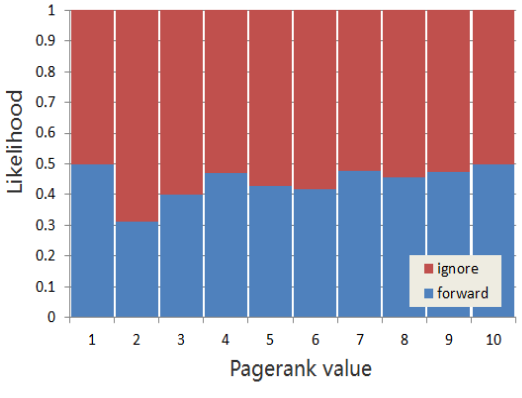


Fig. 1. The user's authority

Thus, the user's authority is relevant to the user's forward behavior, but the correlation is weak.

4.2 User's Activity

Up to now, SINA Weibo has about 200 million registered users, which number about microblogging daily release of about 20 million [8]. The data makes such a huge microblogging information dissemination mechanism analysis and research is particularly difficult. However, according to the data we have obtained, not all users have the same role and status, not all the news can spread throughout the whole network. In general, users will choose to forward the message they are interested, or just read the message without forwarding has its own tendencies.

In order to quantitatively characterize the user's activity, we introduced the concept of active value.

Definition 1: The user's activity is a user activity level on microblogging network, and its value is defined by the following formula:

$$\text{Activity} = \frac{message(u)}{total}$$

Message(u): the number of users message include posting, forwarding and commenting.

Total: the number of all message include posting, forwarding and commenting.

We collected all message as total, then we can calculate the value of each user' activity. Through the EM algorithm, we found that the degree of active users and end-user forwarding behavior is relevant, as shown below Figure 2:

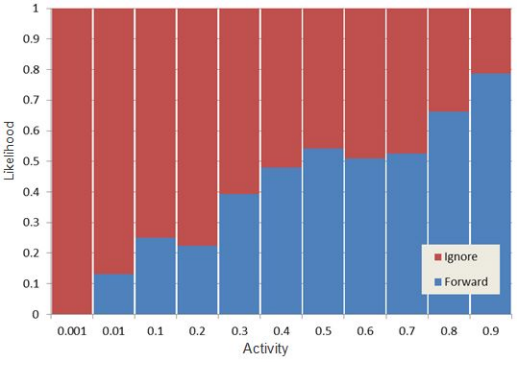


Fig. 2. The user's activity

It can be seen, there is a tight relation between the user's activity and the user's behavior of forwarding or ignoring. The user whose activity is large has a greater probability of forwarding a message, while the user whose activity is low is more inclined to ignore the message.

4.3 User's Preference

In general, each user on the micro-Bo is an independent individual. When a new message is received, the real decision whether or not to forward is the user

themselves. Therefore, the user's interests naturally to an important aspect of our study. For this part of the analysis we start from two perspectives: first discuss the user's own preferences, that is, the similarity of content, followed by discussion between the user's preferences, that is, interests and hobbies that the similarity between users.

Definition 2: The user's preference is the similarity of users interest, which includes two aspects: user VS content，user VS user. Their values are defined by the following formula:

User VS content:

$$\text{Prefer} = \frac{|U \bigcap V|}{|U \bigcup V|}$$

U: the set of user's messages include posting, forwarding and commenting.

V: the set of a specific message.

User VS user:

$$\text{Common} = \frac{|U_1 \bigcap U_2|}{|U_1 \bigcup U_2|}$$

U_1: the set of user1's messages include posting, forwarding and commenting.

U_2: the set of user2's messages include posting, forwarding and commenting.

User VS content

Under normal circumstances, when the user receives a new message he will view the content firstly, then according to their own preferences to decide whether to forward or not. The importance of the message content for users is obvious. Therefore, we need to study the similarity of the user and the message content .In order to take into account the user and the linkages between the content, we can use Formula 2 to calculate the similarity.

So, through the EM algorithm, we found that the correlation between the value of the user's preference and user behavior is strong. The user's preference and end-user forwarding attitude relevant, as shown below Figure 3:

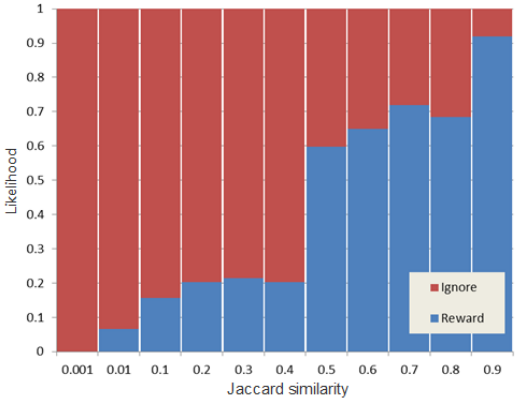


Fig. 3. The similary between user and content

It can be seen, there is a tight relation between the user's preference and the user's behavior of forwarding or ignoring. The user whose preference is large has a greater probability of forwarding a message, while the user whose preference is low is more inclined to ignore the message.

User VS User

There is another situation also attracted our attention: the similarity between message publisher and user. We know that birds of a feather flock together. In other words the people with the same interesting are more likely to be a group. There is the same situation on Micro-Bo. If there is a great similarity between the message publisher and user interests, it means that they are concerned about the message may also have a greater similarity. Therefore, in addition to message content, message publishers can also influence the user's decision on how to deal with (forward or ignore) the message.

Next we study that the user's selection has also been influenced by the news publisher. We calculate the Jaccard similarity between users and message publishers with the Formula 2. Figure 4 shows the similarity between user and message publisher, and compared the likelihood with the both cases given respective forwarding or ignoring.

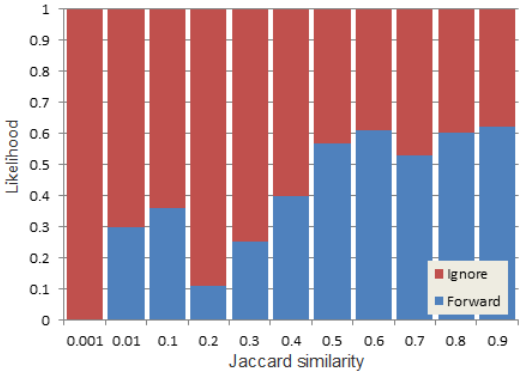


Fig. 4. The similary between user and user

We can find when the user decides to forward message they will take into account the similarity between user and publisher. If there have a greater similarity between the user and publisher, the user will be a greater probability of forwarding the message. Otherwise, if there have a smaller similarity between the user and publisher, the user will be a greater probability of ignoring the message.

4.4 User's Social Relations

Different from the normal social relations, the user's social relations needn't two-way authentication. The relationship between user and its following can be seen as a parasocial relationship. Parasocial relationship is a one-way relationship, for example,

A follows B, it means A has a greater understanding about B, at the same time, B probably has a smaller understanding about A. We can say the relation is a kind of weak relationship. Correspondingly, there is other relation named as two-way relationships, it means there is mutual concern between users. We can say the relation is a kind of strong ties.

Then, we analyze that the user's selection has also been influenced by the relationship between users. Figure 5 shows the social relation between user and message publisher, and compared the likelihood with the both cases given respective forwarding or ignoring.

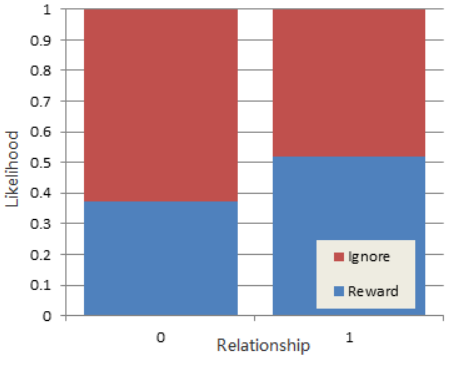


Fig. 5. User's social relations

We can find when the user decides to forward message they will take into account the relationship between user and publisher. If there have two-way relationships between the user and publisher, the user will be a greater probability of forwarding the message. Otherwise, if there have one-way relationships between the user and publisher, the user will be a greater probability of ignoring the message.

5 User Behavior Prediction

In the previous section, we set a series of statistical analysis based crawling SINA Weibo data, and reveal some potential factors which have a positive impact on user behavior. Specifically, we formalize the intuitive behavior as a series of key factors: user activity, user preferences and user of social relations and other factors. Then we calculate the likelihood distribution with the forward and ignore behavior given. Next, we will further use of these proved to have a positive effect of the factors that help identify the user's information sharing predict communication behavior. Next, we will use these factors to predict the behavior of users.

5.1 Method Introduction

Now, we roughly know the main factors affecting user behavior, then how to predict the user's behavior? The traditional method is building a model, then solving.

However, the user's behavior is a complex random process, so it is very difficult using a model to describe. Also, it is difficult to accurately describe the real behavior of the users with the model. We decided to change an idea to solve this problem. Our idea is changing the prediction problem into a classification problem, and we can use machine learning and pattern classification methods to predict user behavior.

We put the five factors mentioned above as the characteristics of machine learning, and put the user's behavior (forward or ignored) as the label, to carry out machine learning. We set up two states: 0 means ignoring, 1 means forwarding, and use support vector machine(SVM) to make pattern classification.

5.2 Experiment and Result

We partly crawled and collected profiles of users and messages on SINA Weibo from November 10th to 17th, 2011. Specific data set is as shown in Table 3:

Table 3. Experimental data sets

Total	Forward	Ignore	Sample	Testing	Accuracy
6956	*2358*	*4598*	*1000*	*5956*	*85.36%*

We take 1000 of total messages as the sample sets, and take the remaining as testing sets. Using the SVM method, the final classification accuracy rate is 85.36%. We also found that these five factors have different importance, and the importance of each factor is as shown below Figure 6:

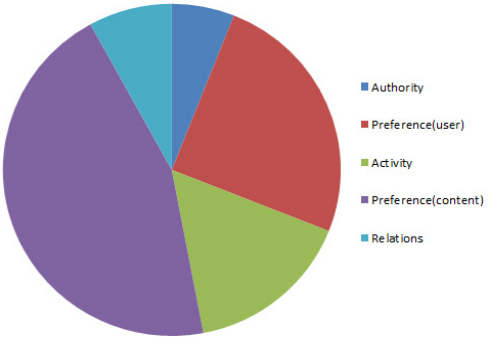


Fig. 6. The importance of each factor

It can be seen, the user preferences for content is the most influential factor when they make a decision. And the user's activity and common interests also play a significant role. Then, the influence of the user's authority and social relationships is relatively small. For us, this is a useful conclusion.

6 Conclusions and Further Work

We analyzed the users' behavior and made a prediction of SINA Weibo. We observed that there are four factors that affect the behavior of users: the user's authority, the user's activity, the user's preferences and the user's social relations. On this basis, we present an idea with machine learning to predict the behavior of users, and use pattern classification methods to solve the prediction problem. Using the SVM method, the final classification accuracy rate is 85.36%. We also find that the user's preferences and activity have greater impact on user behavior, while the authority of the user's social relations and values of the user's behavior also has some impact. Our major contribution is changing the prediction problem into a pattern classification problem, and we can solve it easily with machine learning methods.

Our research is based on the empirical, combining with the classic method for the users' behavior. However, behavior of users on the SINA Weibo is different from the traditional situation. Considering various factors, analysis different types of user behavior and predict user's decision in complex situations will be the focus of our future work.

Acknowledgment. We thank SINA for resources that facilitated the research. This research is supported by the Fundamental Research Funds for the Central Universities(HUST:2011JC067) and CNGI [CNGI-122].

References

[1] Ahn, Y.-Y., Han, S., Kwak, H., Moon, S., Jeong, H.: Analysis of topological characteristics of huge online social networking services. In: Proc. of the 16th International Conference on World Wide Web. ACM (2007)

[2] Kumar, R., Novak, J., Tomkins, A.: Structure and evolution of online social networks. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 611–617. ACM (2006)

[3] Cha, M., Mislove, A., Gummadi, K.P.: A measurement-driven analysis of information propagation in the Flickr social network. In: Proc. of the 18th International Conference on World Wide Web. ACM (2009)

[4] Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: Proceedings of the 19th International Conference on World wide web, WWW 2010, pp. 591–600 (2010)

[5] Castellano, C., Fortunato, S., Loreto, V.: Statistical physics of social dynamics. Rev. Mod. Phys. 81, 591 (2009)

[6] Wu, F., Huberman, B.A.: Social structure and opinion formation. arXiv:cond-mat/0407252 (2004)

[7] Bian, Y.: Bringing strong ties back in: indirect ties, network bridges, and job searches in china. American Sociological Review 62(3), 366–385 (1997)

[8] Bian, Y., Breiger, R., Davis, D., Galaskiewicz, J.: Occupation, class, and social networks in urban china. Social Forces 83(4), 1443–1468 (2005)

[9] Carrington, P.J., Scott, J., Wasserman, S. (eds.): Models and Methods in Social Network Analysis.Cambrige University Press (2005)

[10] Xin, M.: Chinese bulletin board system's influence upon university students and ways to cope with it. Journal of Nanjing University of Technology (Social Science Edition) 4, 100–104 (2003) (in Chinese)
[11] Yu, L., Asur, S., Huberman, B.A.: What Trends in Chinese Social Media. To appear in Proc. of the 5th SNA-KDD Workshop (2011)
[12] Guo, Z., Li, Z., Tu, H.: Sina Microblog: An Information-driven Online Social Network. In: Proc. of IEEE CW 2011, Banff, Canada (October 2011)
[13] Liben-Nowell, D., Kleinberg, J.: Tracing information flow on a global scale using Internet chain-letter data. Proceedings of the National Academy of Sciences 105(12), 4633–4638 (2008)
[14] Kossinets, G., Kleinberg, J., Watts, D.: The structure of information pathways in a social communication network. In: Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 435–443. ACM, New York (2008)
[15] Leskovec, J., McGlohon, M., Faloutsos, C., et al.: Patterns of Cascading Behavior in Large Blog Graphs. In: Proceedings of the Seventh SIAM International Conference on Data Mining, pp. 551–556. SIAM, Philadelphia (2007)
[16] Song, X., Chi, Y., Hino, K., et al.: Information flow modeling based on diffusion rate for prediction and ranking. In: Proceedings of the Sixteenth International Conference on World Wide Web, pp. 191–200. ACM, New York (2007)
[17] Chakrabarti, D., Wang, Y., Wang, C., et al.: Epidemic thresholds in real networks. ACM Trans.Inf. Syst. Secur. 10(4), 1–26 (2008)
[18] Goetz, M., Leskovec, J., Mcglohon, M., et al.: Modeling blog dynamics. In: Proceedings of the Third AAAI International Conference on Weblogs and Social Media, pp. 26–33. AAAI, Menlo Park (2009)
[19] Parshani, R., Carmi, S., Havlin, S.: Epidemic Threshold for the Susceptible-Infectious Susceptible Model on Random Networks. Phys. Rev. Lett. 104(25), 258701 (2010)

Compositionality of Team Mental Models in Relation to Sharedness and Team Performance*

Catholijn M. Jonker[1], M. Birna van Riemsdijk[1],
Iris C. van de Kieft[1], and Maria Gini[2]

[1] Delft University of Technology, Man-Machine Interaction Group, Faculty of EEEMCS, Mekelweg 4, 2628 CD, Delft, The Netherlands
[2] University of Minnesota, Department of CS&E, 4-192 Keller Hall, 200 Union St SE, Minneapolis, MN 55455, USA
{c.m.jonker,m.b.vanriemsdijk,i.c.vandekieft}@tudelft.nl, gini@cs.umn.edu

Abstract. "The better the team mental model, the better the teamwork", or so is said, in which better model refers to the extent to which the model is shared by the team members. This paper argues that according to circumstances, some components of that model are more relevant with respect to team performance than others. Circumstances change with the dynamics of the environment, the team composition and organization, its members, and the team task. Consequently, a compositional approach to measuring sharedness of team mental models is proposed. A case study illustrates the argument and the approach.

Keywords: Shared mental model, team model, measuring.

1 Introduction

A Shared Team Mental Model (STMM) is considered beneficial for teamwork, see e.g., (Cannon-Bowers et al., 1993; Lim and Klein, 2006; Mathieu et al., 2000). Aspects to be taken into consideration are equipment, task, team interaction, and team members, see e.g., (Cannon-Bowers et al., 1993; Mathieu et al., 2000). Sharedness refers to the team members having equivalent mental models. For example, in American Football if the team does not share the offensive plan of the quarterback, chances are high that the other team will intercept the ball. The better the shared mental model the better a team is capable of performing its tasks, even in unforeseen circumstances, such as a ball that is not perfectly caught.

Simplistic reasoning would suggest that to improve team performance, the team members should share full knowledge of the state of the world and of what each team member is doing. In reality the communication overhead alone that this would require makes this totally impractical. Furthermore, for teams in which team members have highly individual expertise having full information on what another team member is

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 242–251, 2012.

doing, would not help team performance; e.g., in incident management a fire fighter does not need to know the specifics of the injuries of a person being attended to by the ambulance personnel. This reasoning induces the *hypothesis* that the extent of sharedness of some aspects or components of team mental models is more important for team performance than that of other components, and in particular, that this extent changes with circumstances.

Testing such a hypothesis requires ways of measuring the extent to which teams have a shared mental model. The formalization of the notions of mental model, sharedness, and team mental model as presented in (Jonker et al., 2010), form a starting point. However, the formalizations do not take compositional models into account, nor do they account for the relative importance of each component.

This paper proposes a compositional approach to construct models that underlie teamwork, in which the following components are identified as potentially relevant to shared mental models for teamwork: domain model, competence/capabilities model, and organizational model. Each of these can, in turn, be composed of smaller models as explained in this paper.

A relevance relation is associated with the composition relation, to indicate to what extent the components contribute to team performance. This relevance relation depends on the current circumstances such as the particular instance of the team task, the performance criteria, the dynamics of the environment (world, equipment, and non-team member actors), the composition and organization of the team, and the team members themselves. The relevance relation is used to formulate a compositional way of measuring sharedness of mental models.

Experiments are done within the context of a case study in the Blocks World For Teams (BW4T) domain, as introduced in (Johnson et al., 2009), to test our hypothesis, and to test the adequacy of our compositional approach. In particular we construct different teams of agents that have to operate in two different scenarios, and measure team performance by the time taken by each team for task completion and communication overhead.

This paper is organized as follows. After a brief introduction to shared mental models, we focus on the components of team mental models and their compositional structure. This is followed by an introduction to the testbed (BW4T) which we use to measure the performance of different types of agents that share different types of team mental models. We show how sharing affects performance.

2 Team Mental Models

Team models should at least distinguish equipment, task, team interaction, and team members (Cannon-Bowers et al., 1993; Mathieu et al., 2000). This section adds some components, discusses compositional structures of team mental models, and defines sharedness of compositional team mental models in such a way that the relevance of the components for team performance is taken into account.

2.1 Components of Team Mental Models

Equipment, task, team interaction, and team members should be part of the team mental model according to (Cannon-Bowers et al., 1993; Mathieu et al., 2000). This paper proposes a restructuring of the team mental model and the addition of information and knowledge about the domain (world and agents), and of the underlying ontologies for the whole team mental model as elements of the team mental model. The usual components that refer to task, team interaction, and team members will be part of the organizational specification.

Ontologies: referring to the domain ontology and ontologies for all parts of the domain model as described below.

World State Model: the relevant aspects of the world state should be known to the team members, and thus be part of the team mental model. This part of the model may also contain knowledge about, e.g., the status of equipment.

Agent Models: what are the relevant agents in the domain, and what are their capabilities. Furthermore, a related concept, but still different: the model should describe to what extent agents are competent with respect to their capabilities. For example, there is a difference between the capability of retrieving a block and the competence in retrieving, e.g., one agent can retrieve much faster than another agent. The next sections show that sharing information about the intentions of team members (and even non-team member agents) can contribute clearly to team performance. The same holds for information about their personalities, preferences, and habits.

Organizational Specification: contains the models about task, team members, and team interaction. We refer to existing methods to specify organizations, such as MOISE (Hübner et al., 2002) and OperA (Dignum, V., 2003). Regarding team members, the model should describe which agents are team members, and what role the agents play in the team. The role specifications will refer to the role behavior and the relevant capabilities and competences for that role. To describe capabilities and competences use is made of the ontologies for the domain related to tasks and world state elements. The team task is also part of the organizational specification: it specifies when a task is finished successfully. The team interaction refers to the expected and allowed communications and the timing thereof. Team interaction further refers to e.g., rules of engagement, doctrine, norms, and role interaction.

2.2 Compositional Structures and Relevance

Not in all situations all possible components of team mental models are relevant for team performance. For example, the agents described in the next section do not have the capability to reason about their own capabilities or those of other agents. As a result requiring the agents to have a shared mental model about these competences will not improve performance.

Different hierarchical composition relations can be constructed of the components of the team mental model. An example of a basic hierarchical structure is depicted in Figure 1. To have a fully shared team mental model, each team member has to have this model within his own mind. As part of this team mental model, each team member maintains an agent model for each of his team mates. For example, the team mental model that agent Arnie maintains, contains an agent model of his team mate Bernie. That agent model that Arnie has of Bernie describes not only the beliefs that Arnie has about the availability, capability, competences, and so on of Bernie, it also describes what Arnie thinks that Bernie believes about the team, in other words, it contains Arnie's idea of Bernie's mental model of the team. In Figure 1, for reasons of simplicity the organizational specification component is not decomposed further, and neither is the recursive "Other agent's team mental model".

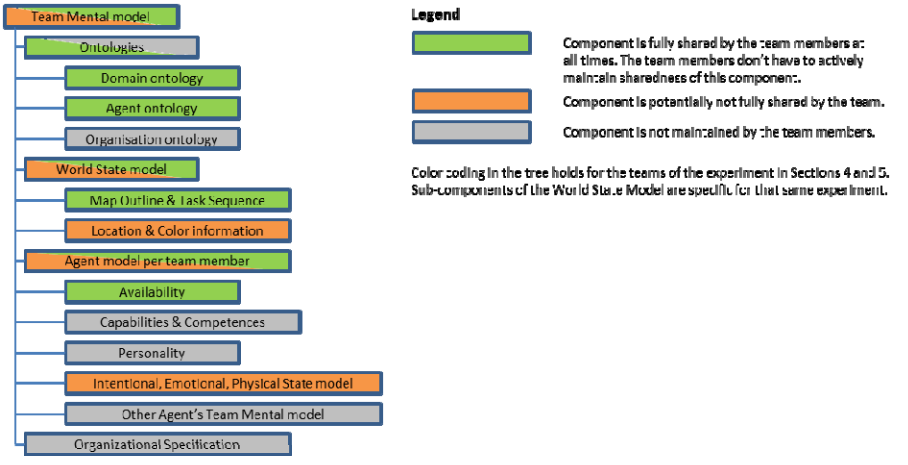


Fig. 1. Hierarchical composition of a team mental model

As mentioned, not all possible components of a team mental model are important in every situation. For example, a team consisting of agents incapable of reasoning about the competencies of other team members does not need to maintain that information. Such components are colored grey in Figure 1. Having sharedness of any of these components would not make the team perform better. Also if all team members are guaranteed to have the same mental model for a component (the green components in Figure 1), then it does not have to be considered as a factor affecting the performance of the teams. Therefore, when measuring the sharedness of a team mental model, only those components should be taken into account that are relevant (colored orange in Figure 1). The color coding is an indication of the relevance relation on components for measuring sharedness of team mental models in such a way that the measure would be predictive of team performance.

3 The Experimental Set Up

An experiment was set up in order to test the hypothesis derived in the introduction:

Hypothesis. The extent to which sharedness of some components of team mental models is more important for team performance than that of other components is situation dependent.

In the following we describe the experimental set up we used to test our hypothesis. In these experiments, aspects of team performance considered are time to finish the task, and communication overhead. The team mental model is kept small and consists only of the intentions of team members and of the world state model. Changes that happen over time while the team is at work affect team performance, see e.g, (Jehn and Mannix, 2001). Thus team performance is monitored over time. Furthermore, we built teams of agents for which the structure of the team mental models is the same, but for which the sharedness differs. We constructed two scenarios, and show that the extent to which sharedness of components contributes to team performance differs per scenario.

The experimental environment is the BW4T domain (Johnson et al., 2009). BW4T is an extension of the classic blocks world that is used to research joint activity of teams. A team of agents has to deliver colored blocks from a number of rooms to the so-called drop zone in a certain color sequence, see Figure 2 (right). The agents can communicate with each other but the cannot see other agents. Furthermore, blocks can only be seen by agents that occupy the room that the blocks are in. If an agent drops a block in a hall way, then this block is lost to all agents. Finally, the world restricts access to rooms to one agent at a time. In the version of the BW4T used in this paper, communication and sensing can be assumed to be reliable, and the world is only changed by the actions of the agents.

The two scenarios differ in the color sequence that has to be deliverd: one with two color repetitions (called medium), and another one with a target sequence of six unique colors (called high). Both scenarios are executed in the same environment consisting of 3 rows of 3 rooms each (A1, A2, A3, B1, B2, B3, C1, C2, C3) connected by corridors. In each scenario, 22 blocks (box_1, … box_22) of various colors were placed in the different rooms. In the first scenario, the task was to bring blocks to the drop zone according to the following sequence: [red, white, white, yellow, darkBlue, darkBlue]. The second task sequence was [red, white, yellow, darkBlue, orange, lightGreen].

The agent teams that have comparable structures in their team mental models but different sharedness of these models were programmed using the GOAL agent programming language (Hindriks, 2009). The agent programming to solve the BW4T tasks was based on the decision cycle presented in Figure 2 (left). Note that we need not produce the most efficient agent for this problem! One template agent was modeled, that, in principle, can solve the task alone. The agents spawned from this template are telling the truth and believe everything they are told. Agent capabilities were left out of consideration in the modeling of the agent, and out of the team models. Four homogeneous teams were formed on the basis of the template agent.

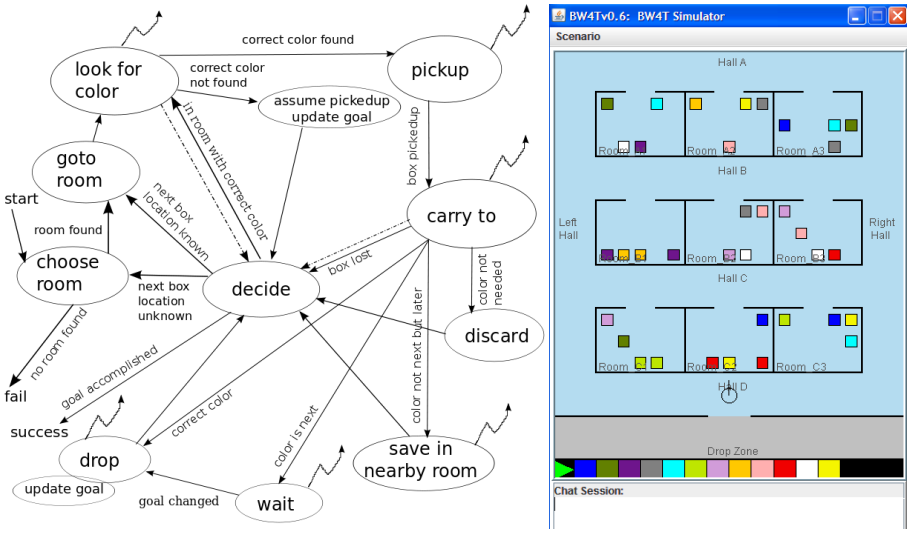


Fig. 2. Agent Decision Cycle (left). Snapshot of the BW4T environment (right).

Team A consisted of agents that do not communicate at all. The agents in Team B communicate only world information to each other. Team C agents only communicate their intentions to each other. Finally, the agents in team D communicate world information and intentions to each other. Each team consisted of three agents, called Bot0, Bot1, and Bot2. This fact was known to all agents, in all teams, in all scenarios. To give an idea of the GOAL code of the agents, we include some examples of action rules for processing world information. The first rule processes the percept of a block of a certain color, by inserting this information into the agent's belief base and communicating it to the other agents. The second rule processes the corresponding message by inserting the content into the belief base. The third rule adopts a goal of holding a block and sends the information about this intention to the other agents. The last rule processes messages concerning the intention of another agent to deliver a block.

```
% If Block of Color is perceived, insert in belief base and send to others
forall bel(percept(color(Block,Color)))  do
  insert(color(Block,Color)) + send(allother, :color(Block,Color)) .

% received messages are inserted in belief base and message deleted
forall bel(received(Sender, color(Block,Color))) do
  insert(color(Block,Color)) + delete(received(Sender, color(Block,Color))).

% macro expressing which block has the color to be delivered next in sequence
#define nextColorLocKnown(Block)
  bel(nextColorToDeliver(Color), color(Block, Color), at(Block,Location)).
```

```
% If I know where a block of the next color to be delivered is, and I don't already have the goal of holding a
% block and I'm not already holding a block, then adopt the goal of holding the block and
% send this intention to others
If nextColorLocKnown(Block), not(a-goal(holding(Block2))),  bel(color(Block, Color),
        not(in('Drop_Zone')), not(holding(_)))
  then adopt(holding(Block)) + send(allother, !deliver(Block, Color)).

% If a message is received that another agent wants to deliver a Block of Color,
% then insert this in belief base and delete the message
forall bel(received(Sender, imp(deliver(Block, Color))))
  do insert(imp(Sender, deliver(Sender, Block, Color)))
       + delete(received(Sender, imp(deliver(Block, Color)))).
```

4 Results of the Pilot Experiment

In the following, we present the results of a pilot experiment with the set up described in the previous section. The results focus on team performance in terms of time to complete the task and communication overhead.

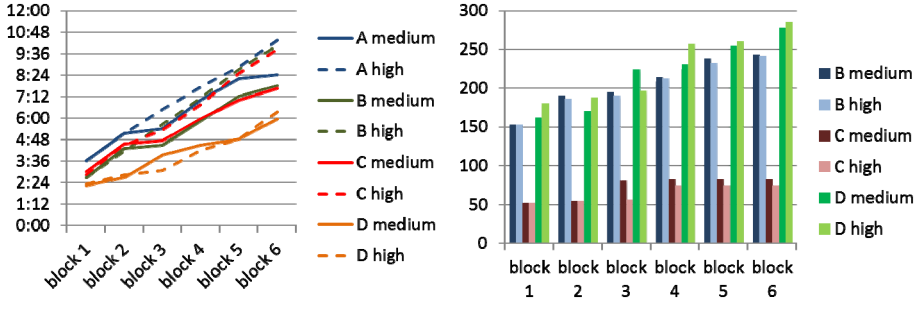


Fig. 3. Team Performance: Time (left) and Communication overhead (right)

The time to complete the task by the different teams in the two scenarios is depicted in Figure 3 (left). The figure shows for each team and scenario the time at which each block in the goal sequence was delivered. The right side of Figure 3 shows the amount of communication required from one block delivery to the next. As more of the sequence is delivered, clearly more information on locations and colors is communicated. Timewise, the performance of team D (full communication) was the best, and there is not much difference between the two scenarios. The results for team C are comparable to those of team B. This means that communicating world information and communicating intentions seem to have a similar effect on the required time. Also the effect of not communicating certain information seems to have a greater effect in the high scenario, i.e., the scenario with different colored blocks. This may be due to the fact that if two agents go for the same color due to lack of communication, this is

more of a problem if there are no double colors in the goal sequence. If there are, it may still be useful to collect blocks of the same color. Also one can see that the difference in performance between team A (no communication) and teams B and C (either world information or intentions are communicated) is not very large. Only in case of full communication we can see a clear improvement in the amount of time required to finish the task.

Our hypothesis is supported both by the findings of our pilot experiment and by practical experience. When considering team performance, team members do not need to have a fully shared team mental model consisting of all possible elements that literature discusses. In fact, practical evidence shows that overhead required to maintain full sharedness of all aspects that might belong to team mental models is counterproductive due to the communication overhead and information overload (Eppler and Mengis, 2004). This suggests that the selection of the components for which the team members should aim for sharedness depends on the circumstances, i.e., task, environment, team composition and organization, and team members.

It is important to realize that knowing that some component is, given the circumstances, probably relevant for team performance, is not the same as knowing to which extent such a component is relevant. Figure 3 shows that teams B and C outperform team A in the medium scenario, but that B and C perform equally well. The difference between these three teams is in the communication about the orange model components in Figure 1, in particular in team B about the locations of blocks and agents, and the color of blocks, and in team C about the intentions of agents. This suggests that the component that team B communicates about is approximately equally important as the component that team C communicates about. However, in the high scenario, neither team B, nor team C performs better than team A. If the results on team D are not taken into account, two possible combinations of these components can be made: AND, but also OR. If team D would have performed equally well as teams B and C, then, for the circumstances in our experimental set up, teams should communicate on intentions only (see Figure 3, right). Now that team D outperforms teams B and C, for high performance on time, teams should share information on BOTH locations, colors, and intentions. If the communication on locations and colors could not be reduced further, then the weight of the component on intentions should be increased, until a balance is found with respect to the weight of communication overhead in the overall team performance.

5 Measuring Sharedness of Compositional Models

This section presents a formal definition of measuring sharedness of compositional models with respect to a relevance relation in terms of weights. It builds on the measure of sharedness, subject overlap, and model agreement as defined in (Jonker et al., 2010). These notions are all defined relative to a set of questions Q, which can be seen as those questions that determine the purpose of the model. As team members do not have to be clones of each other with respect to all the information and knowledge in their minds, the set of questions Q is to be used to focus the team mental model on those subjects that matter. Therefore, the subject overlap of the team mental models in

the minds of the team members should be complete (denoted by 1 in the definition below). Furthermore, agreement means that these models provide the same answers (at least to extent θ), and thus provide sharedness to at least that extent. The definition of sharedness provided by (Jonker et al., 2010) is then used to determine the sharedness of models that are not composed of submodels.

Definition 1. Shared Mental Model (Jonker et al., 2010). A model M is a mental model that is shared to the extent θ by a set of agents *A* with respect to a set of questions Q, denoted by Sh(M, *A*, θ) iff there is a mental model M_A for each agent $A \in A$ with respect to Q, such that

1. $SO(M,M_A,Q) = 1$,
2. $Agr(M,M_A,Q) \geq \theta$,

We can define the sharedness of the compositional model in terms of a function that takes the sharedness per component to the extent that it is relevant, and follows that same composition relation as the team mental model itself. A compositional linear additive function suffices for the case study, using weights to model the relevance of each component. Following our hypothesis, we expect that then also the answer to the question which compositional function for measuring sharedness is indicative of team performance, should be situation dependent.

Definition 2. Given a set of agents *A*, a model M, and a set of questions Q, then

$U_Q(A, M) = \max\{ \theta \mid Sh(M, A, \theta) \}$, if M is not composed.

$U_Q(A, M) = \sum_{m \in M} w_m U_Q(A, m)$, if M is composed.

Where w_m is the weight of component m, and $\sum_{m \in M} w_m = 1$.

6 Conclusions and Future Work

This paper shows that according to circumstances some of the components of team mental models are more relevant for team performance than others. Circumstances change with the dynamics of the environment (i.e., world, equipment, non-member actors), the team composition and organization, its members, and the team task.

Results of a pilot experiment in which different teams operate in the domain of the BW4T support our compositional approach of team mental models and enable measuring their sharedness. By comparing the team performance of the teams over two different scenarios it becomes clear that of the two components under consideration should be combined differently in the measure of sharedness over the two scenarios.

Finally, the paper proposes a compositional approach to measuring sharedness of team mental models, and suggests a linear additive function for the case study. A formal definition is provided that shows how such functions can be used to compute sharedness of composition models. In combination with the more qualitative analysis approach that can be associated with the color coding presented in Figure 1, such functions make this work generally applicable.

Future work will focus on running full experiments, determining precise sets of questions Q to underpin the formalizations presented in Section 5, and computing

sharedness of the team mental models in various ways to test the practical value of such computations. Furthermore, additional performance measures will be taken into account and more sophisticated agents will be used, e.g., capable of reasoning about their team mates and including errors in sensing and actions.

Acknowledgement. The authors thank the anonymous reviewers for their comments. This work is subsidized by the Dutch Technology Foundation STW, applied science division of NWO and the Technology Program of the Ministry of Economic Affairs; the Pocket Negotiator project with grant number VICI-project 08075.

References

1. Cannon-Bowers, J.A., Salas, E., Converse, S.: Shared mental models in expert team decision making. In: Castellan, N.J. (ed.) Individual and Group Decision Making, pp. 221–245. Lawrence Erlbaum Associates (1993)
2. Dignum, V.: A model for organizational interaction: based on agents, founded in logic. Dissertation, Universiteit Utrecht (2003)
3. Eppler, M.J., Mengis, J.: The Concept of Information Overload: A Review of Literature from Organization Science, Accounting, Marketing, MIS, and Related Disciplines. The Information Society 20(5), 325–344 (2004)
4. Hindriks, K.V.: Programming rational agents in GOAL. In: Bordini, R.H., Dastani, M., Dix, J., El Fallah Seghrouchni, A. (eds.) Multi-Agent Programming: Languages, Tools and Applications, pp. 119–157. Springer, Berlin (2009)
5. Hübner, J.F., Sichman, J.S., Boissier, O.: A Model for the Structural, Functional, and Deontic Specification of Organizations in Multiagent Systems. In: Bittencourt, G., Ramalho, G.L. (eds.) SBIA 2002. LNCS (LNAI), vol. 2507, pp. 118–128. Springer, Heidelberg (2002)
6. Jehn, K.A., Mannix, E.A.: The dynamic nature of conflict: a longitudinal study of intragroup conflict and group performance. The Academy of Management Journal 44(2), 238–251 (2001)
7. Johnson, M., Jonker, C., van Riemsdijk, B., Feltovich, P.J., Bradshaw, J.M.: Joint Activity Testbed: Blocks World for Teams (BW4T). In: Aldewereld, H., Dignum, V., Picard, G. (eds.) ESAW 2009. LNCS (LNAI), vol. 5881, pp. 254–256. Springer, Heidelberg (2009)
8. Jonker, C.M., van Riemsdijk, M.B., Vermeulen, B.: Shared Mental Models: A Conceptual Analysis. In: De Vos, M., Fornara, N., Pitt, J.V., Vouros, G. (eds.) COIN 2010. LNCS, vol. 6541, pp. 132–151. Springer, Heidelberg (2011)
9. Lim, B., Klein, K.: Team mental models and team performance: A field study of the effects of team mental model similarity and accuracy. Journal of Organizational Behavior 27(4), 403 (2006)
10. Mathieu, E., Heffner, T.S., Goodwin, G., Salas, E., Cannon-Bowers, J.: The influence of shared mental models on team process and performance. Journal of Applied Psychology 85(2), 273–283 (2000)

Feature Selection for Improved Phishing Detection

Ram B. Basnet[1], Andrew H. Sung[1], and Quingzhong Liu[2]

[1] Computer Science and Engineering & ICASA, New Mexico Tech, Socorro, NM, USA
{rbasnet,sung}@cs.nmt.edu
[2] Computer Science, Sam Houston State University, Huntsville, TX, USA
qxl005@shsu.edu

Abstract. Phishing – a hotbed of multibillion dollar underground economy – has become an important cybersecurity problem. The centralized blacklist approach used by most web browsers usually fails to detect zero-day attacks, leaving the ordinary users vulnerable to new phishing schemes; therefore, learning machine based approaches have been implemented for phishing detection. Many existing techniques in phishing website detection seem to include as many features as can be conceived, while identifying a relevant and representative subset of features to construct an accurate classifier remains an interesting issue in this particular application of machine learning. This paper evaluates correlation-based and wrapper feature selection techniques using real-world phishing data sets with 177 initial features. Experiments results show that applying an effective feature selection procedure generally results in statistically significant improvements in the classification accuracies of – among others – Naïve Bayes, Logistic Regression and Random Forests, in addition to improved efficiency in training time.

Keywords: feature selection, phishing detection, phishing webpage, evolutionary algorithms, anti-phishing.

1 Introduction

Phishing has become something of a plague on the Internet. A typical phishing webpage may mimic a trusted third party such as a bank, a financial or e-commerce entity, etc. and induces Internet users to divulge their private information, e.g., username, password, bank account, credit card number, etc. Phishing attacks can cost not only the individual consumers but also well-known organizations and corporations whose brands are compromised in the attacks. Despite the efforts by the research community, the industry, and law enforcement to develop solutions to tackle the problem, phishing has shown no sign of abating. A recent report by the Anti-phishing Working Group (APWG) [1] indicated more sophisticated schemes seem to have been used in phishing attacks that also exploited an increased number of brands.

Since a black-list of phishing sites is unable to detect "zero-day" or new attacks [27], a machine-learning approach has been proposed to train a classifier with large amount of data. The classifiers reported in the literature [6, 9, 21], however, seem to

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 252–261, 2012.

include very large numbers of features. Since each feature included can increase the cost (storage, preprocessing, training time, etc.) of a system without possibly contributing to the classifier's performance, there is a strong motivation to design and implement systems with small feature sets as, according to M. Hall, "a good feature subset is one that contains features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other" [2].

In this paper, we evaluate two common feature selection techniques – correlation-based and wrapper-based techniques – for phishing detection. We compare these feature selection techniques using two feature space searching techniques (genetic and greedy forward selection) and conduct the experiments and evaluate results on a real-world dataset with more than 16,000 phishing webpages and more than 32,000 non-phishing webpages.

2 Related Work

A number of recent papers have evaluated various machine learning techniques in detecting phishing emails, URLs, and webpages [5, 6, 8, 9, 12, 21, 25, etc.]. Most research works, however, use all the features that can be conceived at the time and as a result feature selection study in phishing detection can be found sparingly.

In [9] Whittaker et al. describe the design and performance characteristics of Google's phishing blacklist. Their proprietary classifier, implementation of the online gradient descent logistic regression learning algorithm, performs the automatic feature selection—finding potential useful features to include in classification model and discarding the ones that do not contribute to the model.

Toolan et al. apply feature selection techniques to phishing and spam email classification using 40 features [17].

3 Feature Selection Methods

We evaluate two commonly used feature selection techniques in this paper.

3.1 Correlation-Based Feature Selection (CFS)

CFS exploits the inter-dependency or predictability of one variable with another to generate the optimal subset of features with the goals of improving classification performance and reducing the feature dimension. As a simple filter algorithm that evaluates an importance of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them, CFS essentially ranks feature subsets in the search space of all possible feature subsets according to a correlation based heuristic evaluation function:

$$M_s = \frac{k\overline{\mathrm{r_{cf}}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}, \tag{1}$$

where M_s is the heuristic "merit" of feature subset S containing k features, $\overline{r_{cf}}$ is the mean feature-class correlation ($f \in S$), and $\overline{r_{ff}}$ is the average feature-feature

inter-correlation [2]. The numerator of (1) provides an indication of how predictive of the class a set of features are and the denominator provides how much redundancy there is among the features. Fig. 1 is the graphical representation of the correlation-based feature technique.

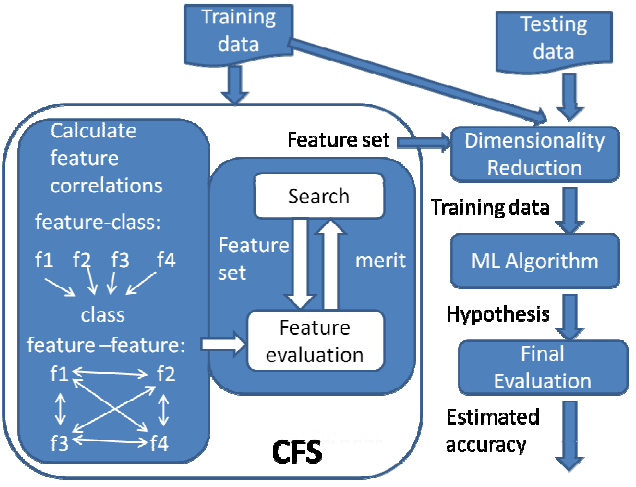


Fig. 1. Graphical representation of correlation-based feature selection (CFS)

3.2 Wrapper Feature Selection (WFS)

Wrapper feature selector evaluates feature subsets by using a machine learning algorithm with the rationale that the induction method that will ultimately use the feature subset should provide a better estimate of accuracy. Though using the induction algorithm itself as the measure stands the best chance of identifying the "optimal" feature subset, wrapper feature selectors give highly variable cross-validation accuracy when the number of instances is small [3], and are prohibitively slow on large data sets using cross-validation [7]. Fig. 2 is the graphical representation of the wrapper feature selection technique.

3.3 Search Techniques

Searching the space of feature subsets within reasonable time constraints is vital in any feature selection technique. There are several search heuristics such as forward selection, backward elimination, best first, search using genetic algorithms, etc. Forward search and backward elimination are common and simple techniques where the algorithms consider only additions or deletions respectively to the feature subset [22], [23]. We evaluate greedy forward search and genetic algorithm in this study.

Genetic algorithms (GA) are adaptive search techniques based on the principles of natural selection and mutation in biology [26]. GA typically maintains a constant sized

population of individuals which represent samples of the space to be searched. Each individual is evaluated on the basis of its overall fitness – how good a feature subset is with respect to an evaluation strategy. The solution space is searched in parallel which helps in avoiding local optima. The algorithm is an iterative process where new individuals (offspring) for the next generation are formed by using two main genetic operators such as crossover and mutation to the members of the current generation. Mutation randomly changes (thus adding or deleting features) one or more components of selected individuals. Crossover combines different from a pair of subsets into a new subset. Better feature subsets have a greater chance of being selected to form a new subset through crossover or mutation, effectively evolving good subsets over time [2], [4], [15].

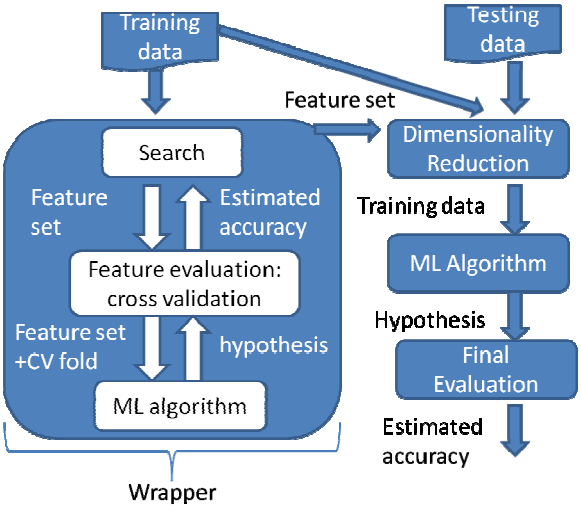


Fig. 2. Graphical representation of wrapper feature selection

4 Experiments and Results

We used 10 times 10-fold cross-validation (unless otherwise stated) to estimate the test accuracy. The experiments were run on a machine with 2 dual-core 2 GHz Intel processors with 4 GB memory. To conduct all the experiments, we used WEKA (Waikato Environment for Knowledge Analysis) data mining framework [14] with default parameter values where appropriate.

We compare feature selection and search techniques using 3 commonly used machine learning algorithms – Naïve Bayes (NB) [17], Logistic Regression (LR) [28], and Random Forests (RF) [16] – in problems similar to ours. We also tried evaluating C4.5 [24] and Multilayer Perceptron, but the Wrapper feature selection technique was prohibitively slower taking months for these slower classifiers.

4.1 Evaluation Criteria

As we formulate the phishing webpage detection problem as binary classification problem, each webpage falls into one of four possible scenarios: true positive (TP, correctly classified phishing webpage), true negative (TN, correctly classified non-phishing webpage), false positive (FP, non-phishing webpage wrongly classified as phishing), and false negative (FN, phishing webpage wrongly classified as non-phishing). Though error rate (fraction of wrongly classified URLs) may be of limited interest in our context where data sets are unbalanced (see next section), we report it anyway to make it easier to compare our results with that from the existing literature. Additionally, we report standard measures such as false positive rate (FPR) and false negative rate (FNR).

4.2 Data Sets

For phishing webpages, we used confirmed phishing URLs from PhishTank [11]. PhishTank, operated by OpenDNS, is a collaborative clearing house for data and information about phishing on the Internet. A phish once submitted is verified by a number of registered users to confirm it as phishing. We collected first set of phishing URLs from June 1 to October 31, 2010. Phishing tactics used by scammers evolve over time. In order to investigate these evolving tactics and to closely mimic our experiments as in the real-world scenario, we collected second batch of confirmed phishing URLs that were submitted for verification from January 1 to May 3, 2011. We used scripts [13] to automatically detect and expand the shortened URLs provided by online service longurl.org.

We collected our legitimate webpages from two public data sources. One is the Yahoo! directory[1], the web links in which are randomly provided by Yahoo's server redirection service [10]. We used this service to randomly select a URL and download its page contents along with server header information. In order to cover wider URL structures and varieties in page contents, we also made a list of URLs of most commonly phished targets. We then downloaded those URLs, parsed the retrieved HTML pages, and harvested and crawled the hyperlinks therein to also use as benign webpages. We made the assumption, which we think is reasonable, to treat those webpages as benign, since their URLs were extracted from a legitimate sources. These webpages were crawled between September 15 and October 31 of 2010. The other source of legitimate webpages is the DMOZ Open Directory Project[2]. DMOZ is a directory whose entries are vetted manually by editors.

Based on the date on which phishing URLs were submitted to PhishTank for verification, we generated two data sets. The first data set, we refer to it as DS1, contains 11,240 phishing webpages submitted before October 31, 2010 and 21,946 legitimate webpages from Yahoo! and seed URLs. The second data set, we refer to it as DS2, contains 5,454 phishing webpages submitted for verification between January

[1] `http://dir.yahoo.com`
[2] `http:www.dmoz.org`

1 and May 3 of 2011 and 9,635 randomly selected legitimate webpages from DMOZ. We discarded the URLs that were no longer valid as the page couldn't be accessed to extract features from their contents.

4.3 Features

We start with a set of 177 features of which 38 are content-based and the rest are URL-based. Content-based features are mostly derived from the technical (HTML) contents of webpages e.g., counting external and internal links, counting IFRAME tags, and checking whether IFRAME tag's source URLs are present in blacklists and search engines, checking for password field and testing how the form data is transmitted to the servers (whether Transport Layer Security is used and whether "GET" or "POST" method is used to transmit form data with password field), etc.

URL-based features include lexical properties of URLs such as counting number of ".", "-", "_", etc. in various parts of URLs, checking whether IP address is used and what type of notation is used to represent the IP address in place of a domain name. URLs and domain part of the URLs are checked against top 3 search engines (Google, Yahoo, and Bing) indexes to see if the URLs are indexed. Features also include checking IPs and domain name of the URLs against the top list of IPs and domains historically popular for hosting phishing and other malicious websites. Features also include a list of eye-catching keywords (e.g., log, click, pay, free, bonus, bank, user, etc.) that are more commonly used in phishing URLs to deceive the end users.

4.4 Feature Selection

Table 1 displays the classification accuracies of Naïve Bayes, Logistic Regression and Random Forests classifiers with and without feature selection using CFS on DS1 data set. Genetic search technique resulted in a subset of 42 features out of 177 features; whereas greedy forward search (Greedy FS) selected all the features (results are not shown as they are same as without feature selection, grayed row). Genetic search technique improved Naïve Bayes classifier's results the most with its error improving from 2.2% to 1.7% with the significant reduction in both FPR and FNR.

Table 1. Classification results of Naïve Bayes, Logistic Regression and Random Forests classifiers using correlation-based feature selection method with genetic search and greedy forward selection search techniques on DS1 data set

Search Technique	# Features	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection	177	2.2	2.2	2.2	0.5	0.3	0.8	0.4	0.3	0.7
Genetic Search	42	1.7-	1.6-	1.9-	0.9+	0.9+	1.0+	0.5+	0.4+	0.6-

+,- statistically significant degradation or improvement

Table 2 shows the classifiers' results using Wrapper feature selection technique. Classification accuracies on Naïve Bayes, Logistic Regression and Random Forests are compared using two search techniques, genetic search and greedy forward selection. Unlike CFS, Wrapper based technique selected smaller subsets of features for all three classifiers using both genetic search and greedy forward search techniques. Though Wrapper based technique was notably slower compared to CFS technique, it aided in significant improvement in the classification accuracies of all classifiers. For example, with the subset of 14 selected features using greedy forward search technique, RF yielded the best error rate of 0.3% along with the best FPR and FNR of 0.2% and 0.5%, respectively on DS1 data set. Besides yielding higher accuracy, the reduced feature subset noticeably improved the training time.

Table 2. Classification results of Naïve Bayes, Logistic Regression and Random Forests classifiers using wrapper feature selection method with genetic search and greedy forward selection search techniques on DS1 data set

Search Technique	Classifier Performance											
	Naïve Bayes				Logistic Regression				Random Forests			
	# Features	Error (%)	FPR (%)	FNR (%)	# Features	Error (%)	FPR (%)	FNR (%)	# Features	Error (%)	FPR (%)	FNR (%)
Without feature selection	177	2.2	2.2	2.2	177	0.5	0.3	0.8	177	0.4	0.3	0.7
Genetic Search	62	1.3-	0.9-	2.1-	70	0.3-	0.2-	0.6-	91	0.4	0.2-	0.6-
Greedy FS	12	1.5-	1.0-	1.6-	13	0.4-	0.2-	0.8	14	0.3-	0.2-	0.5-

+,- statistically significant degradation or improvement

4.5 Effect of Feature Selection on Data Drift

Phishers come up with new tactics over time to invade the existing filters. Features developed and selected from observing a particular data set can yield highly accurate classification results when trained and tested on disjoint subsets of the same data set. But do these results hold on testing new data (possibly from different sources) using the features extracted and selected from older data set? We try to investigate this question in the following experiments.

First, using the selected features from DS1 data set, we ran our experiments on DS2 data set and show the results in Table 3. With CFS and genetic search combination, we see slightly better results for NB, but no improvement in overall error rates for LR and RF classifiers compared to the results using all the features.

Wrapper feature selection method with both genetic and greedy forward search techniques, on the other hand, degraded classification accuracy of most of the classifiers. A subset of 42 features selected from DS1 data set using CFS and genetic search combination yielded a small improvement in error rate for Naïve Bayes on DS2

data set. The same feature subset, however, didn't improve the error rates of Logistic Regression and Random Forests. The combination of wrapper feature selection and greedy forward search technique improved classification accuracy of LR but decreased accuracies for NB and RF classifiers.

Table 4 shows the experimental results on testing newer data set DS2 using the models generated from training older data set DS1. As expected, the classification accuracy degraded significantly for all the classifiers. Interestingly, Naïve Bayes' performance results degraded the least while Random Forests' performance degraded the worst with or without performing feature selection in this context. Results show that the complete features are better than selected smaller subsets when it comes to classifiers' robustness towards concept drift in this context. Results suggest that as phishing tactics change over time, so must the data models in order to keep the models fresh and achieve optimal performance results.

Table 3. Results of using selected features from DS1 data set on DS2 data set

Feature Selection	Search Technique	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection		0.8	0.2	1.9	0.4	0.3	0.7	0.3	0.0	0.7
CFS	Genetic Search	0.7-	0.2	1.5-	0.4	0.1-	0.9+	0.3	0.1+	0.6-
Wrapper	Genetic Search	1.6+	1.0+	2.7+	0.5+	0.4+	0.6-	0.3	0.1+	0.6-
	Greedy FS	2.7+	2.3+	3.4+	0.2-	0.0-	0.6-	0.4+	0.2-	0.8+

+,- statistically significant degradation or improvement

Table 4. Results of training on older data set DS1 and testing the models on newer data set DS2 for the combinations of CFS and wrapper based feature selection techniques with genetic and greedy search techniques

Feature Selection	Search Technique	Classifier Performance								
		Naïve Bayes			Logistic Regression			Random Forests		
		Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)	Error (%)	FPR (%)	FNR (%)
Without feature selection		3.2	0.5	8.0	3.8	0.2	10.2	4.0	0.0	11.1
CFS	Genetic Search	3.6+	0.2-	9.6+	4.8+	0.1-	13.2+	5.2+	0.0	14.2+
Wrapper	Genetic Search	8.5+	0.5	22.7	8.7+	0.1-	24.0+	8.3+	0.0	23.0+
	Greedy FS	5.8+	0.0-	16.0+	7.4+	0.0-	20.6+	16.1+	0.0	44.5+

+,- statistically significant degradation or improvement

5 Conclusions and Future Work

In this paper, we evaluated two common feature selection techniques: correlation based and wrapper based feature selection techniques for phishing website detection. We also evaluated two search methods: genetic search and greedy forward selection. Applying the techniques on real-world data sets, we experimentally demonstrated that feature selection technique can improve classification results when training and testing on the disjoint subsets of a data set.

Though wrapper based feature selection technique was extremely slow (taking several weeks) for slower classifier like C4.5 and Multilayer Perceptron (results not shown) as compared to correlation based feature selection (CFS) technique (taking hours or days), wrapper based technique improved classifiers accuracies significantly compared to CFS technique for the evaluated classifiers. Using all the features, however, yielded better results when training with older data set and testing the generated models with newer data set.

As future work, it would be interesting to evaluate other feature ranking and selection techniques such as principle component analysis, latent semantic analysis, chi-squared attribute evaluation, etc. and other feature space search methods such as greedy backward elimination, best first, etc.

Acknowledgment. This research work was supported by Institute for Complex Additive Systems Analysis (ICASA), a research division of New Mexico Tech and by National Institute of Justice award no. 2010-DN-BX-K223. The opinions and findings expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

References

1. APWG Phishing Activity Trends Report- 2nd Half (2010), http://apwg.org/reports/apwg_report_h2_2010.pdf (accessed on October 21, 2011)
2. Hall, M.A.: Correlation-based Feature Selection for Machine Learning. Hamilton, NewZealand (1999)
3. Kohavi, F., John, G.H.: Wrappers for Feature Subset Selection. Artificial Intelligence 97, 273–324 (1997)
4. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
5. Basnet, R., Mukkamala, S., Sung, A.H.: Detection of Phishing Attacks: A Machine Learning Approach. In: Prasad, B. (ed.) Soft Computing Applications in Industry. STUDFUZZ, vol. 226, pp. 373–383. Springer, Heidelberg (2008)
6. Ma, J., Saul, L.K., Safage, S., Voelker, G.M.: Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs. In: ACM SIGKDD, Paris, France, pp. 1245–1253 (2009)
7. Caruna, R., Freitag, D.: Greedy Attribute Selection. In: 11th International Conference in Machine Learning. Morgan Kaufmann, San Francisco (1994)

8. Zhang, Y., Hong, J., Cranor, L.: CANTINA: A Content-Based Approach to Detecting Phishing Web Sites. In: WWW 2007, Banff, Alberta, Canada. ACM Press (2007)
9. Whittaker, C., Ryner, B., Nazif, M.: Large-Scale Automatic Classification of Phishing Pages. In: 17th Annual Network and Distributed System Security Symposium, California, USA (2010)
10. Yahoo! Inc.: Random Link – random, http://random.yahoo.com/fast/ryl
11. PhishTank - Out of the Net, into the Tank, http://www.phishtank.com/developer_info.php
12. Garera, S., Provos, N., Chew, M., Rubin, A.D.: A Framework for Detection and Measurement of Phishing Attacks. In: 5th ACM Workshop on Recurring Malcode (WORM 2007), pp. 1–8. ACM Press, New York (2007)
13. PyLongURL - Python Library for LongURL.org, http://code.google.com/p/pylongurl/
14. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. ACM SIGKDD Explorations 11, 1–8 (2009)
15. Vafaie, H., Jong, K.D.: Robust Feature Selection Algorithms. In: International Conference on Tools with Artificial Intelligence (ICTAI), pp. 356–363 (1993)
16. Breiman, L.: Random Forests. Machine Learning 45, 5–32 (2001)
17. John, G., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: 11th International Conference on Uncertainty in Artificial Intelligence, San Mateo, USA, pp. 338–345 (1995)
18. Toolan, F., Carthy, J.: Feature Selection for Spam and Phishing Detection: In: eCrime Researchers Summit (eCrime), Dallas, TX, pp. 1–9 (2010)
19. Miyamoto, D., Hazeyama, H., Kadobayashi, Y.: A Proposal of the AdaBoost-Based Detection of Phishing Sites. In: 2nd Joint Workshop on Information Security (2007)
20. Fette, I., Sadeh, N., Tomasic, A.: Learning to Detect Phishing Emails. In: 16th International Conference on World Wide Web, pp. 649–656 (2007)
21. Basnet, R.B., Sung, A.H.: Classifying Phishing Emails Using Confidence-Weighted Linear Classifiers. In: International Conference on Information Security and Artificial Intelligence, Chengdu, China, pp. 108–112 (2010)
22. Kittler, J.: Feature Set Search Algorithms. In: Chen, C.H. (ed.) Pattern Recognition and Signal Processing, The Netherlands (1978)
23. Miller, J.: Subset Selection in Regression. Chapman and Hall, New York (1990)
24. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
25. Basnet, R.B., Sung, A.H., Liu, Q.: Rule-Based Phishing Attack Detection. In: International Conference on Security and Management (SAM 2011), Las Vegas, NV (2011)
26. Holland, J.H.: Adaption in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
27. Ludl, C., McAllister, S., Kirda, E., Kruegel, C.: On the Effectiveness of Techniques to Detect Phishing Sites. In: Hämmerli, B.M., Sommer, R. (eds.) DIMVA 2007. LNCS, vol. 4579, pp. 20–39. Springer, Heidelberg (2007)
28. le Cessie, S., van Houwelingen, J.C.: Ridge Estimators in Logistic Regression. Applied Statistics 41, 191–201 (1992)

Identification of Smartphone-Image Source and Manipulation

Qingzhong Liu[1], Xiaodong Li[1], Lei Chen[1], Hyuk Cho[1], Peter A. Cooper[1], Zhongxue Chen[2], Mengyu Qiao[3], and Andrew H. Sung[4]

[1] Department of Computer Science,
Sam Houston State University, Huntsville, TX 77341, USA
[2] Biostatistics Epidemiology Research Design Core,
Center for Clinical and Translational Sciences,
University of Texas Health Science Center at Houston, Houston TX 77030, USA
[3] Department of Mathematics and Computer Science,
South Dakota School of Mines and Technology, Rapid City, SD 57701, USA
[4] Department of Computer Science and Engineering & Institute for
Complex Additive Systems Analysis, New Mexico Tech, Socorro, NM 87801, USA

Abstract. As smartphones are being widely used in daily lives, the images captured by smartphones become ubiquitous and may be used for legal purposes. Accordingly, the authentication of smartphone images and the identification of post-capture manipulation are of significant interest in digital forensics. In this paper, we propose a method to determine the smartphone camera source of a particular image and operations that may have been performed on that image. We first take images using different smartphones and purposely manipulate the images, including different combinations of double JPEG compression, cropping, and rescaling. Then, we extract the marginal density in low frequency coordinates and neighboring joint density features on intra-block and inter-block as features. Finally, we employ a support vector machine to identify the smartphone source as well as to reveal the operations. Experimental results show that our method is very promising for identifying both smartphone source and manipulations. Our study also indicates that applying unsupervised clustering and supervised classification together (clustering first, followed by classification) leads to improvement in identifying smartphone sources and manipulations and thus provides a means to address the complexity issue of intentional manipulation.

Keywords: Image forensics, smartphone identification, classification, hierarchical clustering, support vector machine, JPEG images, source, operation, neighboring joint density.

1 Introduction

In the analog era, images were routinely used as important evidence in court proceedings. In today's digital age, film has been replaced by digital still and video images. Since digital images can be created and easily modified by digital processing tools, a challenging problem has emerged in getting digital images accepted as valid evidence

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 262–271, 2012.

in courts: how to determine the provenance and authenticity of digital images? As the credibility of an image is an important in the legal context, matching a digital image to a particular camera sensor can be equated to the matching process of an individual bullet to a gun barrel. Comparing bullet scratches allows forensic examiners to associate them with a particular barrel, convincing the court to accept it as the original criminal tool. In an analogous manner, digital forensic research has been focused on locating digital "scratches" within images for the purpose of revealing the source device [1, 2, 4, 5, 6, 7, 8, 9, 16, 18].

With the decreasing cost of mobile phones and increasing quality of megapixel camera phones, mobile phones are increasingly being used to capture pictures in people's daily activities. Source identification based on the different patterns of sensor noise or sensor fingerprint has been shown to be successful. However, once obtained images are reprocessed, e.g., cropping, rescaling (interpolation), and recompressing, it is ineffective in identifying the camera source since the pattern of sensor noise is generally destroyed. Although several methods have been presented to detect a single operation, e.g., cropping, image interpolation, or JPEG-based double compression [6, 11, 14, 15, 17], identifying the camera source based on processed images with the combination of different operations remains problematic.

In this paper, we propose a method that utilizes the marginal density of DCT coefficients in low-frequency coordinates and the neighboring joint density in DCT domain to identify the source and manipulations, based on already processed smartphone images. In addition, hierarchical clustering and a support vector machine are employed to differentiate smartphone sources and processing operations. Experiments indicate that our approach is very promising. The remainder of this paper is organized as follows. We describe feature design in section 2 and conduct experiments in section 3, followed by discussion in section 4, and conclusion in section 5.

2 Feature Mining

2.1 Marginal Density Features

Generally the manipulation of JPEG images will modify the DCT coefficients and hence change the marginal density of DCT coefficients at each specific frequency coordinate. Because most non-zero DCT coefficients are aggregated at low-frequency coordinates and the modification is noticeable at the corresponding low frequency subband, we design the following marginal density features.

An 8×8 DCT block consists of 64 frequency coefficients, with the frequency coordinates from (1, 1) to (8, 8), where the upper-left corresponds to low frequency subband and the right-bottom to high frequency subband. Let F denote the DCT coefficient array of a JPEG image, which consists of $M \times N$ blocks, F_{ij} (i =1, 2, …, M; j = 1, 2, …, N). We select the set of low frequency coordinates as follows:

$$S = \{(2,1), (1, 2), (1, 3), (2, 2), (3,1), (1,4), (2,3), (3,2), (4,1)\}. \quad (1)$$

The feature set consists of the following probability values

$$X = \left\{ \frac{1}{MN} \left(h_{kl}(0), h_{kl}(1), h_{kl}(2), h_{kl}(3), h_{kl}(4) \right) \middle| (k,l) \in S \right\}, \quad (2)$$

where $h_{kl}(m)$ denotes the histogram of the absolute DCT coefficient at frequency coordinate (k,l) with the value m (m=0, 1, 2, 3, 4). Therefore, there are total of 45 features in the marginal density set.

2.2 Neighboring Joint Density Features

Generalized Gaussian distribution (GGD) is widely used in modeling the probability density function (PDF) of a multimedia signal. Although there does not appear to exist a generally agreed multivariate extension of the univariate generalized Gaussian distribution, some researchers define a parametric multivariate generalized Gaussian distribution (MGGD) model. In our previous study, we found that some manipulations such as JPEG double compression and information hiding generally modify the neighboring joint density of DCT [13, 14]. In this study, we surmise that JPEG images obtained from different smartphones and manipulated by different operations will have different properties on the neighboring joint density. Therefore, we design the following neighboring joint density features.

Neighboring Joint Density on Intra-block

Let F demote the compressed DCT coefficient array of a JPEG image, consisting of $M \times N$ blocks, F_{ij} (i =1, 2, …, M; j = 1, 2, …, N). Each block has a size of 8 × 8. The intra-block neighboring joint density matrix on horizontal direction $absNJ_{1h}$ and the matrix on vertical direction $absNJ_{1v}$ are constructed as follows:

$$absNJ_{1h}(x,y)=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{m=1}^{8}\sum_{n=1}^{7}\delta(|c_{ijmn}|=x,|c_{ijm(n+1)}|=y)}{56MN}, \tag{3}$$

$$absNJ_{1v}(x,y)=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{m=1}^{7}\sum_{n=1}^{8}\delta(|c_{ijmn}|=x,|c_{ij(m+1)n}|=y)}{56MN}, \tag{4}$$

where c_{ijmn} is the DCT coefficient located at the m-th row and the n-th column in the block F_{ij}; δ =1 if its arguments are stratified, otherwise δ =0; and x and y are integers. For computational efficacy, $absNJ_1$ is defined as the neighboring joint density features on intra-block and calculated as follows:

$$absNJ_1(x,y)=\frac{absNJ_{1h}(x,y)+absNJ_{1v}(x,y)}{2}, \tag{5}$$

Neighboring Joint Density on Inter-block

The inter-block neighboring joint density matrix on horizontal direction $absNJ_{2h}$ and the matrix on vertical direction $absNJ_{2v}$ are constructed as follows:

$$absNJ_{2h}(x,y)=\frac{\sum_{m=1}^{8}\sum_{n=1}^{8}\sum_{i=1}^{M}\sum_{j=1}^{N-1}\delta(|c_{ijmn}|=x,|c_{i(j+1)mn}|=y)}{64M(N-1)}, \tag{6}$$

$$absNJ_{2v}(x,y)=\frac{\sum_{m=1}^{8}\sum_{n=1}^{8}\sum_{i=1}^{M-1}\sum_{j=1}^{N}\delta(|c_{ijmn}|=x,|c_{(i+1)jmn}|=y)}{64(M-1)N}, \tag{7}$$

Similar to $absNJ_1$, $absNJ_2$ is defined as the neighboring joint density features on inter-block and calculated as follows:

$$absNJ_2(x, y) = \frac{absNJ_{2h}(x, y) + absNJ_{2v}(x, y)}{2}, \quad (8)$$

In our feature design, the values of x and y are in [0, 4] and thus $absNJ_1$ consists of 25 features and $absNJ_2$ also consists of 25 features. In summary, our feature set contains 45 marginal density features and 50 neighboring joint density features.

3 Experiments

3.1 Setup

In our experiments, we captured digital images using five different types of smartphones from four manufacturers. Those images were taken randomly without any particular requirement. The information on these smartphone images are listed in Table 1 and some image samples are shown in Figure 1.

Table 1. Original images obtained by smartphones

Smartphone brand	# Images	Format
HTC G3	149	JPEG
HTC HD2	114	JPEG
Huawei U8150	141	JPEG
Iphone 3	70	JPEG
Nokia E71	125	JPEG

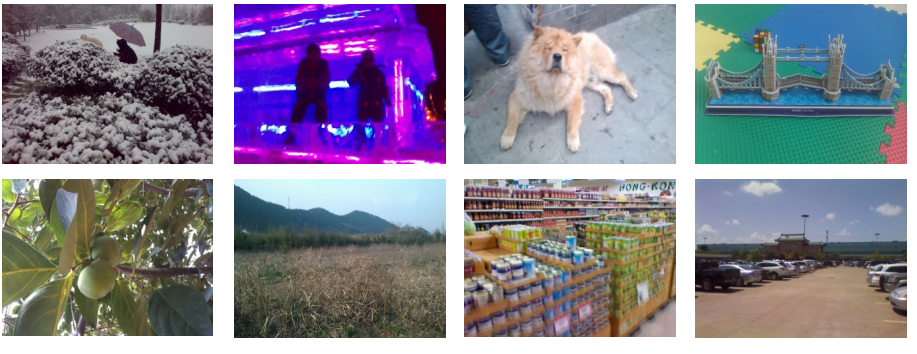


Fig. 1. Sample images used in our experiment

All EXIF (Exchangeable Image File Format) data are removed and all these original images are manipulated by using the following six types of operations:

I. All original images are trans-coded to the JPEG format with the standard quality factor of '75';
II. The first four rows and first four columns are cut from original images in spatial domain and the remaining pixel values are trans-coded to JPEG format with standard quality factor of '75';

III. The first four rows and first four columns are cut from the original images, the remaining data are resized by multiplication with the scale factors of 0.7 and 2 respectively in spatial domain, and then trans-coded to JPEG format with standard quality factor of '75';
IV. All original images are resized by multiplication of original image size with the scale factors of 0.3, 0.5, 0.8, 1.5, and 2 respectively in spatial domain, and trans-coded to JPEG format with standard quality factor of '75';
V. The original images are trans-coded to the images with standard quality of '75', cropped to remove the first four rows and first four columns in spatial domain, then resized by multiplication with the scale factors of 0.5 and 1.5, respectively, and trans-coded to JPEG format at quality of '75';
VI. The original images are trans-coded to the images at quality of '75', and resized by multiplication with the scale factors of 0.5 and 1.5 respectively in spatial domain, and then trans-coded to JPEG format at quality of '75'.

To sum up, these six types of operations include different scale parameters, which result in 13 series of operations as summarized in the second column of Table 2. These 13 operations are applied to each type of the total of 599 smartphone images and thus a total of 7787 processed images are generated for our experiments. Since each of the 13 operations are applied to the five different smartphone brands, a total of 65 class labels are generated and listed in Table 2.

Table 2. The 65 class labels in our experiments

Type	Scale factor	Smartphone brand and class label				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
I	/	1	14	27	40	53
II	/	2	15	28	41	54
III	0.7	3	16	29	42	55
	2	4	17	30	43	56
IV	1.5	5	18	31	44	57
	2	6	19	32	45	58
	0.3	7	20	33	46	59
	0.5	8	21	34	47	60
	0.8	9	22	35	48	61
V	0.5	10	23	36	49	62
	1.5	11	24	37	50	63
VI	0.5	12	25	38	51	64
	1.5	13	26	39	52	65

3.2 Experiments

Smartphone Source Identification Using SVM

Experiment 1. Our first experiment is to identify the smartphone source of type I images, whose operation is essentially JPEG double compression. At each run, we randomly select 60% of the images from each brand of smartphone, and the remaining 40% images are used for testing. LibSVM [3] is employed for our multiclass classification problem, SVM with linear kernel or RBF kernel with default kernel parameters, is respectively performed over 100 random runs. The mean values of the confusion matrices on testing data are summarized in Table 3.

Table 3. Mean testing accuracy over 100 experiments (type I images)

		Prediction outcome (%) using linear kernel					Prediction outcome (%) using RBF kernel				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71	HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
Actual brand	HTC G3	**98.95**	0	0.86	0.10	0.08	**96.75**	0	1.56	0.29	1.41
	HTC HD2	0	**100**	0	0	0	0	**98.67**	0	0.98	0.36
	Huawei U8150	0.09	0	**99.82**	0.09	0	1.64	0	**97.82**	0.52	0.02
	Iphone 3	3.96	0	0	**95.89**	0.14	4.78	0	0	**90.29**	4.93
	Nokia E71	0.92	0	0.34	0.76	**97.98**	0.02	0	0	1.34	**98.64**

Experiment 2. The same to Experiment 1 except that type III images with the scale factor value of 2 are used. The results are shown in Table 4.

Table 4. Mean testing accuracy over 100 experiments (type III with scale factor of 2)

		Prediction outcome (%) using linear kernel					Prediction outcome (%) using RBF kernel				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71	HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
Actual brand	HTC G3	**91.90**	0.27	6.13	1.54	0.15	**72.24**	0	1.08	2.41	24.27
	HTC HD2	1.40	**98.13**	0.38	0	0.09	0.98	**67.64**	0.11	4.27	27.00
	Huawei U8150	13.05	0.02	**86.14**	0	0.79	15.61	0.07	**55.04**	3.30	25.98
	Iphone 3	3.71	0.64	0.14	**95.07**	0.43	0.61	0.79	0	**76.93**	21.68
	Nokia E71	1.92	0.14	1.22	0.36	**96.36**	0.62	0	0.08	1.86	**97.44**

Experiment 3. The same to Experiment 2 except that the type IV images with the scale factor value of 0.5 are used. Experimental results are listed in Table 5.

Table 5. Mean testing accuracy over 100 experiments (type IV with scale factor of 0.5)

		Prediction outcome (%) using linear kernel					Prediction outcome (%) using RBF kernel				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71	HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
Actual brand	HTC G3	**82.43**	1.20	6.66	6.88	2.83	**77.15**	1.05	14.83	2.01	4.95
	HTC HD2	1.11	**89.27**	3.13	4.62	1.87	2.62	**77.04**	5.73	3.51	11.09
	Huawei U8150	5.88	1.80	**83.38**	8.13	0.82	9.41	0.63	**83.61**	1.98	4.38
	Iphone 3	18.32	12.00	14.25	**42.71**	12.71	19.29	11.96	12.18	**28.96**	27.61
	Nokia E71	2.56	5.48	1.38	3.24	**87.34**	6.16	12.74	3.18	2.60	**75.32**

Experiment 4. All types of images with all scale factors are used in this experiment. The experimental results are shown in Table 6.

Table 6. Mean testing accuracy over 100 experiments (all type images with all scale factors)

		Prediction outcome (%) using linear kernel					Prediction outcome (%) using RBF kernel				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71	HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
Actual brand	HTC G3	**80.11**	0.79	12.07	1.77	5.27	**89.02**	0.43	9.20	0.14	1.21
	HTC HD2	1.23	**84.31**	1.58	5.94	6.94	0.88	**94.92**	0.59	1.63	1.98
	Huawei U8150	10.47	0.60	**82.67**	3.35	2.92	5.20	0.36	**93.35**	0.30	0.79
	Iphone 3	8.22	5.75	6.40	**70.15**	9.48	4.86	2.14	4.91	**81.30**	6.79
	Nokia E71	5.50	8.43	1.85	8.40	**75.83**	2.01	2.97	1.17	1.91	**91.95**

Tables 3-6 indicate that it is possible to identify the original smartphone of captured images, although the original images have been differently manipulated. For Types I and II images, the accuracy of the smartphone source identification is apparently higher than the detection accuracy for the other types. Type I involved only double JPEG compression, while Type II was manipulated by cropping, followed by double-compression. It implies that double compression has major impacts on the DCT coefficients in Type II. In addition to double compression, images in other types were rescaled. For these types, double compression may or may not have major

impacts on the modification of the property of DCT coefficients of the processed images, depending on the parameter value of rescaling.

Smartphone Source and Manipulation Identification Using SVM

The experimental results in Tables 3-6 have demonstrated that the detection performance varies across different manipulations. In other words, some manipulations may lead us to low-reliability judgment of the smartphone source if we are not aware of the manipulation. From the image forensics perspective, it is more necessary to identify these different manipulations as well as to recognize the smartphone source. Therefore, aiming to simultaneously detect the smartphone source and the manipulation type, we intentionally mix all the smartphone images of the 65 classes shown in Table 2. As for the smartphone source identification, we randomly select 60% images from each class for training and the remainings for testing over 100 random runs. The mean values (over 100 experiments) of the confusion matrix on the 65 classes are visualized in Figure 2(a) and 2(b). The resulting performance is highly encouraging, compared to the average hit rate under random guess (i.e., 1/65 or 1.5%) for each class.

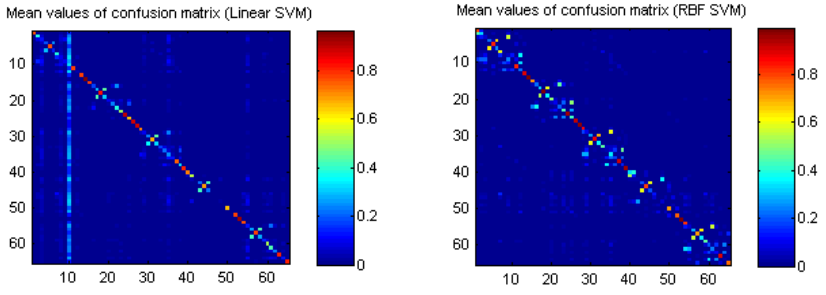


Fig. 2. Mean confusion matrix over 100 experiments ((a) and (b))

In addition, the results in Figure 2 validate our analysis in the experiments of smartphone source identification as follows. Some useful information in identifying the smartphone source might be removed by some manipulations, such as downscaling and recompression. In such cases, the effect of cropping is ignorable and thus not easily distinguishable, resulting in the outcome that fail to discriminate between class 49 (Iphone3 images in type V of scale parameter 0.5) and class 51 (Iphone3 images in type VI of scale parameter 0.5). Therefore, labeling all the images into the 65 classes may not be the best class assignment, since the classification is simply based on all the possible combination of the manipulations and smartphone sources, while ignoring latent complexity hidden in different manipulations.

Smartphone Source and Manipulation Identification with Hierarchical Agglomerative Clustering (HAC) and SVM

In what follows, we discuss how to re-label the 65 classes so as to tackle the aforementioned labeling issue and reflect the complexity in different manipulation into the learning process. First, we take the average of the feature vectors from training data and then apply HAC to the average feature vectors, originally labeled by the numbers from 1 to 65 (Table 2). Figure 3 shows the hierarchical binary clustering tree by using

single linkage with the usual pairwise Euclidean distance. As depicted, original class pairs, 30 and 32, 4 and 6, 43 and 45, 56 and 58, 17 and 19, or the images in types III and IV with the scale factor 2, are grouped together with the minimal distances. In addition, classes 8, 10, and 12, HTC G3 images, manipulated by type IV, V, and VI operations with the scale factor 0.5, are grouped together. It shows that interpolation (rescale) operation generates a major functionality across different types of manipulation. Furthermore, different scale parameters will produce different impacts and thus result in different pairwise distances. Finally, according to the clustering with HAC as shown in Figure 3, we re-label the original 65-class data into 18 classes. For example, we integrate original classes 30 and 32 into new class 1; classes 4 and 6 into new class 2; and so on. In particular, the new class 10 combines the 24 original classes.

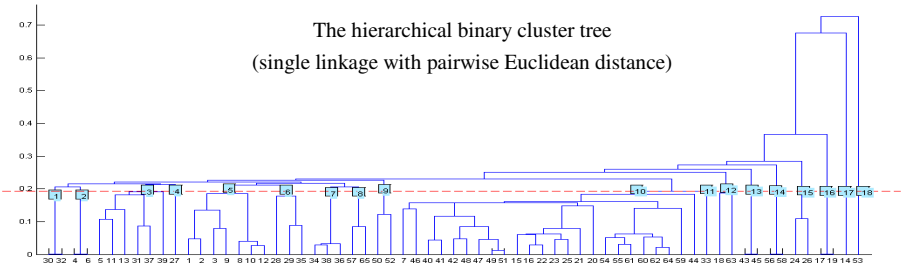
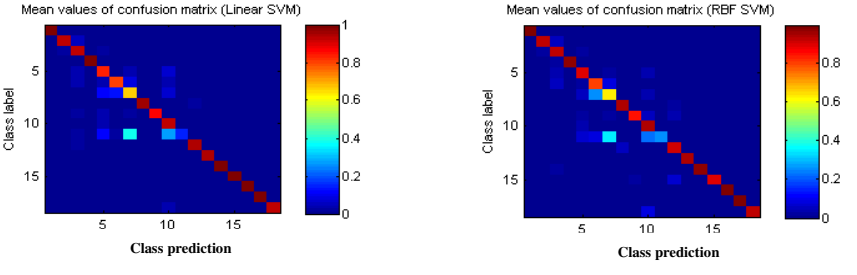


Fig. 3. Hierarchical binary cluster tree with relabeled numbers in the shaded squares



(a) (b)

Fig. 4. Mean confusion matrix over 100 experiments ((a) and (b)) after relabeling classes based on the HAC (single linkage with pairwise Euclidean distance in Figure 3) result

Figure 4 illustrates the average values of confusion matrix over 100 experiments on the newly re-labeled 18 data classes. The experimental procedures are identical to those in the previous experiments. Figure 4 depicts that re-labeled class 11 is prone to be classified as re-labeled classes 7 or 10. In other words, the images captured by Huawei U8150 in type IV with the scale factor 0.3 (original class 33) may not be accurately indentified.

In order to compare the detection performance with that in Table 6, which labels the images only based on the five smartphone sources while ignoring the manipulations, we re-label all types of data with all scale factors into five clusters (classes) as follows:

cluster 1 consists of original classes 30, 32, 4, 6, …, 43, 45, 56 and 58; cluster 2 consists of original classes 24 and 26; cluster 3 consists of original classes 17 and 19; cluster 4 only contains original class 14; and cluster 5 is derived from original class 53. Table 7 summarizes the mean testing accuracy of confusion matrix over 100 experiments. As before, 60% images are randomly selected from each cluster for training and the remaining are used for testing. By comparing the experimental results shown in Table 6 with those in Table 7, we find that the advantage of employing the unsupervised learning (i.e., clustering) prior to the supervised learning (i.e., classification) is noticeable in terms of the overall performance on identifying smartphone source and manipulation simultaneously.

Table 7. Mean testing accuracy over 100 experiments (all type images with all scale factors)

		Prediction outcome (%) using linear kernel					Prediction outcome (%) using RBF kernel				
		HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71	HTC G3	HTC HD2	Huawei U8150	Iphone 3	Nokia E71
Actual brand	HTC G3	**99.90**	0.08	0.01	0	0.00	**99.81**	0.19	0	0	0.00
	HTC HD2	10.32	**89.68**	0	0	0	13.65	**86.35**	0	0	0
	Huawei U8150	1.20	0	**98.80**	0	0	0.52	0	**99.48**	0	0
	Iphone 3	0.09	0	0	**99.91**	0	0.44	0	0	**99.56**	0
	Nokia E71	4.12	0.24	0.42	0	**95.22**	6.16	0	0	0	**93.84**

4 Discussions

To our knowledge, there is no method to detect smartphone sources and reveal the processing history simultaneously. Therefore, we were not able to make a comparison study of performance in our experiments. While using SVM, we only adopted the default kernel parameters for the linear kernel and RBF kernel without performing an exhaustive search to optimize the kernel parameters. The detection performance could be improved by optimizing the kernel parameters. In addition, while using hierarchical agglomerative clustering, we only employed the usual Euclidean distance to measure the pairwise distance and thus other metrics, such as Mahalanobis distance, have not been fully examined. Furthermore, our current smartphone image database is not sufficiently large enough to reflect the complexity of real-life scenarios, and thus more smartphone images and operations with more brands of smartphones should be considered to conduct a more convincing examination. Additionally, although feature selection was conducted in our prior study in multimedia forensics [10, 12], we did not adopt any sophisticated feature selection algorithm, which could be useful to further improve the detection accuracy. All the aforementioned directions will be investigated in future study.

5 Conclusion

In this paper, we propose a method to identify the smartphone source and the manipulations applied to the images under scrutiny. Marginal density and neighboring joint density are extracted as discriminating features, and support vector machines are applied to the features for the detection. Experimental results show that our method is very promising in terms of identifying the smartphone source and revealing manipulations simultaneously. Our study also indicates that it is more reasonable to hybridize clustering and classification techniques together to tackle complex, intentional manipulation, resulting in improved performance.

Acknowledgements. This project was supported in part by Award No. 2010-DN-BX-K223 awarded by the National Institute of Justice, Office of Justice Programs, U. S. Department of Justice. The opinions, findings, and conclusions or recommendations expressed in this publication/program/exhibition are those of the authors and do not necessarily reflect those of the Department of Justice. Partial support from the Research and Sponsored Programs at Sam Houston State University is also greatly appreciated.

References

[1] Alles, E.J., Geradts, J.M.H., Veenman, C.J.: Source camera identification for heavily JPEG compressed low resolution still images. Journal of Forensic Sciences 54(3), 628–638 (2009)

[2] Celiktutan, O., Sankur, B., Avcibas, I.: Blind identification of source cell-phone model. IEEE Trans. Information Forensics and Security 3(3), 553–566 (2008)

[3] Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), http://www.csie.ntu.edu.tw/~cjlin/libsvm

[4] Choi, K.S., Lam, E.Y., Wong, K.K.Y.: Source camera identification using footprints from lens aberration. In: Proc. SPIE, vol. 6069, pp. 172–179 (2008)

[5] Dirik, A.E., Sencar, H.T., Memon, N.: Source camera identification based on sensor dust characteristics. In: Proc. IEEE Workshop on Signal Processing Applications for Public Security and Forensics, pp. 1–6 (2007)

[6] Farid, H.: Image forgery detection, a survey. IEEE Signal Processing Magazine, 16–25 (March 2009)

[7] Gul, G., Avcibas, I.: Source cell phone camera identification based on signular value decomposition. In: Proc. 1st IEEE Inernational Workshop on Informaiton Forensics and Security, pp. 171–175 (2009)

[8] Kharrazi, M., Sencar, H.T., Memon, N.: Blind source camera idetification. In: Proc. of ICIP 2004, pp. 709–712 (2004)

[9] Li, C.T.: Source camera identification using enhanced sensor pattern noise. IEEE Trans. Information Forensics and Security 5(2), 280–287 (2010)

[10] Liu, Q., Sung, A.H., Chen, Z., Xu, J.: Feature mining and pattern classification for steganalysis of LSB matching steganography in grayscale images. Pattern Recognition 41(1), 56–66 (2008)

[11] Liu, Q., Sung, A.H.: A new approach for JPEG resize and image splicing detection. In: Proc 1st ACM Workshop on Multimedia in Forensics, pp. 43–48 (2009)

[12] Liu, Q., Sung, A.H., Qiao, M., Chen, Z., Ribeiro, B.: An improved approach to steganalysis of JPEG images. Information Sciences 180(9), 1643–1655 (2010)

[13] Liu, Q., Sung, A.H., Qiao, M.: Neighboring joint density-based JPEG steganalysis. ACM Transaction on Intelligent System and Technology 2(2), article 16 (2011)

[14] Liu, Q., Sung, A.H., Qiao, M.: A Method to Detect JPEG-Based Double Compression. In: Liu, D., Zhang, H., Polycarpou, M., Alippi, C., He, H. (eds.) ISNN 2011, Part II. LNCS, vol. 6676, pp. 466–476. Springer, Heidelberg (2011)

[15] Liu, Q.: Detection of misaligned cropping and recompression with the same quantization matrix and relevant forgery. In: Proc.of 3rd ACM Workshop on Multimedia in Forensics and Intelligence, pp. 25–30 (2011)

[16] Lukas, J., Fridrich, J., Goljan, M.: Digital camera identification from sensor pattern noise. IEEE Trans. Information Security and Forensic 1(2), 205–214 (2006)

[17] Pevny, T., Fridrich, J.: Detection of double-compression in JPEG images for applications in steganography. IEEE Trans. Information Forensics and Security 3(2), 247–258 (2008)

[18] Tsai, M., Lai, C., Liu, J.: Camera/mobile phone source identification for digital forensics. In: Proc. ICASSP 2007, vol. 2, pp. 221–224 (2007)

Situation-Aware on Mobile Phone Using Co-clustering: Algorithms and Extensions

Hyuk Cho[1], Deepthi Mandava[1], Qingzhong Liu[1], Lei Chen[1], Sangoh Jeong[2], and Doreen Cheng[2]

[1] Department of Computer Science, Sam Houston State University, Huntsville, TX 77341, USA
{hyukcho,dxm015,liu,chen}@shsu.edu
[2] Samsung Electronics R&D Center, San Jose, CA 95134, USA
{sangoh.j,doreen.c}@samsung.com

Abstract. Due to the large number of applications in the mobile phones, users usually go through a fixed menu hierarchy to find a specific interesting application. Hence, in our previous research, we realized the proactive mobile phone application recommendation using co-clustering and demonstrated the promising recommendation performance on a smartphone. The approach first autonomously extracts user's behavioral patterns from the usage log of user interactions with the device as well as environments and then recommends potential applications that might be interesting to the user at the corresponding specific situation. In this paper, as a follow-up to this novel platform of intelligent smartphone-based situation-awareness, we investigate sophisticated methodologies that lead to better performance. To achieve this goal, we considered various co-clustering algorithms with different data transformations and weighting schemes for simulated mobile phone usage data. Through non-exhaustive, but pretty comprehensive experimental setting, we find what specific co-clustering algorithms with what specific data transformations and weighting schemes improve accuracy performance in extracting specific user patterns.

Keywords: co-clustering, data transformation, weighting, pattern extraction, situation aware, recommendation system.

1 Introduction

Mobile phones have a huge number of applications in a predefined menu hierarchy (e.g., listening to music, finding location, and so on). Usually, users go through the fixed menu hierarchies to locate a particular application that is interesting to use at a particular time. In addition, currently-available recommendation systems are bound to a small set of predefined situations on which recommendations are determined. Furthermore, users are sometimes asked to provide detailed situation definitions and also to train the system so as to get proper recommendations.

The desirable solution is for mobile phones to have ability to perceive the environmental status via sensors (sentience), to operate independently from user's involvement in a decentralized manner (autonomy), and to proactively recommend

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 272–282, 2012.

proper applications to the users based upon the user's situation or context (i.e., proactiveness) [1]. To achieve these objectives, several approaches are proposed, including, but not limited to, clustering algorithms (e.g., LBG algorithm and co-clustering algorithms [2]) and classification algorithms (e.g., Naïve Bayesian classification [3] and Support Vector Machine [4]).

The ultimate goal of smartphone-based context awareness is to reduce user's involvement and also provide a personalized recommendation. To address this goal, we recently proposed a novel approach based upon co-clustering and showed its applicability for mobile phone recommendation systems [2]. In this paper, we extend and improve the previously proposed co-clustering approach through various algorithmic improvements. First, we consider five different data transformations to mobile usage log data. Then, we apply different co-clustering algorithms to the transformed data. Finally, we investigate the performance to find what combination of the algorithmic consideration results in improving the performance in terms of clustering accuracy.

The rest of this paper is organized as follows: In Section 2 we briefly discuss some research that motivates the current approach. We describe the considered co-clustering algorithms, data transformations, weighting schemes in Section 3. Then, we formally analyze the effects of data transformations and summarize the analysis results in Section 4. We discuss the experimental details with simulated mobile phone usage log data in Section 5. Finally, the paper is concluded with some remark.

2 Motivation

Many mobile phone-based context aware systems are proposed to provide applications or services to the phone user, where the context awareness operates based on the current context of the user. The current situation (context) can be retrieved in various ways like device status, network information, and other physical resources.

The initial context aware system, "Active Badge Location System", was introduced by Want et al [12] in 1992, where the context is defined with the location of the user. The system was based on the infrared technology to determine the user's current location in order to forward a phone call to a nearby user. GOOG-411 [13] is another context aware system developed for a voice-powered directory assistance service. The Google server recommends services to the users based on the user's location. Focusing on the context framework [14], MobiLife was developed with Nokia and Motorola phones to bring advances in mobile applications and services within the reach of users in their everyday life. K-SCM algorithm [15] was developed on a Nokia phone to provide online clustering of symbol string data, showing how clusters can be associated with higher level contexts. The context recognition based on the fusion of information sources is formulated as the clustering of symbol string data. This approach can be used to assist user interaction with the device.

Our previous research [8] was the first that applied co-clustering algorithm for task recommendations in mobile phones through situation awareness, where Minimum Sum Squared Co-clustering (MSSRCC) was employed to derive the latent patterns at specific situations and environments. Furthermore, we compared a supervised

approach (i.e., Naïve Bayesian classification) with two unsupervised approaches (co-clustering and vector quantization (VQ)) [2], where the experiments showed that their co-clustering approach gave a comparable recommendation performance in terms of purity, while requiring less computation time than VQ.

3 Proposed Work

In what follows, we summarize our previously proposed context-aware recommendation system using co-clustering [2, 8], since it is the framework on which we build up the extended research proposed in this paper.

The main steps include data preparation, data encoding, data transformation, pattern extraction, and recommendation. First, user's log that contains both context variables and corresponding application usages is prepared in a raw data format; the raw data is encoded and transformed as explained in sequel; the encoded and transformed data is used as an input to co-clustering algorithm to extract user's specific patterns, resulting in row clustering centroids; and finally these user specific patterns are utilized for both situation recognition and task recommendation.

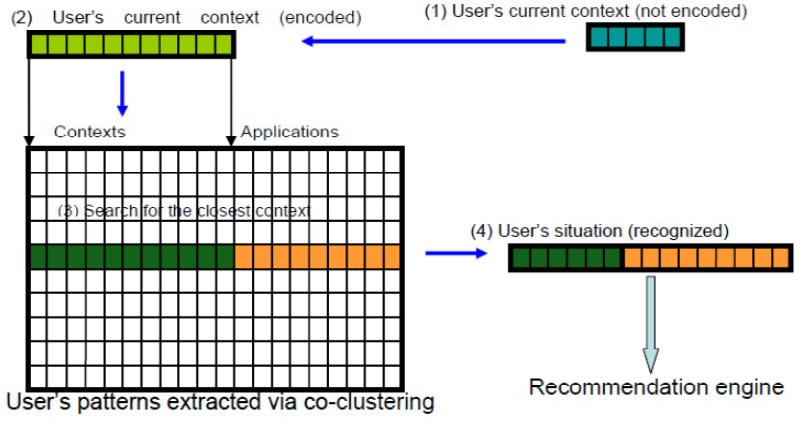


Fig. 1. Situation Recognition and Recommendation

To be more specific, if we request k row clusters, k specific user-patterns will be generated as depicted in the left bottom block (denoted as user's patterns extracted via co-clustering) of Figure 1. Therefore, the user's patterns, each of which consists of context variables and the corresponding applications, are resulted from co-clustering. Note that user's patterns are already encoded, transformed, and co-clustered, and stored in the encoded, transformed form for the situation recognition and task recommendation. Figure 1 depicts the overall steps of situation recognition and application recommendation. First, user's current context variables (not in encoded form (step (1)) comes in user's patterns. However, the current context variables cannot be directly compared with the encoded user's patterns. Thus, the first step is to encode user's current context (step (2)). Then, the encoded context is compared to each of the

context in user's patterns (step (3)). Finally, the closest (i.e., best matched) user pattern is selected (step (4)). The chosen user pattern consists of both context and application, where only the application information is utilized in recommendation.

As an extension, we propose to consider varied algorithmic strategies to enhance the recommendation performance. First, the same data encoding schemes are applied, however different data transformations are considered to the encoded data set to get normalized data. Then, varied co-clustering bases are applied to the normalized data. Finally, the extracted patterns are utilized at situation recognition, where appropriate applications are recommended, based on user's situation matched to the usage pattern.

3.1 Data Preparation

We assume all the data are collected from the various sensors in real situations. However, not all the sensors are available in mobile phones, thus we reuse the same synthetic data [2, 8], which simulate a phone usage log of daily-life situations typical to a user. Each row in the resulting data consists of both context and application variables that together simulate one of the following 12 specific situations: (1) driving along, (2) using public transportation, (3) working in cubicle, (4) playing tennis, (4) meeting at work, (6) staying at home, (7) strolling around at work, (8) checking weather, (9) travelling during weekend, (10) spending time with family, (11) shopping around, and (12) going to church. We consider the usage log data that contains 3,070 situation vectors in total, where each of 12 situation groups consists of the following number of situation vectors generated according to each user-defined statistic, respectively: 284, 258, 233, 293, 269, 270, 225, 223, 201, 254, 284, and 276.

3.2 Data Encoding

Since the data are assumed to be collected from sensors, each data entry ranges in a different unit. Therefore, data preprocessing is required to map the values in a unified value. Accordingly, we employ data encoding and data transformation schemes at the preprocessing step. We use both 1-of-N encoding [5] and binning.

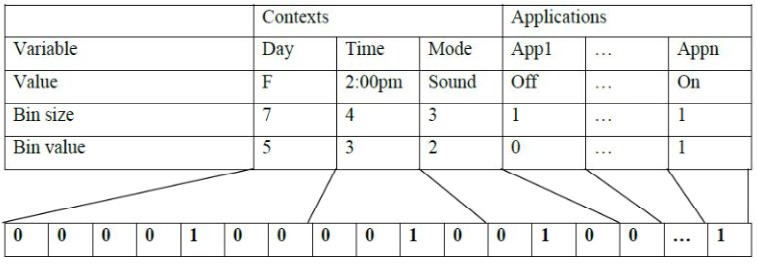


Fig. 2. 1-of-N encoding and binning

- **1-of-N encoding.** An integer starting from 0 is assigned to each of context variables, while each application entry uses binary encoding (i.e., 0 or 1) to indicate

whether it is off (i.e., 0 for no use) or on (i.e., 1 for use). Figure 2 illustrates an example of the 1-of-N encoding and binning techniques that are applied to both contexts and applications. For example, bin size of context variable "Day" is 7, since it contains seven values, each of which is assigned to one of seven days in a week. If it is on Friday the bin value is 4. Notice that every application's bin size is 1, since 0 or 1 is assigned to specify "off" and "on", respectively.

- **Binning.** Each context variable in a different continuous (or categorical) unit is mapped into fixed size of discrete values as illustrated in Figure 2. However, we consider the total number of applications as the corresponding bin size, since we consider only one application runs at a time.

3.3 Data Transformation

Transformation of the raw data is considered one of the most important steps for various data mining processes since the variance of a variable will determine its importance in a given model [6]. We apply the following different data transformation schemes and observe the performance of each data transformation on the co-clustering algorithm.

- **No Transformation (NT).** The centering or scaling is not taken. In other words, $a'_{ij} = a_{ij}$, where a_{ij} denotes row i and column j of data matrix and a'_{ij} denotes the transformed value of a_{ij}.
- **Double centering (DC).** It is defined as $a'_{ij} = a_{ij} - a_{i\cdot} - a_{\cdot j} + a_{\cdot\cdot}$ for $i = 1, \cdots, m$ and $j = 1, \cdots, n$, where $a_{i\cdot}$ is the mean of row i, $a_{j\cdot}$ is the mean of the column j, and $a_{\cdot\cdot}$ is the mean of all the elements. When double centering is applied to data matrix, the row mean and column mean becomes zero.
- **Mean centering (MC).** Column MC is defined as $a'_{ij} = a_{ij} - a_{\cdot j}$ and row MC is defined as $a'_{ij} = a_{ij} - a_{i\cdot}$.
- **Standard deviation normalization (SDN).** Column SDN is defined as $a'_{ij} = a_{ij}/\sigma_{\cdot j}$ for $i = 1, \cdots, m$ and $j = 1, \cdots, n$, where $\sigma_{\cdot j}$ is the standard deviation of column j. Row SDN is defined similarly with $\sigma_{i\cdot}$. Through column SDN each column has a unit variance and through row SDN each row has a unit variance.
- **Z-score transformation (ZT).** Column ZT is defined as $a'_{ij} = (a_{ij} - a_{\cdot j})/\sigma_{\cdot j}$ for $i = 1, \cdots, m$ and $j = 1, \cdots, n$. Row ZT is defined similarly as $a'_{ij} = (a_{ij} - a_{i\cdot})/\sigma_{i\cdot}$. It is also called "autoscaling", where each row/column of the transformed data matrix will have zero mean and unit variance.

3.4 Co-clustering Bases

We briefly review Bregman co-clustering (BCC) algorithm [7], since BCC is main learning algorithm through which we extract (or recognize) user's peculiar patterns. BCC frame unifies six Euclidean co-clustering schemes and six I-divergence co-clustering schemes. Each co-clustering algorithm is formulated with specific summary statistics. For example, basis 1 is with row cluster average and column cluster average; basis 2 is with co-cluster centroid; basis 3 is with row marginal and

co-cluster centroid; basis 4 is with column marginal and co-cluster centroid; basis 5 is with row marginal, column marginal, and co-cluster centroid; and basis 6 is with row cluster centroid, column cluster centroid, and co-cluster centroid. Detailed discussion on co-clustering bases can be found in [7, 9, 11]. Through data transformation, data entries can be negative values, which are not allowed in I-divergence co-clustering algorithms. As a result, we use the six Euclidean co-clustering algorithms in this paper. We evaluate the effect of varied algorithmic strategies on the accuracy performance and discuss the experimental results in next section.

4 Experimental Results

We investigated varied combinations of input parameters for the comparison study to find the best performing combination. The input parameters considered include co-clustering basis, data transformations, application weight of the user pattern vector, and number of column clusters. Note that we fix row cluster number to be 12, since as discussed earlier we use the synthetic data with 12 specific user's situations (i.e., *a priori* assigned 12 classes). All these input parameters are inter-related and thus we want to investigate what combination improves accuracy performance. We assess situation clustering quality according to *a priori* assigned situation class labels, using the external evaluation measure, defined as *accuracy* (%) = $1/T\ (\sum_{i=1}^{k} \mathrm{t_i})\ *\ 100$, where T denotes the total number of context vectors (e.g., T=3,070 for our simulated phone usage data), k denotes the number of situation (i.e., row) clusters, and t_i denotes the number of context vectors correctly clustered into situation class i. We first form a confusion matrix whose (i,j)-th entry counts the number of context vectors in cluster i that belong to the true situation class j. Each t_i is a diagonal element of the corresponding confusion matrix whose cluster labels are permuted so that the sum of diagonal elements is maximized [2]. The situation recognition and recommendation accuracy values are measured by comparing the clustering results with the original user specifications upon generating the usage log. To enhance the robustness of the experimental results, all the accuracies are averaged over 20 random runs and the corresponding standard deviation values are given in the parentheses.

As a preliminary experiment, we investigated how much the number of data points affects accuracy performances of situation recognitions and witnessed that the accuracy performance is improved as more data points are employed (result not shown). This suggests that we need to use enough number of data to faithfully reflect the specification at data generation and discover better latent patterns at co-clustering step.

4.1 Effect of Co-clustering Basis

All the six Euclidean co-clustering algorithms are applied to the normalized data obtained from each of the five normalizations. Performance is evaluated for each co-clustering algorithm in terms of the clustering accuracy. In our experiment, we fix $k = 12$ (i.e., number of row clusters or number of 12 situations), since we evaluate the co-clustering performance with the external evaluation measure for the known 12 situation classes. In addition, we present the performance with $l = 4$ (i.e., number of column clusters), since it results in more stable performance than other values (not

shown) for our data. Furthermore, weight for application part of each situation vector is to be one (i.e., 1). By fixing these parameters, we focus more on the effect of six co-clustering bases with two data transformations, column ZT and column SDN.

Figure 3 shows the performance of different co-clustering bases, among which co-clustering with basis 6 gives the best result, which is also consistent with the analysis and experimental results [9, 11]. To be more specific, we obtain 88.1(±2.1)% accuracy with column SDN and 89.4(±1.5)% with column ZT. Interestingly, the data transformation effect between column SDN and ZT is negligible over all the six co-clustering bases. The similarity between column SDN and ZT was analyzed in [9, 11].

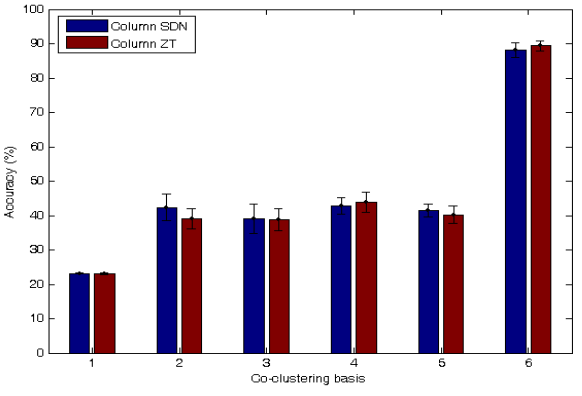


Fig. 3. Effect of co-clustering basis ($k = 12$, $l = 4$, and application weight = 1)

4.2 Effect of Data Transformation

We also investigate how much each data transformation affects the overall accuracy performance. As before, we set $k = 12$, $l = 4$, and application weight = 1. In addition, we use basis 6, since we learn that it is most robust among all the six bases as shown in Figure 3. Notice from Figure 4 that we obtain the worst accuracy

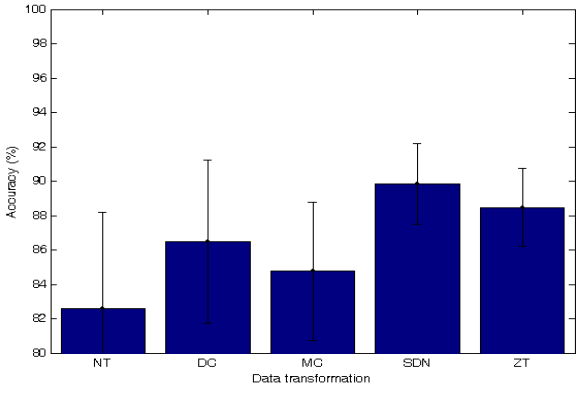


Fig. 4. Effect of data transformation (basis 6)

performance with no normalization (denoted as NN) and column SDN leads to the best accuracy performance of 89.8(±2.4)% with basis 6 co-clustering. As expected, ZT results in the compatible accuracy to that of SDN. Cho [9-11] analyzed the effect of various data transformations with all the six co-clustering bases, where the similar effect of the two data transformation was characterized in terms of removing hidden scaling and shifting patterns in a data matrix.

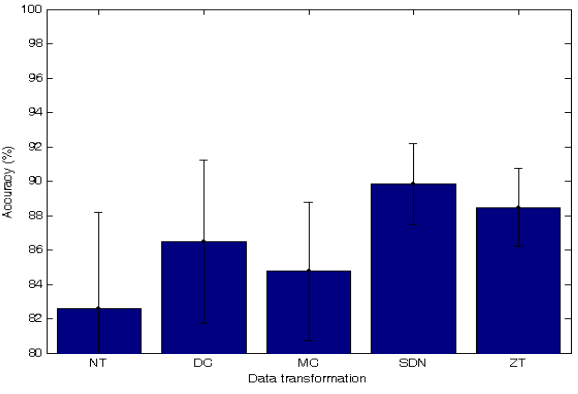


Fig. 5. Effect of data transformation (basis 6)

4.3 Effect of Application Weight

As explained in data encoding, the weight of all the applications is given one (i.e., 1) if users use the application for a particular situation, otherwise the application will be assigned zero (i.e., 0). While, the context part uses categorical values starting from 0. Therefore, there is a difference in encoding context variables and applications, where each context is binned (i.e., extended) in a corresponding variable-sized bin, however applications are assigned in a fixed unit-sized bin. Therefore, we want to find out how much accuracy changes with different weights to application part of each situation vector. From previous experiment, we achieve a good accuracy with the combination of basis 6 with column SDN (or column ZT). Therefore, while fixing this setting (i.e., basis 6 with column ZT), we vary column clusters and application weights.

From the non-exhaustive experiments, we find that the stable accuracy performance is achieved with $l = 7$ or $l = 10$ column clusters and with the application weight = 4 as shown in Figure 5. We achieve the best accuracy of 90.8(±3.6) with the combination of basis 6 with ZT, $l = 7$, and application weight = 4. The second best accuracy of 89.6(±2.1) is achieved with $l = 10$. Notice that the combination we find from the experiments might not be the best, since non-exhaustive search is taken and also some randomness can be involved. Therefore, in what follows, we investigate the effect of column cluster number on accuracy.

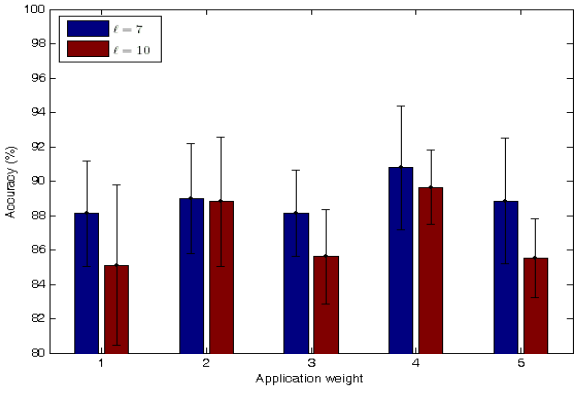


Fig. 6. Effect of application weight (column ZT)

4.4 Effect of Column Clusters

Row (i.e., situation) and column clusters are important variables that can affect the overall performance in our comparison study. As before, the number of row clusters is fixed to 12 (i.e., $k = 12$), since we consider them as *a priori* assigned situation classes (i.e., known factor) for the simulated phone usage log data. In this experiment, we want to estimate the best number of column clusters, fixing basis 6 with column SDN or ZT, while fixing application weights as well. As shown in Figure 6, with column SDN, we achieve the best accuracy of 90.9(±3.0)% with basis 6 with 3 column clusters (i.e., $l = 3$). While, with column ZT, we achieve the best accuracy of 90.4(±2.5) with 4 column clusters (i.e., $l = 4$). As we expect from the previous experiments, the accuracy difference between column SDN and column ZT is not significant. The accuracy performance varies over the different numbers of column clusters for both the cases as shown in Figure 6, however the overall performance with $l = 1$ to 4 column clusters is consistently better than those with $l > 4$ column clusters.

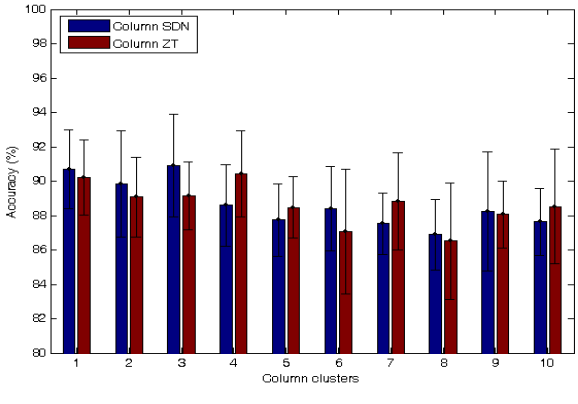


Fig. 7. Effect of column clusters (basis 6)

5 Conclusion and Future Work

To the best of our knowledge, we are the first who have applied co-clustering to situation-aware task recommendation problem in pervasive mobile environment and demonstrated its applicability with the prototype on a smartphone [2, 8]. The proposed approach has satisfied many desired properties of situation-aware systems as follows: (1) no need for predefined situations, (2) no need for user-defined profiles, (3) no need for user to train the system, (4) ability to adapt to user habit changes, and (5) considerations of many context variables, not just location and time.

As an extension to our prototype co-clustering-based situation-aware, we intend to investigate varied sophisticated methodologies, including data encodings, data transformations, co-clustering bases, application weights, and column clusters. From the experiments we validated co-clustering with basis 6 generally gives good accuracy results independent from other specific strategies. In addition, with specific data transformations, in particular column SDN and column ZT, we almost always obtain the best accuracy as mathematically analyzed in [9,11].

In this research, we focus on investigating accuracy performance with varied preprocessing and co-clustering strategies, thus the experimental study is limited to comparison for only co-clustering. Although we provided some comparison study in [2, 8], it is more desirable to do rigorous comparison study with other existing approaches. Currently, we assume patterns are extracted from a fixed usage log data in a mobile device, however it is more practical to provide the incremental extraction of dynamic user patterns from between- and within-environments. In addition, the privacy issue in collecting user's usage data has been ignored and thus the emerging issues to protect user privacy should be seriously considered in the near future.

The proposed approach can be applicable to digital information forensics. For example, criminals' behavioral and environmental patterns can be extracted from handheld devices retrieved at crime scenes by applying the pattern extraction and situation-aware algorithms. Furthermore, the proposed algorithms can be applied to extract web-usage patterns from web-access logs of suspects.

Acknowledgements. This project was supported in part by the Research and Sponsored Programs (RSP) at Sam Houston State University under Enhanced Research Grant (ERG).

References

[1] Biegel, G., Cahill, V.: A framework for developing mobile, context-aware applications. In: PerCom 2004, pp. 361–365 (May 2004)

[2] Jeong, S., Kalasapur, S., Cheng, D., Song, H., Gibbs, S., Cho, H.: Clustering and naïve bayesian approaches for situation-aware recommendation on mobile devices. In: ICMLA 2009, pp. 353–358 (December 2009)

[3] Kim, Y., Cho, S.: A recommendation agent for mobile phone users using bayesian behavior prediction. In: UBICOMM 2009, pp. 283–288 (October 2009)

[4] Oku, K., Nakajima, S., Miyazaki, J., Uemura, S.: Context-aware SVM for context-dependent information recommendation. In: MDM 2006 (May 2006)
[5] Mitchell, T.: Machine Learning. McGraw-Hill (1997)
[6] Sánchez, F.C., Lewi, P.J., Massart, D.L.: Effect of different preprocessing methods for principal component analysis applied to the composition of mixtures: Detection of impurities in HPLC-DAD. Chemometrics and Intelligent Laboratory Systems 25(2), 157–177 (1994)
[7] Banerjee, A., Dhillon, I.S., Gosh, J., Merugu, S., Modha, D.S.: A generalized maximum entropy approach to Bregman co-clustering and matrix approximation. JMLR 8, 1919–1986 (2007)
[8] Cheng, D., Song, H., Cho, H., Jeong, S., Kalasapur, S., Messer, A.: Mobile situation-aware task recommendation application. In: NGMAST 2008, pp. 228–233 (September 2008)
[9] Cho, H., Dhillon, I.S.: Effect of data transformation on residue. Technical Report TR-07-55, Dept. of CS, The University of Texas at Austin (2007)
[10] Cho, H., Dhillon, I.S.: Co-clustering of human cancer microarrays using minimum sum-squared residue. In: IEEE/ACM TCBB, pp. 385–400 (July- (September 2008)
[11] Cho, H.: Data Transformation for Sum Squared Residue. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010, Part I. LNCS (LNAI), vol. 6118, pp. 48–55. Springer, Heidelberg (2010)
[12] Want, R., Hopper, A., Falcao, V., Gibbons, J.: The active badge location system. ACM Transactions on Information Systems 10, 91–102 (1992)
[13] GOOG-411, `http://www.google.com/goog411/index.html`
[14] Boda, P.: Developing context-aware and personalized multimodal applications in the MobiLife EU project. In: ICMI 2005 (October 2005)
[15] Flanagan, J.A.: Unsupervised Clustering of Context Data and Learning User Requirements for a Mobile Device. In: Dey, A.K., Kokinov, B., Leake, D.B., Turner, R. (eds.) CONTEXT 2005. LNCS (LNAI), vol. 3554, pp. 155–168. Springer, Heidelberg (2005)

Persian Text Classification Based on K-NN Using Wordnet

Mostafa Parchami, Bahareh Akhtar, and MirHossein Dezfoulian

Bu-Ali Sina University, Department of Computer Engineering, Hamedan, Iran
{Parchami,Akhtar,Dezfoulian}@basu.ac.ir

Abstract. K-NN is widely used for text classification purpose. Basic K-NN has poor accuracy; other methods should be applied to basic K-NN to improve accuracy and efficiency. In this paper we propose a method that uses wordnet to increase similarity of documents under the same category. Documents are represented by single words and their frequencies, by using wordnet, frequency of related words is changed to acquire higher accuracy. Information gained is used to eliminate terms that are not discriminated. Words like "and", "or" and "that" in English are not important in text classification and the best way to eliminate them is to calculate their information gain. PCA is used to reduce number of features and increase speed of the method. Applying this method, we designed a faster and much accurate classifier for Persian language. Experiments show that applying this preprocessing will increase accuracy and speed of K-NN. Accuracy of the proposed K-NN classifier on Hamshahri corpus is 88.18%.

Keywords: Text Classification, K-nearest Neighbors, Wordnet, Principle Component Analysis, Persian language, Information gain.

1 Introduction

By daily increase in the number of documents over internet it becomes more and more important to classify these documents under appropriate categories because finding similar documents is important for many purposes. It is much faster to search inside a relevant category to find documents about a special subject.

Text classification is the task of assigning one or two classes to a passage from a set of predefined classes [1].

Text classification has been investigated through recent years and many different classifiers have been tested for this purpose. History of text classification started at the beginning of the 60th but is known as a subset of information systems since 90th. Text classification has been widely used in different applications such as indexing based on controlled vocabularies, text refinement, and automatic hyper-text generation, extracting emotional information from text and generally in all applications that need document classification or selective and adaptive distribution of documents [2].

Text classification has two main steps which are: Indexing and classification. The first step is to index documents for training phase of the classifier and the second step is to classify text using the trained classifier.

Document representation has an important impact on the classification task. The simplest and most widely used way is to represent the document by its words but

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 283–291, 2012.

using stem of words instead of words themselves is more popular. Sometimes N-grams are used instead of single words. In this way document is represented by a string of N words [1].

One issue in text classification is large number of training sets; Number of features extracted for text classification is too big and needs a huge space to store the index. A compression on feature space should be done to decrease number of features. Indexing of stored data is important in retrieving speed. As regards to the size of index file and frequency of operations on this file, considering speed of retrieval is critical in design of indexers.

Text classifiers use similarity measurement to classify documents under relevant class. Similarity function is applied to the vector of features and the distance between the query and documents in the indexed database is calculated. This shows the importance of document representation and features extracted from the document. The simplest way is to only count number of occurrence of different words in the document and classify documents based on this feature vector but "tf" is the most popular feature used to classify documents.

Simplicity, accuracy and efficiency are three important factors in text classifiers design. The speed of a classifier depends on its algorithm and the indexing method used to index labeled database. The accuracy could be improved by extracting better features, using better document representation and using an appropriate similarity measurement function.

A wide range of supervised learning algorithms has been applied to this area, such as K-Nearest Neighbor (KNN) [3], Centroid-Based Classifier (CB) [4], Naive Bayes [3], Decision Trees [3], Winnow [5], Voting [6], and Support Vector Machines (SVM) [3].

Among all these algorithms, K-Nearest Neighbor is a widely used text classifier because of its simplicity and efficiency. Its training-phase consists of nothing more than storing all training examples as classifier, thus it has often been called as lazy learner since "it defers the decision on how to generalize beyond the training data until each new query instance is encountered" [6], [7].

The rest of the paper is organized as follows. In section 2 we will describe feature extraction and document representation. In section 3, design of the classifier is described. Section 4 presents experimental results and in section 5, the conclusion of the obtained results in section 4 is described.

2 Feature Extraction

In the simplest way, a document can be viewed as a set of words. For document classification purpose, a document can be represented by frequency of its words but a single word may be used in its different grammatical forms and still have the same meaning.

Using a stemmer can eliminate the above problem, so here we use a stemmer and replace stem of words in the document. By stemming similarity of documents having same words but in different forms is increased.

After replacing stem of words in documents, frequency of these stems in the whole database and each document is obtained. However frequency of stems is the most popular feature used in text classification, it is not a good discriminating feature. Many high frequency terms like "and", "or", "the" and so on are not suitable for distinguishing the topic of the text. So these high frequency words should be eliminated. Yet there is another problem in Persian language, some of these words are used as a prefix to another word and change the meaning and even the category of the document. To cross this problem we use information gain.

If we have C_i classes where i ranges from 1 to m, the information gain of feature t shown by InfoGain(t) is calculated by equation (1)[8].

$$InfoGain(t) = P_r(t) \sum_{i=1}^{m} P_r(C_i \mid t) \log_2 \frac{P_r(C_i \mid t)}{P_r(C_i)} + P_r(\bar{t}) \sum_{i=1}^{m} P_r(C_i \mid \bar{t}) \log_2 \frac{P_r(C_i \mid \bar{t})}{P_r(C_i)} \quad (1)$$

Where $P_r(C_i)$ is calculated by dividing number of documents pertaining to class C_i by total number of documents. $P_r(t)$ is calculated by number of documents that feature t appeared at least once in those documents and $P_r(C_i \mid t)$ is number of documents in class C_i that feature t appeared at least once in them. $\bar{t}$ stands for documents that feature t, did not appear in the document.

Here obtained features are only frequency of stems. After calculating information gain of each word, words with high frequency and low information gain should be eliminated.

To eliminate these words, if frequency of a word is within the top 20% of frequencies and its information gain is within the last 20% values of information gains then the word is eliminated and its frequency is not preserved in the feature vector. This makes sure that word with low impact on the classification eliminated.

The structure of Persian language is different from Latin languages. The first and the most important difference is that Persian is written from right to left and becomes more complex when numbers and formulas that are written from left to right are added to the document. Another difference is that Persian language does not have a strict grammar; so that a single statement can be rewritten in many different ways and still have the same meaning. The latter difference makes N-gram representation seem inapplicable. To address these problems we used single word representation and then replacing stem of words.

Replacing stems, increases similarity of documents in the same category which have same words in different grammatical forms. Still other preprocessing could be done on the document to increase similarity of documents in the same category. We used wordnet as a tool to make classification more precise.

WordNet [9] is a lexical database of English Language, which groups nouns, verbs, adjectives and adverbs into sets of synonyms[1], each expressing a distinct concept. Synsets are interlinked with conceptual-semantic and lexical relations, including hypernymy, meronymy, causality, etc [10].

[1] Synset.

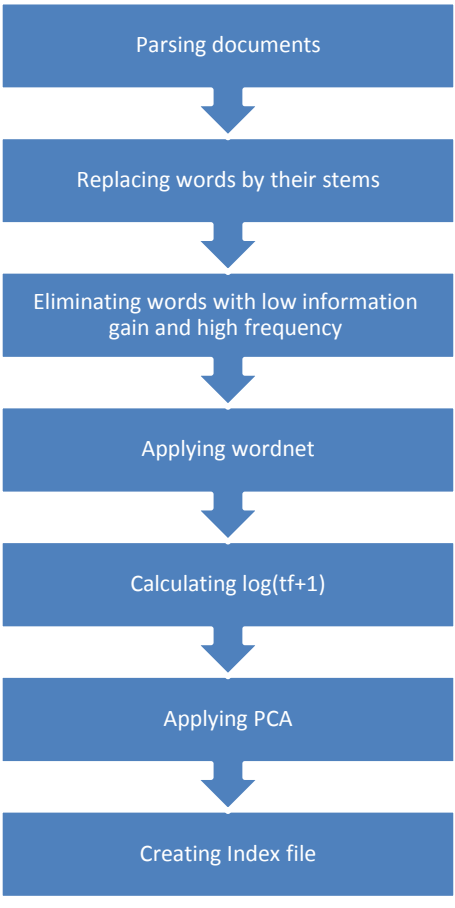


Fig. 1. Indexing Process

The first wordnet was developed for English language at Princeton University (PWN). Over time it has become one of the most valuable resources in applications for natural language understanding and interpretation, such as word-sense disambiguation, information extraction, machine translation, document classification, and text summarization and, last but not least, Semantic Web applications [9], [11].

Two Persian wordnets developed in [12] and [13]. We have used wordnet ontology to increase similarity. For this purpose each word in the document is given to the wordnet and if related word suggested by the wordnet is used at least once in the document, its frequency is replaced by the maximum frequency of the word and frequency of the related word.

$$frequency = \max\{frequency(term), frequency(related\_term)\} \quad (2)$$

Frequency of related words that are not used in the document is set to zero. Doing this preprocess over words of the document ensures that different documents that are written by different writers. Different writers may use different words to specify a single intention and by this preprocess similarity of such documents is increased.

In the next step, "tf" is calculated for documents in the training database by equation (3). Finally $\log(tf+1)$ is used as the extracted feature [14].

$$tf = \frac{1+\log(termfreq)}{1+\log(average\_termfreq)} \tag{3}$$

After counting words and calculating frequency of words in a document yet some more works should be done. Speed of a classifier is important in many applications. Here we use PCA to reduce time needed to measure similarity between documents. PCA transforms features into a new feature space and then eliminates features having low covariance. Large feature vectors make the classifier slow, so by reducing number of features using principle component analysis the speed is increased and only features that seem to be discriminated in the new feature space are preserved. This process, extremely decreases the size of index file.

Now indexed database is created and stored in the index file. In the next section, the classifier itself is described. Figure 1, shows the diagram of indexing process.

3 Classifier

For the classification purpose, we used K-NN because of its simplicity and efficiency. For the training phase, feature vector obtained from the previous phase is just stored in a file.

K-NN finds K nearest documents to the document d_j which is to be classified; then it counts number of positive and negative samples in those k documents. Based on number of positive and negative samples; the algorithm gives a score to the document. In this algorithm parameter K is critical. There is no way to predict K. So the best way is to initialize k with a random number and then change it to improve results. For similarity measurement we use the cosine distance as follows: [7]

$$Sim(\vec{d}_t,\vec{d}_j) = \frac{\vec{d}_t.\vec{d}_j}{\left|\vec{d}_t\right|\times\left|\vec{d}_j\right|} = \frac{\sum_{k=1}^{t} w_{ki}\times w_{kj}}{\sqrt{\sum_{k=1}^{t} w_{ki}^2}\times\sqrt{\sum_{k=1}^{t} w_{kj}^2}} \tag{4}$$

Where t denotes number of features extracted for documents.

To classify unknown document d_j, neighbors of this document are scored using equation (5):

$$Score(d_j,C_i) = \sum_{d_k\in KNN(d_j)} Sim(d_j,d_k)\delta(\vec{d}_k,C_i) \tag{5}$$

Where KNN(d) indicates K nearest neighbors of document d and $\delta(\vec{d}_k,C_i)$ stands for the classification for document d_j with respect to class C_i, that is:

$$\delta(\vec{d}_k, C_i) = \begin{cases} 1 & \vec{d}_j \in C_i \\ 0 & \vec{d}_j \notin C_i \end{cases} \tag{6}$$

So the decision rule of K-NN classifier can be specified as:

$$C = \arg\max_{C_i} (Score(d_j, C_i)) \tag{7}$$

4 Experimental Results

Many factors are used to evaluate classification but precision, recall, and F-measure are mostly used to evaluate classifications accuracy, exactness and completeness.

In a classification task, the Precision for a class is the number of items correctly labeled as belonging to the class (true positives) divided by the total number of elements labeled as belonging to the class (the sum of true positive and false positive). Recall in this context is defined as the number of items correctly labeled as belonging to the class divided by the total number of elements that actually belong to the class (the sum of true positive and false negative). F-measure is a weighted average of Precision and Recall and is used to measure performance in the field of information retrieval [15].

$$\text{Precision} = \frac{\#Pages\_correctly\_classified}{\#pages\_correctly\_classified + \#misclassified\_pages} \tag{8}$$

$$\text{Recall} = \frac{\#Pages\_correctly\_classified}{\#pages\_correctly\_classified + \#unclassified\_pages} \tag{9}$$

$$F - measure = \frac{2 * recall * \text{Precision}}{recall + precision} \tag{10}$$

To evaluate the algorithm, we randomly selected 2000 documents from Hamshahri corpus; 1600 documents of this set are used for training and 400 documents used as test data.

Hamshahri is one of the first online Persian newspapers in Iran that has been published for more than 20 years and it has presented its archive to the public through its website [16] since 1996. Creation of Hamshahri collection goes back to [17] in which Darrudi et al. employed a crawler to download available on-line news from the web site of Hamshahri newspaper and conducted a few experiments and presented some of the characteristics of the Persian language on this corpus [18].

Testing different values for K shows that the best value for K in our application is 10. By choosing K=10 for K-NN algorithm, the efficiency measurements of the proposed classifier are presented in table 1. The classifier correctly classified 336 documents, misclassified 45 documents and was not able to classify 19 documents.

The classifier implemented in Matlab, and FarsNet[2] used as the wordnet. FarsNet is developed by the National Language Processing Labratoary as the first Persian wordnet [19].

The implementation of K-NN algorithm on a Chinese corpus in [20] has an average precision of 0.6864, another implementation of this algorithm on Turkish language has an average precision of 0.8612 [21], and with fuzzy K-NN in [22] has average of 0.804.

The proposed classifier with a precision of 0.88; seems to be more efficient than other implementations of K-NN on Persian language. The classifier proposed in [1] has precision of 79.9%. The speed of K-NN increased because of the principle component analysis.

Table 1. Efficiency measurements of the proposed classifier

Precision	Recall	F-measure
88.18%	94.64%	91.29%

Figure 2 shows accuracy of the classifier for different values for K.

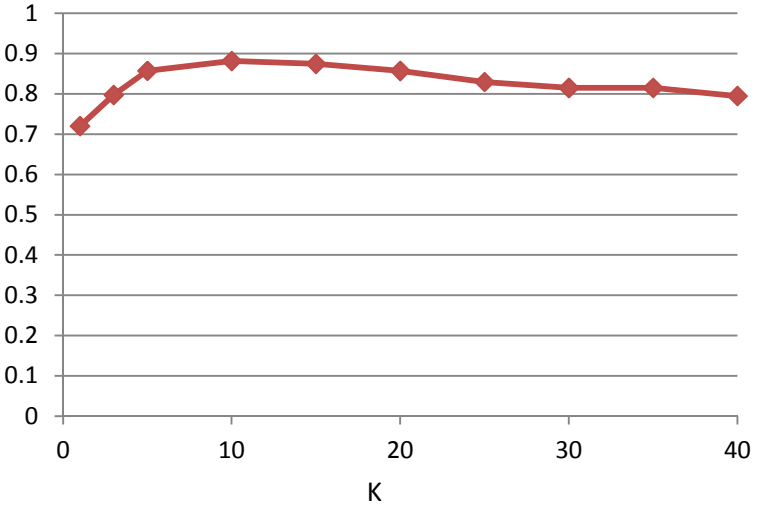


Fig. 2. Precision for different values of K

[2] http://nlp.sbu.ac.ir/site/

5 Conclusion

There are different methods for text classification, many factors are important in this field such as accuracy and efficiency. Document representation has a great impact on the accuracy of the classification and feature extraction is an important phase in text classification.

In the proposed method, accuracy is increased by 3 techniques. At first all words are replaced by their stems, information gain is used to eliminate words with high frequency and low information gain which means that word is not discriminated and finally wordnet is used to adapt terms frequency.

The efficiency of the proposed classifier is improved by use of PCA to decrease number of features. The similarity function works much faster with less number of features.

Finally applying the proposed method for indexing and classifying Persian documents shows a great improvement. Accuracy of the proposed classifier is 88.18%.

For the future works some experiments can be done to improve the classification accuracy, other classifiers like Support vector Machine can be applied, additionally other similarity functions can be used in the K-NN.

References

1. Keikha, M., Khonsari, A., Oroumchian, F.: Rich document representation and classification: An analysis. Knowledge-Based systems 22, 67–71 (2009)
2. Nather, P.: Text Categorization, Diploma thesis (2005)
3. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1–47 (2002)
4. Han, E.-H(S.), Karypis, G.: Centroid-Based Document Classification: Analysis and Experimental Results. In: Zighed, D.A., Komorowski, J., Żytkow, J.M. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 424–431. Springer, Heidelberg (2000)
5. van Mun, P.P.T.M.: Text Classification in information retrieval using winnow, `http://citeseer.ist.psu.edu/cs`
6. Aas, K., Eikvil, L.: Text ctegorisation: A survey, `http://citeseer.ist.psu.edu/aas99text.html`
7. Tan, S.: An effective refinement strategy for KNN text classifier. Expert Systems with Applications 30, 290–298 (2006)
8. Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., Wang, Z.: A novel feature selection algorithm for text categorization. Expert Systems with Applications 33, 1–5 (2007)
9. Fellbaum, C.: WordNet: An Electronic Lexical Database and Some of its Applications. MIT Press, Cambridge (1998)
10. Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Pasca, M., Soroa, A.: A Study on Similarity and RelatednessUsing Distributional andWordNet-based Approaches. In: NAACL 2009 Proceedings of Human Language Technologies: Annual Conference of the North American Chapter of the Association for Computational Linguistics (2009)
11. Sagot, B., Fiser, D.: Building a free French wordnet from multilingual sources. In: Proceedings of OntoLex (2008)

12. Keyvan, F., Borjian, H., Kasehff, M., Fellbaum, C.: Developing PersiaNet: The Persian Wordnet. In: 3rd Global Wordnet Conference (2007)
13. Rouhizadeh, M., Yarmohammadi, M.A., Shamsfard, M.: Developing the Persian WordNet of Verbs; Issues of Compound Verbs and Building the Editor, Resource Centre for Indian Language Technology Solutions (2009)
14. Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research 3, 1289–1305 (2003)
15. Obozinski, G.: Multi-class and Structured Classification. Practical Machine Learning CS 294 (2006)
16. Hamshahri Newspaper, http://www.hamshahri.net/
17. Darrudi, E., Oroumchian, F., Hejazi, M.R.: Assessment of a modern Persian corpus. In: Proceedings of the Second Workshop on Information Technology and Its Disciplines (WITID). ITRC, Iran (2004)
18. AleAhmad, A., Amiri, H., Darrudi, E., Rahgozar, M., Oroumchian, F.: Hamshahri: A Standard Persian Text Collection. Knowledge-Based Systems 22, 382–387 (2009)
19. Shamsfard, M., Hesabi, A., Fadaei, H., Mansoory, N., Famian, A., Bagherbeigi, S., Fekri, E., Monshizadeh, M., Assi, M.: Semi Automatic Development of FarsNet; The Persian WordNet. In: 5th Global WordNet Conference (GWA 2010), Mumbai, India (2010)
20. Li, B., Yu, S., Lu, Q.: Proceedings of the 20nd International Conference on Computer Processing of Oriental Languages (2003)
21. Ufuk, I.: M.S. Thesis, Report of Text Categorization (2001)
22. Basiri, M.E., Nemati, S., Ghasem Aghaei, N.: Comparosion of Persian text classifiers using kNN and fkNN algorithms and feature selection based on information gain and document frequency. In: 13th Conference of Computer Society of Iran, Sharif university of technology, Tehran (2007)

An Efficient Minimum Vocabulary Construction Algorithm for Language Modeling

Sina Lin[1], Zengchang Qin[1,3], Zehua Huang[1,2], and Tao Wan[4]

[1] Intelligent Computing and Machine Learning Lab
School of ASEE, Beihang University, Beijing, China
[2] School of Advanced Engineering, Beihang University, China
[3] Robotics Institute, Carnegie Mellon University, Pittsburgh, USA
[4] School of Medicine, Boston University, Boston, USA
zcqin@buaa.edu.cn

Abstract. In learning a new word by a dictionary, we first need to know a set of "basic words" which are frequently appeared in word definitions. It often happens that you cannot understand the word you looked up because there are still some words you do not understand in its definitions or explanations provided by the dictionary. You can keep looking up these new words recursively till they all can be well explained by some *basic words* you already knew. How to automatically find a minimum set of such basic words to define (or recursively define) the entire vocabulary in a given dictionary is what are going to discuss in this paper. We propose an efficient algorithm to construct the *Minimum Vocabulary* (MV) using the word frequency information. The minimum vocabulary can be used for language modeling and experimental results demonstrate the effectiveness of using the minimum vocabulary as features in text classification.

1 Introduction

The emergence of a complex language is one of the fundamental events of human evolution, some words are believed to be more complex than others because they present more precise semantic meanings that can be well explained by using some "basic words". In learning a new language, the dictionary is a powerful tool to learn a new complex word based on the words you already knew though some explanations are unavoidably reciprocal or circular, as "*hind, the female of the stag; stag, the male of the hind.*" When you are learning a new language, after you have grasped some basic words and start to use dictionary to learn yourself, it often happens that you cannot understand a word you just looked up because there are still some words you do not understand in its explanations provided by the dictionary. You can keep looking up these words till they all can be well explained by the basic words you knew. However, if you find yourself stuck in a recursive process of keeping looking up different new words or in a reciprocal process like the above *hind-stag* example, it means you may need a bigger set of "basic words". A simple question arose, *what are these basic words in a given dictionary?*

It's been a long standing question to find whether such a set of basic words exists that can be used to define the entire vocabulary in this fashion. This is called the *Minimum Vocabulary Problem* (MVP) [3]. MVP aims to find a minimum vocabulary set which

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 292–301, 2012.

can define (or recursively define) the entire vocabulary in one dictionary. MVP has been widely studied in literature, many of which focused on suggesting appropriate size of vocabularies for different level language learner [5,6]. These works are mainly based on manual analysis and corpus statistics, which are less informative in language modeling. For example, general service list [13] selected 2000 basic words for new English learners. Research for using an automatic algorithm to solve MVP could help us to uncover the intrinsic structure of human language by finding these basic "semantic bricks" for building a language. Fig. 1 gives a schematic illustration how a word "haphazard" can be explained by using other words in its definition from the Collins dictionary. This tree will be extended till all the leaf nodes are words in the minimum vocabulary (MV).

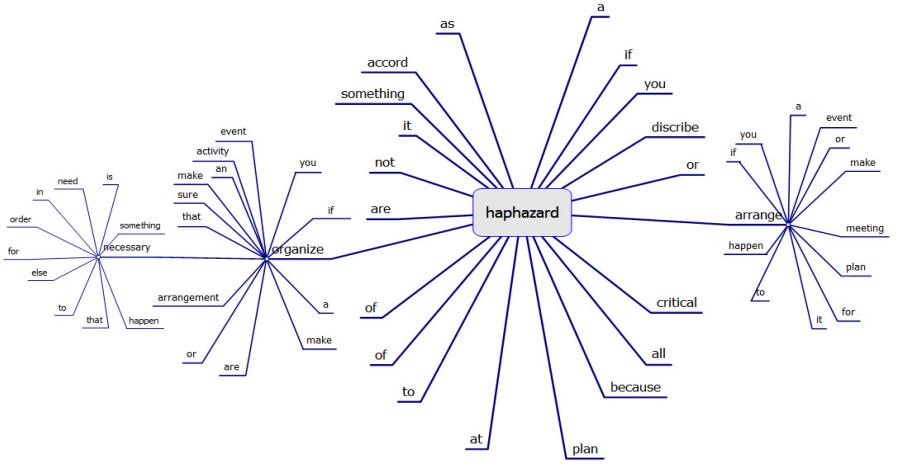


Fig. 1. An example of word dependency. All the leaf node words are belonging to the minimum vocabulary, they are used to explain a complex word "haphazard".

A computation solution of this problem is first proposed in [3], in which a graph model is used to reformulate the problem as calculating the basis of a definition graph and achieves fast computation using approximation. But this method requires much work on pre-processing and reduces the applicability. In this paper, we only focus on English language though the same methodology can be applied to many other alphabetic languages such as German, French and Spanish. We propose a new model by adopting the methodology of relational database design [11]. By transforming the whole dictionary into a functional space, we can model word-definition relations as a functional dependency of graphs. The problem is then simplified to seek the minimum closure of a set that shares the common strategy with calculating candidate key of a relational database [11]. However, if taking account of the enormous size of a dictionary, traditional methods would fail because of computational complexity. In this paper, we propose a fast approximation by combining each word with the prior knowledge of the word frequency and iteratively calculating the basic word set to construct the minimum vocabulary. Based on experimental studies, this method makes a good balance on both performance and computation efficiency.

This paper is organized as follows. In Section 2 we describe the formulation of the problem and a new algorithm is proposed. In Section 3, experimental results of three well known dictionaries are presented. We also introduce the concept of using the MV as features descriptors in language modeling and apply the model in text classification. Finally, the conclusions are given in Section 4.

2 Minimum Vocabulary Model

We use $\mathcal{D}$ to denote a given dictionary and $w \in \mathcal{D}$ is the word in $\mathcal{D}$. $\mathcal{E}(w)$ stands for the set of words in the definition (explanation) of w. We also define $w \in \mathcal{E}(w)$ for mathematical consistence because it is true that word can be explained by itself. The main idea is to set up a directional relation from w to its explanations $\mathcal{E}(w)$. Given the nature of this problem, mathematics of relational database can be used. For given two sets of words $S_i, S_j \subseteq \mathcal{D}$, we propose a dependency property developed from the similar concept in relational database.

Definition 1. *Given two sets of words S_i, $S_j \subseteq \mathcal{D}$, S_i depends on S_j, or S_j determines S_i (denoted by $S_j \Rightarrow S_i$) when:*

$$\forall w \in S_i : \ \mathcal{E}(w) \subseteq S_j$$

In other words, if we knew all the words in S_j, we can also know all the words in S_i because all the words in S_i can be explained by words in S_j. Any set S depends on itself based on the above definition, i.e.: $S \Rightarrow S$.

The following properties [1] hold for the dependency relation:

1. **Reflexivity:** If Y is a subset of X ($Y \subseteq X$), then $X \Rightarrow Y$.
2. **Augmentation:** If $X \Rightarrow Y$, then $X \cup Z \Rightarrow Y \cup Z$, for any word set Z.
3. **Transitivity:** If $X \Rightarrow Y$ and $Y \Rightarrow Z$, then $X \Rightarrow Z$.

These properties can be easily proved using the definition of relational dependency. Therefore, an unknown word set can be inferred from a set of basic words through definition relations, this can be well explained through the empirical experience that a new (complex) word could be learnt from a certain amount of very simple words.

Definition 2. *The closure of S, or S^+, is the set of words that can be determined by S, or $S \Rightarrow S^+$ where:*

$$S^+ = \{x | w \in S, x \in \mathcal{E}(w)\}$$

The solution of MVP is about to find a minimum set of S that $S^+ = \mathcal{D}$.

2.1 MV Construction Algorithm

Previous research in relational database design provides a few efficient solutions for calculating the minimum set of all relational attributes which is referred as *candidate key*. Saiedian and Spencer [12] proposed a graph method to extract candidate key with

Algorithm 1. Minimum Vocabulary Construction Outline

```
1: procedure MINIMUM VOCABULARY(𝒟)
2:     S = 𝒟
3:     repeat
4:         for all w_r ∈ S do
5:             if w_r ∈ {S − ∑_r w_r}^+ then
6:                 S ← {S − w_r}
7:             end if
8:         end for
9:     until No more word w_r can be removed from S
10:    return S
11: end procedure
```

time complexity of $O(kn^2)$, where n is the number of items and k is the number of dependencies between these items. Since they focused only on database design, when the size of database is small, the problem is tractable. However, for our problem, a fair dictionary commonly has over 30000 words and millions of dependency relations, their method is computationally inefficient.

In out approach, the main idea for MV construction is simple. We start from a set S assigned with $\mathcal{D}$ ($S = \mathcal{D}$), then we iteratively remove redundancy word w_r that can be explained (or recursively explained) by the rest of words denoted by the set $\{S - \sum_r w_r\}$. The removing word satisfies that $w_r \in \{S - \sum_r w_r\}^+$. The algorithm terminates when no more word can be removed from S to satisfy the above conditions. The pseudo-code is shown in Algorithm 1. However, it hasn't consider the key factors such as the removing order and the closure computation, both of which are curial for fast computation and effective performance, that will be discussed in the next section.

2.2 Familiarity and Frequency of Words

As we can see from the previous section, the key problem is to decide a preference of removing order of the words. Based on empirical knowledge of language, people tend to use words they are more familiar with to explain those are not, we therefore need to choose the attribute implying "familiarity" to evaluate word preference. There is a rich literature in human perception of "familiarity" and some insightful discussions on this topic are available in [2,8]. Here we adopt the simplest familiarity measure in terms of word frequency.

In this research, frequency statistics on BNC database [7] is used to assign each word with an attribute of frequency. This attribute defines the preference of removing redundant word in applying Algorithm 1. In details, we sort the removing words sequentially based on word frequency in ascending order. The lower frequency a word has, the higher possibility to be removed from the basic word set. $R[w]$ is used denote the frequency of word w.

The closure computation for a large set has a high computational complexity. Therefore, in order to propose a reliable solution for the MVP, we propose an efficient algorithm for closure computation. Line 5 in Algorithm 1 of closure computation can be

modified from calculating the entire set closure to measuring whether a word can be explained by a set of high frequency words. We present details of the proposed method on calculating in Algorithm 2.

For each pair $(w, R[w])$, we calculate the max preference word in its definition word set $\mathcal{E}(w)$. We define the word with maximum frequency in $\mathcal{E}(w)$ is

$$mf(w) = \max_{i \in \mathcal{E}(w)} R[i] \tag{1}$$

If $mf(w) < R[w]$, it means w can be explained by words with higher preference. Then $R[w]$ can be replaced with $mf(w)$. If not, it means w temporarily can not be removed from the word set, then w becomes the candidate of basic word. For each iteration of scanning overall words, we get all candidate basic words $\mathcal{S}$. Since all words can be explained by $\mathcal{S}$, we have $\mathcal{S}^+ = \mathcal{D}$. Note preferences of all words are updated throughout each iteration, some words might change their preferences. We need reprocess all words until the full coverage of all words. The algorithm runs in $O(kn)$, where n is the size of dictionary and k is the number of process iterations. The pseudo-code is given in Algorithm 2.

Algorithm 2. Word Frequency Based Minimum Vocabulary Construction

```
    procedure MINIMUM VOCABULARY(D)
 2:     Sort w ∈ D based on word frequency R[w]
        for w ∈ D do
 4:         RemovingTag[w] ← False
        end for
 6:     repeat
            for all w ∈ D do
 8:             if mf(w) < R[w] then
                    R[w] ← mf(w)
10:                 RemovingTag[w] ← True
                end if
12:         end for
        until convergence
14:     for all w ∈ D do
            if RemovingTag[w] == False then
16:             add w to S
            end if
18:     end for
        return S
20: end procedure
```

3 Experimental Studies

Since the model is heavily based on the word dependency relationship. For the same word, its definition may not be identical in different dictionaries. In our experiments, we tested 3 well known dictionaries using the new proposed algorithm: *Collins,*

Webster and *Longman* [1]. In these dictionaries, each phrase is consisted of phonetic symbols, several definitions and example sentences. In the following experiments we only consider the word itself and the first of its definitions, other semantic information is ignored at this stage . We lemmatize each word using Python NLP Toolkit [9]. Word frequency introduced in Section 2.2 is computed throughout word frequency list [2] based on BNC Corpus [7]. We perform the same lemmatization and assign each word in dictionary with a frequency $R[w]$.

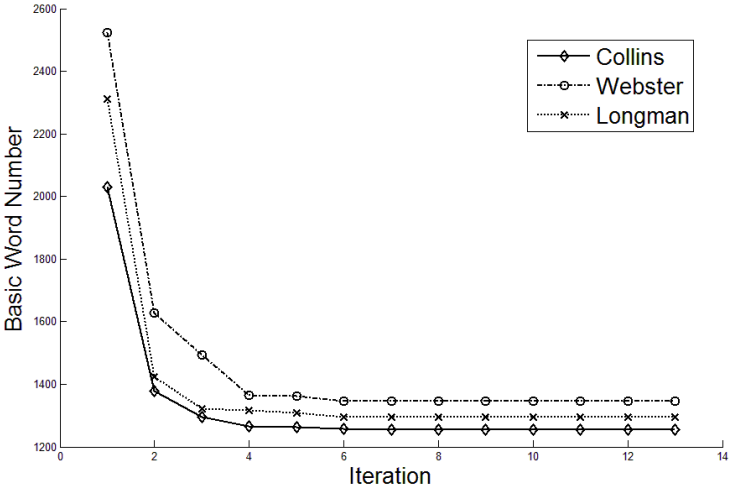


Fig. 2. Minimum vocabulary for Collins, Longmand and Merrian-Webster based on the frequency-based minimum vocabulary construction algorithm

3.1 Minimum Vocabulary of Dictionaries

Each dictionary has average 30000 words, Algorithm takes about 1 minute to finish the computation. The sizes of MV for Collins, Longman and Merrian-Webster dictionary are 1256, 1295 and 1346, respectively. Fig. 2 illustrates the number of words in the MV with the increasing number of iterations. The results show that the size of MV converges very fast and becomes stable after about 5 iterations. For the 3 given dictionaries, the sizes of MV are very similar.

Table 1 shows the results of overlapping of MV in 3 dictionaries. The common basic words appear in all dictionaries are 672, taking up about 50% in each dictionary's basic word set. This can be explained by that dictionary use simple and high preference (frequency) words in word definitions. Therefore, most words appear in definition would be among the small set of high frequency words and the MVs for different dictionaries have a high overlapping.

[1] The source of these dictionaries can be obtained for free from the following links: `http://debian.ustc.edu.cn/debian-uo/dists/sid/ustc/pool/stardict/`

[2] Available at the link: `http://www.kilgarriff.co.uk/bnc-readme.html`

Table 1. The number of common words in the MVs and the percentages of them in the three given dictionaries: Collins, Longman and Webster

Combination	Number	Percentage of Common Words in Dictionary
Collins ∩ Longman	824	Collins: 66% Longman: 64%
Collins ∩ Webster	861	Collins: 69% Webster: 64%
Longman ∩ Webster	957	Longman: 74% Webster: 72%
Collins ∩ Longman ∩ Webster	672	Collins:54% Longman: 51% Webster: 50%

3.2 Minimum Vocabulary Properties

In the framework of using MV for language analysis. A word may be explained in several layers like a tree, where all the leaf nodes are the words in the MV (e.g., see Fig. 1). The maximum level of the MV interpretation of a particular word may have some implications on their semantic complexities. More layers a word has, more semantically difficult this word is. We summarize the statistics of word levels across the whole dictionary. The left-hand figure of Fig. 3 shows the histogram of the word levels. The right-hand figure illustrates the accumulated percentage of words under the given word level. For example, 80% of words are under the level 10 and nearly 90% of words are under the level 15.

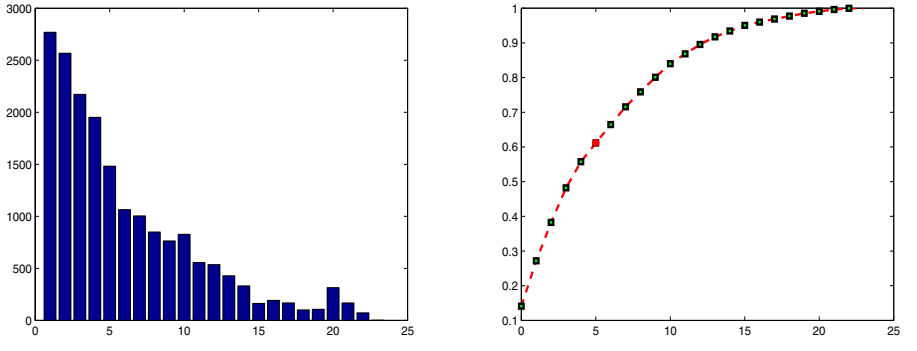


Fig. 3. Left-hand: histogram of word levels. Right-hand: accumulative percentage of words under the given word level.

In order to validate effectiveness of using the MV as the language model, we analyze the similarities between MVs of synonym pairs comparing to the similarities between non-synonym pairs. The similarity measure between to MVs can be defined by:

Definition 3. *Given two sets of words W_1 and W_2 , the relative similarity degree is defined by the ratio between the intersection of W_1 and W_2 and the union of these two sets.*

$$sim(W_1, W_2) = \frac{|W_1 \cap W_2|}{|W_1 \cup W_2|} \tag{2}$$

where $|\cdot|$ represents the cardinality of a set.

We collect a set of synonym pairs from the Oxford Thesaurus [3]. Phrases and stop words are removed. Considering that our focus are complex words, we also removed the synonyms in the given MV. In our experiment, we take two pairs of synonyms where A and B is one pair and C and D is another pair, i.e.:

$$A \leftrightarrow B,\ C \leftrightarrow D \tag{3}$$

We then calculate similarities between each pair of words, the following relations should hold:

$$sim(A, B) \geq sim(A, C)$$

$$sim(C, D) \geq sim(B, C)$$

We calculated 1650 pairs of synonyms and obtained 75.2% of which satisfy the above relations. This experiment can verify that MV based measure can reflect certain semantic relations with good confidence. In the next section, we will use MV as language features in text classification.

3.3 Document Feature Descriptor

Language modeling is an important topic in computational linguistics, techniques such as latent semantics indexing, topic models were used to map a text into a low dimension semantic space [14]. In such a space, different natural language problems can be studied by capturing the semantic meaning of the original text, e.g., question answering [10]. In this study, the MV also can provide the semantic relations between complex words and a text can be modeled in the MV space, that is how the MV can be used as descriptors for natural language modeling.

The MV with approximately 1300 words are obtained based on proposed algorithm. Words in the MV form the basic structure of a language. The meaning of the MV is highly compressed and may help to uncover intrinsic relations between words. For example, some cognates and synonyms may derive from same basic word ancestors. This property offers basic words a potential usage for a sound document descriptor. To demonstrate this, we employ the MV model to construct a document feature descriptor and apply it to text mining.

We denote $\mathcal{S}(w)$ as the basic words that w depends on, and $F(w_i)$ is the frequency of the ith basic word in $\mathcal{S}(w)$. The major purpose of this feature descriptor is to represent document using basic word histogram. To apply this, we first replace all the words by using basic words and calculate the basic histogram H_M.

We evaluate the MV descriptor on TechTC-100 Test Collection [4]. We use preprocessed feature vectors provided from this dataset, in which texts were simply tokenized and digitalized, no further processing was employed such as TF-IDF and lemmatization. We retrieve the feature descriptor by computing H_M based on the Minimum Vocabulary set of Collins Dictionary and obtain a 1253 dimension vector to represent each document. We compare our descriptor with original feature vector provided from the dataset. The classification tasks is performed between two classes of documents. We

[3] `http://thesaurus.com/browse/Oxford`

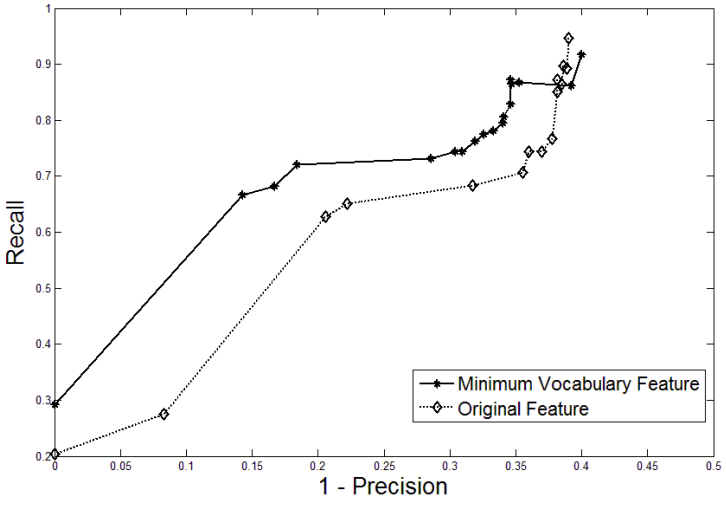


Fig. 4. ROC curve of classification by using MVs as text features

select linear SVM as the classifier. The recall and precision results is shown in Fig. 4. As we can see from the figure, the MV descriptor outperforms the original word feature significantly. The time consumption is also benefited from the downsize of feature vector, it reduces from 3.745s to 1.042s on a Dual Core 500MHZ machine.

4 Conclusions

This paper proposed an efficient computation method for the Minimum Vocabulary Problem. We proposed a new algorithm to construct the MV for a dictionary in order to investigate the word-explanation relationship by employing the word frequency regularizer. The empirical studies on three well known dictionaries are given. We also studied the properties of MV and use it in language modeling. The MV can be considered as the most basic "semantic bricks" for a language. Some initial investigations with experimental results of using MV features in text classification are given. Future works will study how to use the MV to solve other natural language processing problems.

Acknowledgment. This work is partially funded by the NCET Program of MOE, China and the SRF for ROCS. The second author also thanks the China Scholar Council for visiting fellowship (No. 2010307502) to CMU.

References

1. Armstrong, W.W.: Dependency structures of data base relationships. Information Processing 74, 580–583 (1974)
2. Cancho, R.F.I., Solé, R.V.: Least effort and the origins of scaling in human language. Proceedings of the National Academy of Science 100, 788–791 (2003)

3. Chandrasekharan, N., Sridhar, R., Iyengar, S.: On the minimum vocabulary problem. Journal of the American Society for Information Science 38(4), 234–238 (1987)
4. Gabrilovich, E., Markovitch: Text categorization with many redundant features: Using aggressive feature selection to make SVMs competitive with C4.5. In: ICML (2004)
5. Hazenberg, S., Hulstun, J.H.: Defining a minimal receptive second-language vocabulary for non- native university students: An empirical investigation. Applied Linguistics 17(2), 145–163 (1996)
6. Hirsh, D., Nation, P.: What vocabulary size is needed to read unsimplified texts for pleasure? Reading in a Foreign Language 8, 689–696 (1992)
7. Kilgarriff, A.: Putting frequencies in the dictionary. International Journal of Lexicography 10(2), 135–155 (1997)
8. Kilgarriff, A.: Using word frequency lists to measure corpus homogeneity and similarity between corpora. In: ACL-SIGDAT Workshop on Very Large Corpora, pp. 231-245 (1997)
9. Loper, E., Bird, S.: NLTK: The natural language toolkit. In: Proceedings of the ACL 2002 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics, vol. 1, pp. 63–70 (2002)
10. Qin, Z., Thint, M., Huang, Z.: Ranking Answers by Hierarchical Topic Models. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS (LNAI), vol. 5579, pp. 103–112. Springer, Heidelberg (2009)
11. Ramakrishnan, R., Gehrke, J.: Database Management Systems. McGraw-Hill, Inc. (1999)
12. Saiedian, H., Spencer, T.: An efficient algorithm to compute the candidate keys of a relational database schema. The Computer Journal 39(2), 124–132 (1996)
13. West, M.P.: A General Service List of English Words. Longman (1976)
14. Zhao, Q., Qin, Z., Wan, T.: What Is the Basic Semantic Unit of Chinese Language? A Computational Approach Based on Topic Models. In: Kanazawa, M., Kornai, A., Kracht, M., Seki, H. (eds.) MOL 12. LNCS (LNAI), vol. 6878, pp. 143–157. Springer, Heidelberg (2011)

Effective Co-reference Resolution in Clinical Text

David Hinote[1], Carlos Ramirez[2], and Ping Chen[1]

[1] University of Houston - Downtown, Houston, TX
hinoted5@gator.uhd.edu, chenp@uhd.edu
[2] University of Puerto Rico, Río Piedra, PR
carlosfabianramirez@gmail.com

Abstract. The 2011 I2B2 challenge involves co-reference resolution in medical documents. Concept mentions have been annotated in clinical texts, and the mentions which co-refer in each document are to be linked by co-reference chains. There have been systems developed for co-reference resolution by various organizations. The aim of this study was to use the systems which are publicly available, as well as build a rule based algorithm tailored for this challenge, and test these systems on the data provided for this challenge. The study shows the publically available systems do manage to find some of the co-referent links, and the rule based system developed for this challenge performs well finding the majority of the co-referent links. The system that was used to provide the final outputs for the challenge had 89.6% overall performance average.

1 Introduction

Co-reference resolution is the process of linking together concepts which refer to the same entity. The ability to have computers automatically find this type of relation in text documents is of interest to people in the field of artificial intelligence because it can lead to having systems which can summarize texts and answer questions posed about information contained within those documents [1-2]. Automatic summaries and question answering systems could be of great value to personnel in the healthcare industry as well. Because of these possibilities, a challenge was hosted in 2011 by i2b2 [3] (Informatics for Integrating Biology & the Bedside) in order to advance co-reference technology for the field of medicine. Annotated data has been provided by four institutions: Partners HealthCare, Beth Israel Deaconess Medical Center, The University of Pittsburgh, and the Mayo Clinic. This data includes the original texts for each document, a concepts file for each document which describes each concept mention in the texts, and chain files which identify manually created co-reference chains in each of the texts as an example of how chains are to look after processing. The concept mentions to be linked are nouns or descriptive phrases in the medical texts which represent people, actions, objects, or ideas and have been given types accordingly. There were two methods used by the hosts of the challenge to annotate the data sets used in the I2B2 shared task. One method is the I2B2 style annotations which include 5 concept categories: people, problems, tests, treatments, and pronouns. The other method used is ODIE (Ontology Development and Information Extraction)

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 302–312, 2012.

style annotations which include 8 categories: disease or syndrome, sign or symptom, procedure, people, other, none, laboratory or test result, and anatomical site. Each type of concept mention will only co-refer with a concept mention of the same type, with the exception of pronouns which can co-refer with any type of mention [4]. In this paper, we briefly describe the three publicly available systems and their performance on the i2B2 data, the construction and performance of the rule based system, and the results of the our participation in the i2b2 challenge.

2 Evaluation Methods

Each system was evaluated in two ways. The first method was to compare each link with the provided co-reference chain annotations, and count it as correct only if it matches exactly with the provided annotation. With this method, single unlinked concept mentions are not considered, and links that fall in the same chain but skip an antecedent are considered incorrect. This type of scoring is referred to as exact match scoring in this paper. The second method of evaluation is with a script provided by I2B2, which conducts 4 types of examinations of the chain output for each system: B-Cubed [5], MUC [6], Blanc [7], and CEAF [8]. Overall performance results using both methods are listed in this paper after each system description. The results are in the form of a F1 score which is the harmonic mean of precision and recall [9].

3 Other Systems Used in the Study

There are a number of systems publicly available for co-reference resolution that can be found by conducting internet searches on most popular search engines. These systems will discover co-referent links and chains from raw text input. To do this the software uses internal functions to find concepts, and then link them [10-12]. The three systems used for testing were chosen for their level of development and usability. In this study BART [10], the Stanford co-reference system [11], and LingPipe [12] were tested on the provided training data.

3.1 Bart

BART is an acronym for Beautiful Anaphora Resolution Toolkit and it was developed from a project done at the 2007 Johns Hopkins Summer workshop [10]. It is available on the website created for the project: http://www.bart-coref.org/. Once set up, text is sent to it through a web service, and output is returned in XML format. The output contains detected concept mentions and if they belong to a chain, the chain identifier is included in the XML tag of the concept mention. A translator was created to compare the BART output to the chain files included with the input texts. Only concept mentions detected by the BART system and listed by the I2B2 annotations were considered for testing. All other mentions and co-referent links were discarded. After running the BART system on files from each of the data sets provided, it had an overall F1 Score performance of .775 on the Beth Israel training data, .712 on the Partners Healthcare training data, and .436 on the Mayo Clinic training data. All three

of these scores were obtained from the evaluation scripts provided by I2B2. Individual concept type linking scores using the exact match scoring are listed in the "Results" section on Table 1.

3.2 The Stanford Co-reference System

The Stanford co-reference system is an ongoing project by the Stanford Natural Processing Language Group [11]. It can be found at the Stanford Natural Language Processing Group's website: http://nlp.stanford.edu/software/dcoref.shtml. It uses what is called a "Multi-pass sieve" to perform co-reference resolution, which is a layered approach to detecting links between mentions. It starts with the strongest match first then uses more and more relaxed criteria for matches as it runs down the layers of co-referring rules [11]. Like BART, it uses its own internal functions to identify concept mentions. Input and output for this system was done by calling the Java classes directly from the computational program developed for this study. Input was supplying the raw text in a string, and output from this system comes in the form of a map stored in an array. Each element of the array holds the location, in the form of line number and word number in the text, of a source mention, and a destination mention. A simple mapping function was constructed to convert the Stanford concept locations to I2B2 concept locations. Only concept mentions that were found by the Stanford system and listed by the I2B2 annotations were considered, all other mentions and co-referent links were discarded. After running the Stanford system on files from each of the data sets provided, it had an overall F1 Score performance of .627 on the Beth Israel training data, .633 on the Partners Healthcare training data, and .423 on the Mayo Clinic training data. All three of these scores were obtained from the evaluation scripts provided by I2B2. Individual concept type linking scores using the exact match scoring are listed in the "Results" section on Table 1.

3.3 LingPipe

LingPipe is a suite of natural language processing tools provided by the Alias-i company as a commercial NLP product. It is available at no cost for research purposes at the Alias-i website: http://alias-i.com/lingpipe. LingPipe performs Co-reference resolution through a set of heuristic algorithms which link together mentions found by internal functions [12]. Input for the system was through command line functions specifying the location of the input text documents, and output was a text document containing xml tags surrounding discovered concept mentions and a chain identifier if the mention was found to be co-referent. A translator similar to the one used to map the BART system output was constructed to make the data useable in this study. After filtering out concept mentions not annotated in the I2B2 data, it had an overall F1 Score performance of .628 on the Beth Israel training data, .601 on the Partners Healthcare training data, and .423 on the Mayo Clinic training data. All three of these scores were obtained from the evaluation scripts provided by I2B2. Individual concept type linking scores using the exact match scoring are listed in the "Results" section on Table 1.

4 Our Rule Based System

Because the specifications for co-reference for the i2b2 challenge were well defined, and the type of data provided is specific kind of document [4], the type of system built for our study is rule based. Our method is designed only to find co-reference chains in text documents when the concepts are also given as input. When processing the data sets provided by I2B2, the gold standard concept files that came with the data were used to mark the concepts in the text documents. The system was developed by examining a sample of files from the pool of data and writing linking functions, or rules, based on observation. The linking functions were checked across the entire data set to get an idea of which rules worked, and which did not. The system consists of six text processing routines, and uses four data sources to aid in creating co-referent links. The general architecture for our final system is depicted in the chart below (Figure 1).

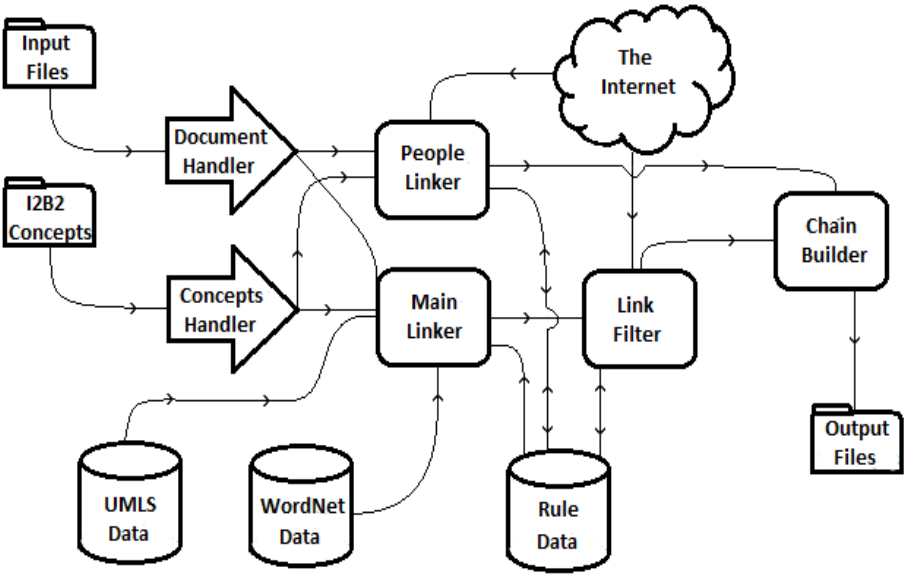


Fig. 1. Data flow chart for the rule based system

4.1 Data Input and Access

The first two routines in the system are made to read in the text being examined and the concepts that are to be linked from the files provided by I2B2. The document handler breaks the text into tokens using white space boundaries, with each space character indicating the end of one word and the beginning of the next. The text is then stored in a two dimensional array where the first dimension is the line number, and the second dimension is the word number. A representation of this operation is depicted below in figure 2.

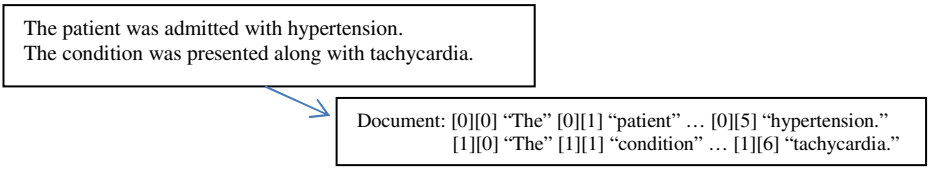


Fig. 2. Representation of document handler functionality

The document handler controls access to this matrix and gives the system a way to easily find the location of the concepts in the text, and a way to search the words surrounding the concepts for information about the concept. The concept handler reads in each concept and stores it in an array giving each concept a number based on its position in the array. Each element in the array holds the start line, start word, end line, end word, type, and the text within each concept. The concept handler is made to give easy access to the attributes of each concept. An example of concept storage can be found below in figure 3.

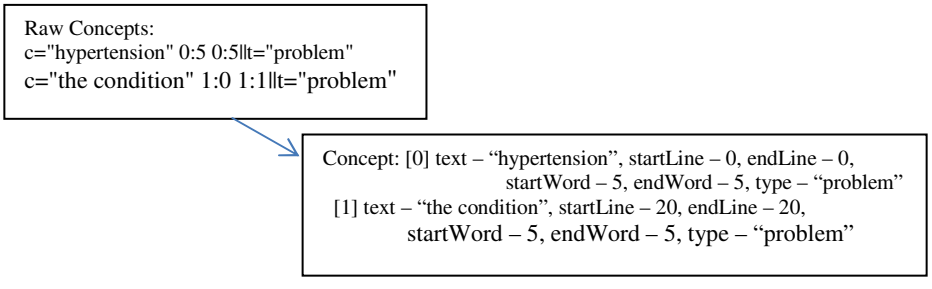


Fig. 3. Representation of concept handler functionality

4.2 Main Linker

The next routine in the algorithm is the main linker, and it is made to match all the concepts which are not in the person category. Every concept that passes through this linker is compared to each of the other concepts of the same type in the document and links are recorded if they meet the programmed criteria. Decisions made by this linker are binary meaning they either match or do not match. At this stage, every link that is detected is kept, which means a concept can have links to many concepts within the document, rather than at most two which is a characteristic of co-reference chains. The main linker uses string matching, the UMLS [13] (Unified Medical Language System) database, and the WordNet [14] database to determine if two concepts might have the same meaning. The main linker traverses the concept list and runs each one through its set of rules, and stores detected links in a list of pairs that is organized later on in the chain builder.

Non-personal Pronoun Match. The first step with each concept is to check if it is a pronoun type. If it is a pronoun type concept and the word is "which" or "that," it is linked to the concept that immediately precedes it if the two concepts have less than two words between the two concepts. There are other pronouns mentioned, but any

rules written for them only resulted in performance loss, we were unable to build a reliable rule for any other pronoun.

Be Phrase Match. The next step with each concept is to check the type of the concepts which immediately precede and follow the concept. If they are of the same type, the text in between the two concepts is examined and if it contains any words that indicate it is a "be phrase," the two concepts are linked because they are probably saying "something is something." Words and phrases that are commonly found in the "be phrases" are stored in the rule database, and were added to the database manually based on observations of gold standard links.

Match by Meaning. After the "be phrase" match, the concepts are examined and linked by their meanings. First, the concepts are conditioned by filtering out what we refer to as "common words." These common words include conjunctions (and, or, as, but, etc.), adjectives (large, blue, painful, etc.), and pronouns (he, she, it, etc.). The conjunctions and pronouns which are filtered out are chosen to be eliminated from the concept if they appear in the common words table of the rule database. Each of the words that appear in the common words table was manually placed there. Adjectives are detected by searches in the WordNet database. After elimination of the common words, any non-letter characters, such as punctuation and hyphens, are removed. After this conditioning, the concepts are compared to every other concept of the same type on the document in three ways.

Head and Synonym Match. First, every leftover word in the concept is compared to every leftover word in each of the other concepts of the same type in the document by a word comparison method. This word comparison method will declare the words a match if the first 80% of the characters in the shorter word match the same number of characters in the longer, or if they are found to be WordNet synonyms. If every word in one of the concepts is matched to a word in the other concept, a link between the two is recorded.

UMLS Match. The second comparison is through the UMLS database. Both concepts are searched for in the MRCONSO table of the UMLS database after the conditioning, and if they are found in the database and their UMLS concept numbers match, a link between the two is recorded.

Acronym Match. The third type of comparison is a check for acronyms. The first letters of each word in concepts that have two or more words are taken and are compared to whole words in other concepts, and if a whole word is found that matches either all the first letters, or some of them in order, a link is recorded.

4.3 People Linker

All concept types are processed though the same path in the algorithm except for the mentions of type "person" or "people". These mentions are processed by the people linker. As with the main linker, all decisions made by this linker are binary.

Identifying People Mentions. When the people linker is called to examine a document, it runs through several subroutines to identify "person" type mentions as being doctors or the subject of the document.

Medical Personnel. The first step performs internet searches on each concept mention. The mention being processed is sent to a search engine if it meets certain criteria, and the results are scanned for certain key words to indicate if the mention is referring to a doctor or medical personnel. Every mention that is found to be of medical personnel is stored in a list for later use.

The Subject (Patient). The second step is to find a name in the document to represent the subject of the document. The function checks each concept and if it meets these criteria:

- It is not a pronoun.
- It is not found to be a doctor according to the previous check.
- It does not have the doctor salutation, Dr.
- It has no medical title at the end, M.D.
- It does not contain common words stored in the rule database such as "patient" or words that would indicate it is a family member.

That concept is marked as the subject of the document. If no such concepts that fit that criteria are found, the first occurrence of a concept that says "patient" or "pt" is marked as the subject since the patient has been the subject of the document in every document. After finding an appropriate representation of the subject, every concept that has the words "patient" or "pt" in them and no words that refer to a family member are linked to the subject concept.

The Subject's Gender. The third step is to find the gender of the subject, and this function simply counts the number of masculine and feminine pronouns in the document and the type that is more frequent is declared to be the gender of the subject.

Matching People Mentions. After gathering information about the "person" and "people" type concept mentions, the algorithm move on to actually create links between these mentions.

Introduction Match. If two concepts are found to be no more than 2 words apart with one starting with a doctor salutation, or ending with a medical title, and the other was marked as referring to a doctor by the first internet searches, the two concepts are linked as this likely indicates an introduction of someone. An example of this kind of match is: "Dr. Smith, your optometrist" where "Dr. Smith" and "your optometrist" are two separate mentions.

Head Match. After linking the introductions, a matching function is run that works the same way as the head matching function in the main linker. Certain words are removed from concepts, such as salutations, pronouns, titles, single letters, as well as punctuation, then, they are compared to each other. If all of the words, up to 80% of the length of the word, in each concept appear in the other concept, a link between them is recorded. This match will link people's names together, including those that appear with an initial for the first name in one instance and the full name in another.

Pronoun Linking. The next step in the people linker is to match third person pronouns to the names to which they refer. This is done by searching the sentence in which pronoun concepts are contained.

Third Person No Proper Names in the Sentence. If the sentence has only pronoun mentions in it, each of the pronouns in that sentence are linked to the subject concept if they are of the same gender as the subject. If it is not the same gender as the subject, the closest preceding concept that is not a pronoun is linked to it.

Third Person With Proper Names in the Sentence. If there is one name in the sentence, and the name's position in the sentence is before the pronoun, then it is linked to that name. If there are multiple names in the sentence, any pronoun that is the gender of the subject is linked to the subject and the others are linked to the first name in the sentence that is found to be a doctor.

Other pronouns including First and Second Person Pronouns. After this, any person concepts that are first person pronouns are linked together, and any second person pronouns are linked to the subject. The last step is to link any pronoun type mention which is the word "this" to the next person mention if it is within 3 words of it and the next mention is not a doctor, then, any pronoun type mention which is the word "who" is linked to the previous person type mention that is not any type of pronoun.

4.4 Link Filtering

After the semantic links are made in the main linker, they are passed over to filters to eliminate links that actually refer to two different entities based on clues found in the sentences surrounding the mentions in question. These clues include descriptive phrases such as dates, locations, or descriptive modifiers not included in the span of the mention. These clues are found by using regular expressions for dates and key words stored in the rule database for locations and descriptive modifiers compared by string matching. These clues are only searched for if the word preceding or following each mention is one of the key words stored in the rule database. Examples include in, on, are, is, etc. The filter portion of the algorithm also eliminates links using WordNet, any mention that is found to be an adjective with no noun included has any links to it removed.

4.5 Building the Chains

Once the linkers and the filter have finished their jobs, the final output is created from the "web" of links that has been made. The first concept with links is found and each link is traversed to the next concept, and each of those links is followed in a recursive fashion. A list of each concept visited is kept, and though concepts can be linked more than one time, they are added only once to the list. After every link has been examined in the "web," the list of concepts is sorted according to each concepts position in the text. Concepts that appear in the beginning of the text are at the top of the list. Once a chain is constructed, it is written to an output file in the I2B2 format.

4.6 Rule Based Algorithm Performance

After running the algorithm on files from each of the data sets provided, it had an overall F1 Score performance of .891 on the Beth Israel training data, .912 on the Partners Healthcare training data, and .789 on the Mayo Clinic training data. All three of these scores were obtained from the evaluation scripts provided by I2B2. Individual concept type linking scores using the exact match scoring are listed in the "Results" section on Table 1.

4.7 Availability

The software written for this challenge will be available for use by anyone, and can be found at http://cms.uhd.edu/faculty/chenp/class/4319/project/.

5 Results

After running each of the systems on the provided data, the following exact match performance data described in "Evaluation Methods" was obtained from the results (Table 1).

Table 1. Exact match F1 scores for the four systems on individual concept mention types in the Beth Israel, Partners Healthcare, and Mayo Clinic training data sets

UHD	Beth Israel	.958	.69	.389	.597	N/A
	Partners Healthcare	.953	.696	.462	.624	N/A
	Mayo Clinic	.593	.667	N/A	.5	.453
BART	Beth Israel	.59	.202	.166	.3	N/A
	Partners Healthcare	.475	.206	.253	.263	N/A
	Mayo Clinic	.41	0	N/A	0	0
Stanford	Beth Israel	.205	.076	0	.096	N/A
	Partners Healthcare	.251	.073	.074	.061	N/A
	Mayo Clinic	.069	0	N/A	0	0
LingPipe	Beth Israel	.243	.015	.029	.092	N/A
	Partners Healthcare	.139	.067	.088	.066	N/A
	Mayo Clinic	.071	0	N/A	0	0

5.1 Combining Results

Once result data was collected, combinations of link results from the rule based system and the BART system were examined since the BART system showed the highest amount of correct link predictions. After combining the results from the two systems as a union of the sets, the statistics showed an increase of about 1% in recall but a decline of about 15% in precision, bringing the f1 score down overall.

5.2 Challenge Results

In order to participate in the challenge, each team participating was given test data that did not include the gold standard co-reference chains. After processing the data, each team submitted the data for evaluation by the hosts of the challenge. Our system had an F1 score average of .895 on the data sets provided for the testing, which according to the hosts of the competition, places the system in the first tier for performance. The first tier included the four top scoring systems. The scores in the first tier ranged from .895 to .915.

6 Conclusion

Since the goal of the 2011 I2B2 shared task was to mark concept mentions as co-referent or not, the rule based system developed for this study was used to mark links in the test data released by the organization for the challenge. This decision was made based on the results from cross-checking the performance of each system on the training data provided. The results show the BART system performed the best out of the three publicly available co-reference systems tested in this study on this specific collection of data. The results also show that manually creating rules for co-reference based on observation of training data is a valid way to accomplish this co-reference task, particularly with the person type concepts in the I2B2 style annotations, and in this case performed well using the guidelines laid out by the hosts of the competition.

Acknowledgements. This work is funded by National Science Foundation grant CNS 0851984 and Department of Homeland Security grant 2009-ST-061-C10001.

References

1. Azzam, S., Humphreys, K., Gaizauskas, R.: Using Coreference Chains for Text Summarization. Department of Computer Science, University of Sheffield (1999)
2. Morton, T.: Using Coreference for Question Answering. Department of Computer and Information Science, University of Pennslyvania (1999)
3. I2b2 (2011), http://www.i2b2.org
4. Uzuner, O., et al.: 2011 i2b2/VA co-reference annotation guidelines for the clinical domain (2011), https://www.i2b2.org/NLP/Coreference/assets/

5. Bagga, A., Baldwin, B.: Algorithms for scoring coreference chains. In: Proceedings of the Linguistic Coreference Workshop at The First International Conference on Language Resources and Evaluation, LREC 1998 (1998)
6. Vilain, M., et al.: A model-theoretic coreference scoring scheme. In: Proceedings of the Sixth Message Understanding Conference, MUC-6 (1995)
7. Lita, L.V., Rogati, M., Lavie, A.: Blanc: Learning evaluation metrics for mt. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (2005)
8. Luo, X.: On coreference resolution performance metrics. In: HTL-EMNLP (2005)
9. Wikipedia – The free encyclopedia. Wikipedia (2011), http://en.wikipedia.org/wiki/F1_score
10. BART coreference system (2007), http://www.bart-coref.org/
11. Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., Manning, C.: A multi-pass sieve for coreference resolution. In: EMNLP 2010, Boston, USA (2010)
12. Alias-i (2008), http://alias-i.com/lingpipe
13. U.S. National Library of Medicine (2011), http://www.nlm.nih.gov/research/umls/
14. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)

VAHA: Verbs Associate with Human Activity – A Study on Fairy Tales

Hui-Ngo Goh, Lay-Ki Soon, and Su-Cheng Haw

Faculty of Computing and Informatics, Multimedia University, Jalan Multimedia, 63100 Cyberjaya Selangor, Malaysia
{hngoh,lksoon,sucheng}@mmu.edu.my

Abstract. Named entity recognition (NER) is a subtask in information extraction which aims to locate atomic element into predefined categories. Various NER techniques and tools have been developed to fit the interest of the applications developed. However, most NER works carried out focus on non-fiction domain. Fiction based domain displays a complex context in locating its NE especially name of person that might range from living things to non-living things. This paper proposes VAHA, automated dominant characters identification in fiction domain, particularly in fairy tales. TreeTagger, Stanford Dependencies and WordNet are the three freely available tools being used to identify verbs that are associated with human activity. The experimental results show that it is viable to use verb in identifying named entity, particularly in people category and it can be applied in a small text size environment.

Keywords. Named entity recognition, fairy tales, verb, dominant character.

1 Introduction

The concept of named entity recognition (NER) is not new in the area of information extraction (IE). It has been 21 years since the first NER published which focused on extracting company names [1] using heuristics and hand-coded rules. Thereafter, various predefined categories of NER have been explored to fit the interest of the applications intended to be developed. Among all, "name of people, organzaition and location"[2], [3], [4] are the most commonly explored predefined categories. Generally, NER aims to locate and extract significant atomic elements in texts into predefined categories.

Casey *et al.* employed machine learning approach to extract multiple NEs ranging from high level (place, person) predefined categories to low level (soccer player, universities) predefined categories in the web environment. A set of seed entities and relations, and learn templates are used to automatically generates training data [5]. Einat *et al.* applied conditional random fields and dictionary to extract personal names from email [6]. However, the unstructured nature of written email produced inconsistent performance results among corpora used. Repetition of names [7], [8] within documents have also been used to extract NE, but it might not perform well for documents that are small in size.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 313–322, 2012.

In 2010, Le *et al.* studied the use of inductive logic programming to extract named entities (name, diploma, organization, research) in Vietnamese language [9]. In the same year, Laura *et al.* proposed domain adaptation of rule-based annotator to enhance domain customization for NER by manually coded 104 features of domain-independent CoreNER library [2]. Public datasets of CoNLL03, Enron and ACE05 were used to train and test the "person, location and organization" entities.

However, most NERs developed above focused on non-fiction based documents. Non-fiction implies communicative works whose descriptions are generally written as facts. Therefore, non-fiction documents usually exhibit certain patterns in representing its NE. For an instance, name of person may start with designator, capital letter of the first character, and naming in a human way. On the other hand, fiction documents usually exhibit complexity and uncertainty in locating its NE. For example, the name of a person may be represented in diverse spectrums, ranging from living things (animals, plants, person) to non-living things (vehicle, furniture).

In this paper, we propose VAHA, a fully automated named entity recognition framework to overcome the above mentioned issue by studying the nature of verb(s) that associates with human activity. We aim to extract dominant characters in fairy tales. Dominant characters are the person depicted in a narrative and actively engage to get audience attention. Usually they are clearly identified through an impact play in a story regardless of the life span of their appearance in the story. Fiction-based domain is used to test the proposed framework. A predefined category of "name of person" is being investigated but our approach focuses on recognizing dominant characters in fairy tales. Stanford dependencies (SD) and TreeTagger are used to shallowly parse the natural language input files to identify the potential dominant character(s). Clauses that are tagged with the sequences of (i) noun(s) - verb(s) denotes Subject-Verb (S-V) and (ii) verb(s) - noun(s) implies Verb-Object (V-O) are being extracted. The extracted S-V and/or V-O at sentence level will then be verified with semantic dependencies produced by SD to conform it represents the sentence meaning. Later, two features of WordNet, namely derivationally related form (DRF) and sentence frames (SF) are used to substantiate verb that associates with human activity. Finally, subject or object that attach to the verb that associates with human activity will be regarded as dominant character. Part of this work is an extension of our previous effort in identifying protagonist in fairy tales using verb. This paper is segmented into 3 sections of technologies background, proposed system framework, experiments performed and end with conclusion.

2 Technologies Background

2.1 TreeTagger

TreeTagger[1] is a tool developed within the TC project at the Institute for Computational Linguistics of the University of Stuttgart for annotating text with

[1] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

part-of-speech and its lemma information. It is readily available in eight languages and adaptable to other languages with the availability of the training corpus. In this work, English is used to annotate on the selected natural language text file.

```
Between/IN these/DT pieces/NNS grew/VVD the/DT most/RBS beau-
tiful/JJ large/JJ white/JJ flowers/NNS;/: so/RB the/DT swal-
low/NN flew/VVD down/RP with/IN Tiny/NP,/, and/CC placed/VVD
her/PP on/IN one/CD of/IN the/DT broad/JJ leaves/NNS./SENT
```

Fig. 1. Natural language sentence with its corresponding TreeTagger tag

Fig. 1 shows the sentence that has been annotated with TreeTagger. Each word is attached with its corresponding part-of speech (POS) tag. An assumption is formed where dominant character(s) of a fairy tale is often tagged as "NOUN" generally, and specifically as common nouns and/or proper nouns. In this work, "NOUN" is identified as any POS tags that start with the label "N" such as "NN", "NNS" or "NP" whereas "VERB" is POS tags that start with the label "V". Clauses that are tagged with the sequences of (i) noun(s) - verb(s) denotes Subject-Verb (S-V) and (ii) verb(s) - noun(s) implies Verb-Object (V-O) are extracted. As shown in Fig. 1, there are two S-V extracted, namely "pieces/NNS grew/VVD", "swallow/NN flew/VVD".

2.2 Stanford Parser

2.2.1 Stanford Dependencies

Stanford Dependencies (SD) formulates its dependencies relation based on forty-eight grammatical relations according to the predefined tregex patterns over phase-structure trees [10]. Tregex is a matching patterns in trees based on tree relationship and regular expression. SD is represented in triplet structure with a grammatical relation used to tie up the right dependencies of two tokens as shown in Fig. 2.

In VAHA, SD is used to filter out each extracted S-V and V-O that does not conform to its corresponding sentence meaning. Grammatical relation of "nn", "nsubj", "nsubjpass" or "dobj" are used to examine against each extracted S-V and V-O at sentence level. Only S-V and V-O that have its corresponding SD with any of the four mentioned grammatical relations will be kept for further analysis. "nn" denotes noun compound modifier that serves to modify the head noun, "nsubj" implies nominal subject which is the syntactic subject of a clause whereas "nsubjpass" refers to passive nominal subject which is the syntactic subject of a passive clause; "dobj" means direct object of a Verb Phrase (VP) which is the object of a verb.

```
det(pieces-3, these-2)
prep_between(grew-4, pieces-3)
det(flowers;-10, the-5)
advmod(beautiful-7, most-6)
amod(flowers;-10, beautiful-7)
amod(flowers;-10, large-8)
amod(flowers;-10, white-9)
nsubj(grew-4, flowers;-10)
mark(flew-14, so-11)
det(swallow-13, the-12)
```

```
nsubj(flew-14, swallow-13)
dep(flowers;-10, flew-14)
prt(flew-14, down-15)
prep_with(flew-14, Tiny,-17)
conj_and(flew-14, placed-19)
dobj(placed-19, her-20)
prep_on(placed-19, one-22)
det(leaves.-26, the-24)
amod(leaves.-26, broad-25)
prep_of(one-22, leaves.-26)
```

Fig. 2. Stanford dependencies

Fig. 2 presents the grammatical dependencies for the sentence in Fig. 1. Based on the two extracted S-V mentioned in section 2.1, "pieces/NNS grew/VVD" does not conform with the sentence meaning as its grammatical relation is none of the four mentioned relation. It is not the "pieces" that grew, but in fact, it is the flower that grew. Hence, "pieces/NNS grew/VVD" will be discarded for its verb analysis. However, S-V of "swallow/NN flew/VVD" will be kept for further verb analysis as it conform to the sentence meaning and has the grammatical relation of "nsubj".

2.3 WordNet

WordNet [11] is an English lexical database that group set of words into Synonyms (Synsets). Each synset is interconnected by conceptual relations. Hence, it illustrates the co-reference among synsets in database in revealing the semantic represented. As of 2006, the WordNet database contains total of 155,287 unique words where verb has taken up 11,529 words organized in 13,767 synsets for a total of 25,047 word-sense pairs [12] which is sufficient to be used in this project. Two features of WordNet are used in this work, namely derivationally related forms (DRF) and sentence frames (SF). DRF indicates words that are derived from the same root. It shows relationship existed between groups of synsets. SF is specifically designed for VERB group; it contains a list of generic sentence frames exemplifying the types of simple sentences in which the verbs in the synset can be used.

3 Framework and Experimental Setup

This section explains the framework as well as the experimental setup.

Input : Fiction web pages[2] which contain eight fairy tales, as listed in Table 1 are chosen as they contain diverse spectrum of dominant character(s). Some of the dominant characters are represented as its real name like human being while some of the dominant characters are being symbolized as animal or inserts.

[2] http://www.kidsgen.com

Table 1. Fairy tales with the corresponding word count

Fairy Tale	Word Count
The Story of Snow White	1913
Cinderella	1077
Beauty and the Beast	1357
Rapunzel	1393
Thumbelina	4348
Ugly Duckling	841
Sleeping Beauty	1317
Ant and the Grasshopper	142

Step 1: Document cleaning
Each fairy tale web page is cleaned automatically using HTML Context Extractor [3]in order to get rid of non-text content (banner, audio, video, images). A pure text file (.txt) is produced at the end of the cleaning process.

Step 2: Pre-linguistic processing
Each pure text file will be shallowly parsed using two freely available text processing tools, namely (a) TreeTagger and (b) Stanford Parser.

(a) The POS tags annotated on the pure text file served as a scheme in extracting clauses (S-V and/or V-O) that contain potential dominant character(s).
(b) Grammatical relations supplied by Stanford dependencies (SD) is used to verify that the extracted patterns (S-V and V-O) conform with its corresponding sentence meaning.

Step 3: Feature extraction
Based on each annotated fairy tale using Treetagger, potential dominant characters are extracted based on two patterns, namely (a) Subject-Verb (S-V) and (b) Verb-Object (V-O).

(a) For S-V pattern, clause that contains noun(s)-verb(s) and noun(s)-who-verb(s) that are adjacent to each other will be extracted.
(b) For V-O pattern, determiner might appear in between verb and object. It is an article which is to introduce a noun. "a", "an" and "the" are the examples of article. Therefore, clause that contains verb(s)-noun(s) or verb(s)-determiner-noun(s) that are adjacent to each other will be extracted.

Step 4: Data filtering
Two filtering processes of (a) conformation of extracted S-V and V-O with its corresponding sentence and (b) main verb identification are performed to prepare an accurate data for verb analysis.

(a) Four grammatical relations of "nn", "nsubj", "nsubjpass" and "dobj" from SD are used to countercheck against all the extracted S-V and V-O. It aims to filter out unconformation of S-V and V-O with its corresponding sentence meaning. An example is illustrated in section 3.2.

[3] http://senews.sourceforge.net/KCE_README.html

(b) Given the filtered S-V and V-O from step 4(a), only verb which has POS tag of "VV", "VVD", "VVG, "VVN", "VVP" and "VVZ" are preserved for verb analysis. These POS tags have the main verb that describes an event taken by a subject or action imposed on an object, such as "take/VV", "took/VVD", "taking/VVG", "taken/VVN", "take/VVP" and"takes/VVZ". Section 2.1 describes TreeTagger POS tags.

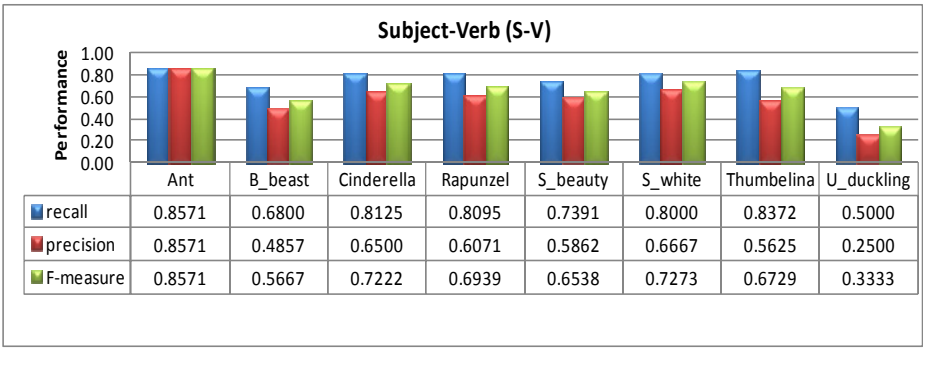
Finally, a set of filtered clauses is produced. They are notated as S-V_{can} and V_{can}-O.

Step 5: Verb analysis
Each extracted V_{can} that forms S-V_{can} or V_{can}-O will be served as keyword search in WordNet for retrieving its corresponding senses description. Only V_{can} that has the returned group(s) of VERB and /or ADJECTIVE from WordNet will be regarded for its DRF or SF. For DRF, each returned description will be examined sentence by sentence. In the presence of either key phrases of "someone who", "a person who", "to whom" and "one who" in the sentence, the verb is considered to be associated with human activity. However, in the absent of four key phrases in DRF, sentence frames is used to study the verb's usage. For S-V, if all the V_{can}'s sentence frames start with "Somebody ---- ", it implies an action has to be taken by human. For an instance, the word "sighed" for S-V of "Cinderella/NP sighed/VVD" in the story of "Cinderella". While for V-O, if all the V_{can} sentence frames end with "---- Somebody", it denotes an action is taken on human. Hence, the word is considered to be associated with human activity.

Output: Finally, S and O which are attached to verb(V) that associates with human activity are considered as dominant characters in the investigating fairy tale.

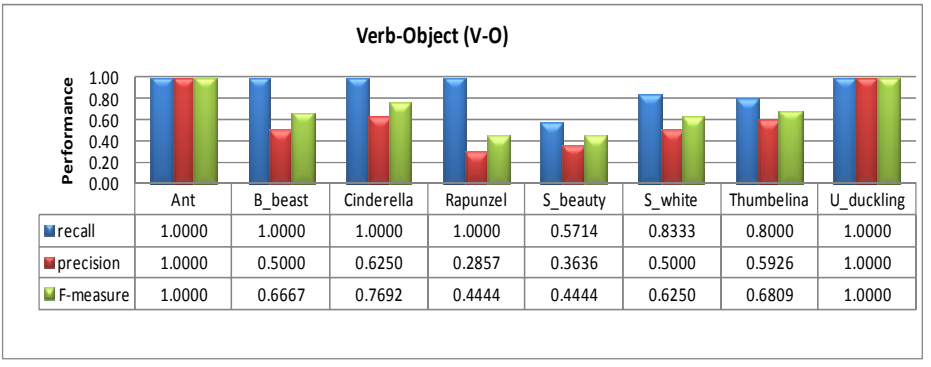
4 Results and Discussion

The eight chosen fairy tales that come in different file sizes and have diverse dominant characters are experimented to verify the performance of our proposed VAHA in identifying dominant character(s). The evaluation metrics used are precision, recall and F-measure. The experimental analysis were carried out individually for S-V as depicted in Fig. 3 and V-O as presented in Fig. 4, while the overall performance of VAHA is shown in Fig. 5.



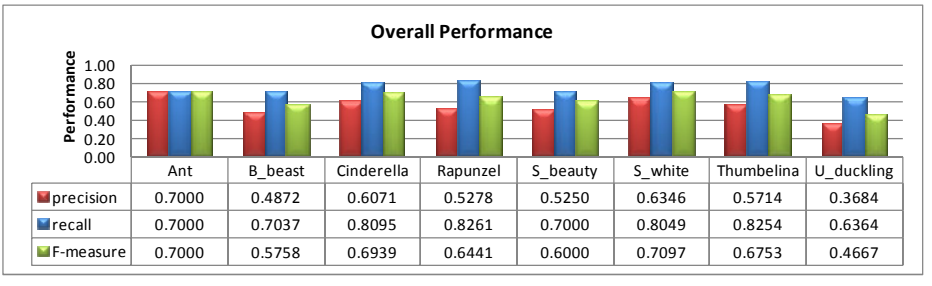
	Ant	B_beast	Cinderella	Rapunzel	S_beauty	S_white	Thumbelina	U_duckling
recall	0.8571	0.6800	0.8125	0.8095	0.7391	0.8000	0.8372	0.5000
precision	0.8571	0.4857	0.6500	0.6071	0.5862	0.6667	0.5625	0.2500
F-measure	0.8571	0.5667	0.7222	0.6939	0.6538	0.7273	0.6729	0.3333

Fig. 3. Performance results of "S-V" pattern for dominant character(s) identification



	Ant	B_beast	Cinderella	Rapunzel	S_beauty	S_white	Thumbelina	U_duckling
recall	1.0000	1.0000	1.0000	1.0000	0.5714	0.8333	0.8000	1.0000
precision	1.0000	0.5000	0.6250	0.2857	0.3636	0.5000	0.5926	1.0000
F-measure	1.0000	0.6667	0.7692	0.4444	0.4444	0.6250	0.6809	1.0000

Fig. 4. Performance results of "V-O" pattern for dominant character(s) identification



	Ant	B_beast	Cinderella	Rapunzel	S_beauty	S_white	Thumbelina	U_duckling
precision	0.7000	0.4872	0.6071	0.5278	0.5250	0.6346	0.5714	0.3684
recall	0.7000	0.7037	0.8095	0.8261	0.7000	0.8049	0.8254	0.6364
F-measure	0.7000	0.5758	0.6939	0.6441	0.6000	0.7097	0.6753	0.4667

Fig. 5. Performance results for overall dominant character(s) identification

As shown in Fig. 5, file size and dominant characters' groups do not impact the performance results of dominant character identification. This can be validated through the story of "Ant and Grasshopper" which has file size of only 142 words with the character group of inserts yields the best performance result compared to other fairy tales. Out of seven clauses of "*grasshopper* – was hopping", "*ant* - passed", "*grasshopper* – invited", "*ant* - sit", "*ant* -went" "winter-came" and "*ants* - distributing" being extracted for S-V pattern, six of the subjects (highlighted in italic) were correctly classified as character based on the verb attached to them except for the subject of "winter". While, "asked – the ant", "invited – the ant" and "said – the grasshopper" are the three clauses extracted for V-O pattern. The word of "asked" and "invited" are related to human activity. However, an observation was done for the word "said" of V-O pattern in all the investigated fairy tales. V-O clause that has the pattern of "verb-determiner noun" and appear immediately after the punctuation mark of " " " or " , " always denote noun as character except for one clause of "said – the spirit" in the story of " Snow White". "replied – the field mouse" and "exclaimed - the field mouse" are the two examples that share the same characteristics of the word "said" in the story of "Thumbelina". Therefore, clause that exhibits the above characteristics for V-O pattern are taken as a heuristic in identifying dominant characters.

The analysis for S-V pattern is rather straightforward as DRF exhibits its description of key phrases of "someone who", "a person who", "to whom" and "one who" in

the form of active sentence. For example, "someone who consumes", implies subject must be a person, while, V-O pattern possessed a more complex situation where an action (verb) can be taken on a person or thing. Besides, "drank – the dew", "heard – a voice", "spun – gold" and "sing – a wedding song" are some of the V-O extracted from fairy tales. As such, a correspondence senses between DRF and SF is needed to verify the uncertainty. SF that ends only with "---- something" implies the said object must be a thing and SF that ends only with "---- somebody" implies the said object must be a person. However, for the cases where SF that has the mixture of "something/somebody" at subject and/or object are currently ignored for this work and will be explored in future work. Therefore, the performance result of V-O pattern (Fig. 4) generally performs better than S-V pattern (Fig. 3). This is due to the number of clauses extracted for S-V pattern is more than the V-O pattern. Hence, the possibility of subject wrongly classified as character is higher. Moreover, structural format of V-O pattern has led to more details handling compared to a straightforward S-V pattern.

Recall, precision and F-measure are interrelated. Good information extraction should reflect high recall and high precision for high F-measure, which is hard to be achieved. High recall with low precision or low recall with high precision always becomes a struggling effort for researcher. An effort to improve either factor might cause the other factor to be deteriorated. In this work, high recall implies most of the verbs that associate with human activity are in fact attached to dominant character while low precision is due to many verbs which are not associated with human activity were being extracted as according to the patterns of S-V and V-O. Hence, the number of extracted pattern has increased and greatly impacts the result for precision. This scenario can be seen in Fig. 3, 4 and 5.

Table 2. Dominant dominant characters for fairy tales

Fairy Tale	Dominant Character
The Story of Snow White	**Snow White, King, Queen, Stepmother**(witch**, peddler woman), Prince, Huntsman,**
Cinderella	**Cinderella, Prince, Coachman**, **Stepmother**, Stepsister, **Fairy, Minister**.
Beauty and the Beast	**Merchant, Beast, Beauty.**
Rapunzel	**Enchantress, Rapunzel, Prince, Husband,** Wife.
Thumbelina	**Tiny, Toad, Swallow, Mole, Field Mouse, Cockchafer, Butterfly, Bird**, Prince.
Ugly Duckling	**Mother duck, Duckling, Geese, Hen**, Swan, Farmer
Sleeping Beauty	**Briar Rose**, Witch, **King, Queen, Frog, Prince**.
Ant and the Grasshopper	**Ant, Grasshopper**

Five different versions of each fairy tale were used to identify the common dominant characters. The existence of dominant characters in at least three versions the same fairy tale will be chosen as true dominant character. Table 2 presents the true dominant characters for each fairy tale. Dominant characters which are highlighted in bold are the dominant characters identifiable using VAHA. The repetitive occurrence of dominant character in subject and object is counted as one occurrence. In the

stories of "The Story of Snow White", "Beauty and the Beast" and "Ant and Grasshopper", our approach is capable to identify all of the listed dominant characters. However, VAHA did show good performance too in the rest of the stories. The active participation of each dominant character in story flow will likely increase its affiliation to verb that associates with human activity. An example of S-V and V-O patterns for "Ant and Grasshopper" shown above explain this justification. The dominant character "stepsister" is unidentifiable in the story of "Cinderella" as there are only two S-V pattern of "stepsister – getting", "stepsister – gaped". Moreover, the verbs of "getting" and "gaped" are not associated to human activity.

5 Conclusion

This paper describes VAHA, an algorithmic framework for automatic identification of dominant characters in fairy tales by studying the nature of verb that associates with human activity. Two different groups of dominant characters were used to test on VAHA, namely, entity and human alike name of dominant characters. TreeTagger, Stanford Dependencies and WordNet are the three freely available tools being used to identify verb that associates with human activity. Different handling has been taken on S-V and V-O pattern due to the different structural representation and characteristics of DRF exhibits. Our experimental results show that verbs can be used as a determinant in identifying "people" named entity in general and protagonist in specific. For future work, we wish apply VAHA in news articles and to look into verb disambiguation to improve the performance result of dominant characters identification.

References

1. Lisa, F.R.: Extracting Company Names from Text. In: Proc. IEEE Conference on Artificial Intelligence Applications, pp. 20–32 (1991)
2. Laura, C., Rajasekar, K., Li, Y.Y., Frederick, R., Shivakumar, V.: Domain Adaptation of Rule-Based Annotators for Named-Entity Recognition Tasks. In: Empirical Methods in Natural Language Processing, Massachusetts, pp. 1002–1012 (2010)
3. Andrew, M., Li, W.: Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-Enhanced Lexicons. In: 7th Conference on Natural Language Learning, pp. 188–191 (2003)
4. Dan, K., Joseph, S., Huy, N., Christopher, D.M.: Named Entity Recognition with Character-Level Models. In: 7th Conference on Natural Language Learning, pp. 180–183 (2003)
5. Casey, W., Alex, K., Nemanja, P., Lyle, U.: Web-Scale Named Entity Recognition. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, Napa Valley, California, USA, pp. 123–132 (2008)
6. Einat, M., Richard, C.W., William, W.C.: Extracting Personal Names from Email: Applying Named Entity Recognition to Informal Text. In: Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Vancouver, pp. 443–450 (2005)

7. Charles, S., Andrew, M.: Collective Segmentation and Labeling of Distant Entities in Information Extraction. In: ICML Workshop on Statistical Relational Learning (2004)
8. Razvan, C.B., Raymond, J.M.: Relational Markov Networks for Collective Information Extraction. In: ICML- Workshop on Statistical Relational Learning (2004)
9. Le, H.T., Nguyen, T.H.: Name Entity Recognition using Inductive Logic Programming. In: Symposium on Information and Communication Technology, Vietnam, pp. 71–77 (2010)
10. Marie-Catherine, D.M., Bill, M., Christopher, D.M.: Generating Typed Dependency Parses from Phrase Structure Parses. In: LREC (2006)
11. Christiane, F.: WordNet:An Electronic Lexical Database. MIT Press, Cambridge
12. WordNet Statistic, http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

A Semantic-Based Social Network of Academic Researchers

Elnaz Davoodi[1] and Keivan Kianmehr[2]

[1] Institute for Advanced Studies in Basic Sciences
Zanjan, Iran
elnazood@gmail.com
[2] University of Western Ontario
London, Ontario, Canada
kkianmeh@uwo.ca

Abstract. We proposed a framework to construct a semantic-based social network of academic researchers to discover hidden social relationships among the researchers in a particular domain. The challenging task in in the process is to detect accurate relationships that exist among researchers according to their expertise and academic experience. In this paper, we first construct content-based profiles of researchers by crawling online resources. Then background knowledge derived from *Wikipedia* ,represented in a semantic kernel, is employed to enrich the researchers' profiles. Researchers' social network is then constructed based on the similarities among semantic-based profiles. Social communities are then detected by applying the social network analysis and using factors such as experience, background, knowledge level, personal preferences. Representative members of a community are identified using the eigenvector centrality measure. An interesting application of the constructed social network in academic conferences, when there is a need to assign papers to relevant researchers for the review process, is investigated.

Keywords: Social Network Analysis, Clustering Analysis, Semantic-based Similarity, Information Retrieval.

1 Introduction

The study of Social Networks (SNs) started in sociology [1] to analyze social communities of humans within different contexts, including social interactions, business communications, international relations, political movements, etc. However, it is applicable to every domain where the interactions between items can be represented using a kind of network such as web mining [2,3], biological networks [4], etc. The basic and the most time consuming step in the process of Social Network Analysis (SNA) is to build the social network. Once the social network is constructed, different measures can be applied to study the characteristics of the social network and hence it is necessary to decide correctly and clearly on elements of the model, namely the actors and the interactions between them. A relation is generally defined as a specific kind of contact, connections,

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 323–332, 2012.

or tie between a pair of actors; however it is not an attribute of one actor, but is a correlation that exists only so long as both actors maintain their association [5]. Thus, it is important to discover accurate relations among actors which reflect their interaction in reality such that if relationships cannot be discovered and formed accurately, the network structure will not demonstrate actors' interactions truly. Indeed, in common methods of social network construction, the semantic associations among individuals are not considered and their relationships may be falsely built. Discovering semantic relationships among entities of a social network which leads to a semantic-based social network is a promising solution to this problem; however discovering semantic relationships is a challenging task. To cope with the problem of finding semantic relationships, external knowledge should be extracted from existing knowledge resources. There exist several ontologies like WordNet [11] which have been used as external sources for embedding background knowledge to text documents [6], but these ontologies are manually built and their coverage are too restricted. Further, using ontology terms as replacement or additional features may cause information loss, or introduce noise. For these reasons, in this research we make use of *Wikipedia*, the largest electronic knowledge repository, to enhance social entities representation and consequently discover semantic relationships among them.

A semantic-based social network can be utilized in a wide range of domains [8], especially in knowledge management when there is a need to search for a group of experts who have relevant knowledge and expertise with respect to users' information need. In this paper, we have investigated a typical application of a semantic-based social network of academic researchers to the review process of papers in academic conferences and scientific journals. In fact, in academic conferences, submissions vary among a wide range of topics and due to the importance of the review process for selecting papers; the conference chair attempts to precisely assign papers to members of program committee who are able to judge and review any particular paper with a high level of confidence. Therefore, one of the most crucial tasks of a program chair is to select appropriate researchers from committee members to review papers which are relevant to the researchers' area of expertise, knowledge, and experience. We argue that if communities of academic researchers can be accurately identified, papers can be assigned to the most appropriate researchers in the specific group in which all members resemble semantically and have similar expertise relevant to a particular topic.

This research intends to conduct a thorough investigation on the issue of social network extraction of academic researchers. Previous researches have not considered the semantic relationships among social entities [7]. Hence, in this paper we aim to build a semantic-based social network of researchers. In the proposed framework, a profile is first constructed for each individual researcher by using the information collected from different online sources. Then, profiles are enriched by integrating external semantic knowledge extracted from *Wikipedia* and accordingly semantic associations among researchers can be discovered among enriched profiles. A social network is constructed according to the similarities among the experts' profiles and communities are then detected by a clustering

analysis technique. Finally, representatives of communities are identified by applying eigenvector centrality. Eventually, each submitted paper can be assigned to researchers in a community based on the similarity between the semantic-based profiles of community representatives and the domain of submitted papers which is manually decided by the conference chair or is automatically identified based on the keywords listed in the paper.

The rest of the paper is organized as follows. Section 2 describes our primary approach for constructing the researchers' profiles. Section 3 describes our proposed method for constructing the semantic-based social network of academic researchers. Section 4 reports the experimental results. Finally, the paper is concluded in section 5.

2 Constructing Researchers' Profile

To build a rich researcher's profile, different types of relevant information need to be collected. The profile information is obtained by crawling the web and extracting information to create a profile from relevant web pages to a particular individual. Profiles constructed in this manner contain relevant information such as work experience, educational history, social activities, abilities and specialties, interests, etc. to each individual.

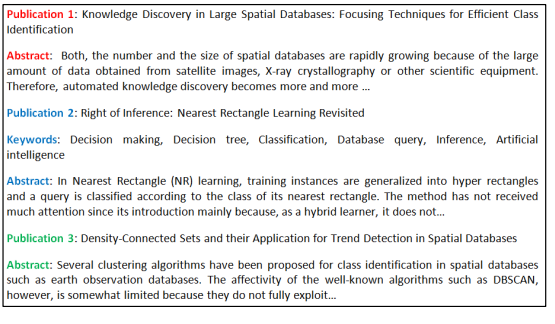


Fig. 1. An example of a researcher's profile

In this research work, we have focused on researchers' academic experiences which are reflected in their publications. In particular, three different categories of information in researchers' publications are considered: title, abstract, and keywords which are all published in digital libraries. For retrieving this information, a wide range of digital resources such as *DBLP*, *ACM*, *IEEE*, and *Springer* have been crawled. First, *DBLP* is used to extract the publication list of each researcher as well as some information related to each paper like address of digital editions, date of publishing, etc. Once the titles of all publications are extracted from the list by the crawler, for each publication the address of digital library that publishes the digital edition of the publication, is used for extracting its abstract and keywords. In this stage, if digital libraries do not have any access

limitation, content of publications can be obtained; otherwise the crawler uses *Google Scholar* for extracting abstract and keywords of corresponding papers that are not freely available to public through digital libraries and need users to subscribe. Fig. 1 demonstrates a part of a profile constructed in this manner for a researcher.

3 Constructing a Semantic-Based Social Network of Researchers

Wikipedia, the world largest electronic encyclopedia to date, has been recently used for text representation enrichment [9]. *Wikipedia* is a well-formed document repository in that each article only describes a single topic. The title of each article is a succinct phrase which is considered as a concept. Equivalent concepts are related to each other by redirected links and are referred to the same page on the *Wikipedia* directory. Meanwhile, each article (concept) belongs to at least one category, and categories are organized in a hierarchical structure. In the context of our work, we take advantage of *Wikipedia* ontology to embed semantic information into profiles. A social network which is constructed by considering semantic-based relationships among its actors is more reliable and will be a true representative of a real social network.

3.1 Extracting Semantic Knowledge from *Wikipedia*

To extract semantic knowledge from *Wikipedia*, a content-based method is applied to enable system find proximity between *Wikipedia* concepts, thus connections between concepts can be established. In this method, each *Wikipedia* article (*i.e.*, concept) is represented by a *tf-idf* vector. The similarity between concepts are measured by computing the cosine similarity of their corresponding vectors. Then, a symmetric concept-concept matrix, called semantic kernel S, is created to present similarities among all pairs of *Wikipedia* concepts. Each element $S_{i,j}$ of this matrix determines the cosine similarity between a pair of concepts with indexes i and j, respectively, where $i, j \in \{1, 2, \ldots, c\}$ and c is the total number of concepts considered. If a row and a column refer to the same concepts or two synonym concepts, the similarity value is 1. Note that queries on synonym concepts are redirected to the same page by *Wikipedia*. Further, the more similar two corresponding concepts are, the higher the value of the corresponding entry is. This kernel represents semantic relationships among all *Wikipedia* concepts according to similarities of their corresponding articles.

3.2 Integrating Semantic Knowledge into Researchers' Profiles

To integrate the semantic knowledge represented in matrix S into profiles, first a type of relation needs to be defined that associates profiles to *Wikipedia* concepts. For this purpose, a scheme based on the concept match is adopted to map the text document profiles to the *Wikipedia* concepts directly. In this mapping

scheme, profiles are scanned and similarity-based correlations between *Wikipedia* concepts and each profile are measured. To calculate the similarity between a profile and a concept, the *tf/idf* representation method is utilized. Profiles and concepts are presented in form of vectors in which dimensions are *Wikipedia* concepts. Expert profiles are considered as a collection of documents and each concept is considered as a phrase query which can be assumed a short text document. In addition, all operations that are applied to documents in *tf/idf* approach, like porter stemmer or removing stop words, now are applied to concepts that are considered as query phrases. Finally, the cosine similarity is used to measure the similarity between pairs of corresponding vectors of document profiles and *Wikipedia* concepts. The result is presented in a document-concept matrix D in which a row entry represents a profile, columns are *Wikipedia* concepts, and each element $D_{i,j}$ denotes the cosine similarity between a document i and a concept j of *Wikipedia*, where $i \in \{1, 2, 3, \ldots, n\}$, $j \in \{1, 2, \ldots, c\}$, n is the number of documents, and c is the number of concepts.

Once the document-concept similarity matrix is built, the semantic knowledge represented by the semantic kernel can be integrated into the profile representation. For this purpose, a linear combination of the document-concept matrix D and the semantic kernel S is applied and and a new semantic-based document-concept similarity matrix R is generated. The new matrix represents the semantic-based profiles. Each element $R_{i,j}$ is calculated as follows:

$$R_{i,j} = \sum_{k=1}^{c} D_{i,k} \times S_{k,j} \tag{1}$$

, where k is the number of concepts, $1 \leq i \leq m$ is the row index and $1 \leq j \leq n$ is the column index. As the formula shows, the occurrences of all other concepts in i^{th} document affect the semantic relationship between j^{th} concept and i^{th} document as well by considering the weights of all concepts' similarities to the jth concept. In other word, the weight of each concept's influence on the semantic relationship between a specific concept j and a document i is equal to the similarity of that concept to concept j. Fig. 2 shows an example semantic-based document-concept similarity matrix resulted from the linear combination of a document-concept matrix D and a semantic kernel S. In this example, only three *Wikipedia* concepts are shown.

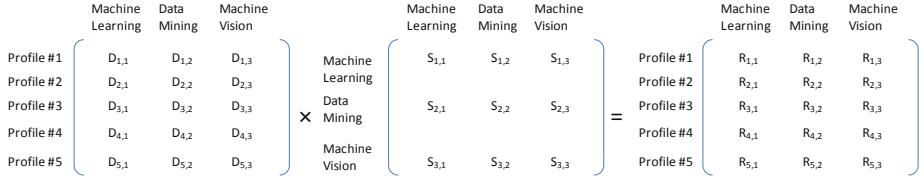


Fig. 2. Linear combination of D and S that produces R

3.3 Construction the Social Network of Researchers

In order to build the social network of researchers, a relationship between researchers should be defined. For this purpose, researcher profiles are considered as nodes of the network and semantic-based similarities among all pairs of profiles are considered as edges. In order to compute the semantic proximity of profiles to each other, an operation widely used in social network analysis, namely folding, is applied. Assume the semantic-based document-concept similarity matrix R, in which rows represent documents and columns represent concepts. Multiplying the similarity matrix R, by its transpose R', will produce a new symmetric matrix in which rows and columns both represent profiles and elements quantify the semantic relationship between pairs of researcher profiles. This recently generated similarity matrix is used to construct the links in the social network of researchers. For each pair of profiles, if their corresponding similarity in the similarity matrix is a none zero value, then a link is established between the corresponding researchers in the network.

3.4 Detecting Communities of Researchers

A community is typically thought of as a group of nodes with more interaction amongst its members than between its members and the remainder of the network. Different clustering algorithms can be applied for this purpose. In this study, the aim is to detect communities of researchers such that there are stronger similarities between cluster members, in terms of expertise, knowledge, and experience, than between cluster members and other members of network. We have chosen k-means clustering algorithm to detect the communities of researchers. Further, two measures, homogeneity and separateness [10], are used to evaluate clustering solutions. Since these objectives are conflicting, k-means algorithm is applied with various numbers of clusters (k) until an acceptable compromise is achieved. In other words, we have to trade off between maximizing homogeneity and minimizing separation. In order to apply k-means algorithm to cluster the social network, each node (researcher) is represented by a vector whose features are the semantic-based similarities to all other actors in the network. Clearly, the recently generated similarity matrix can be used for the clustering purpose as each row of the matrix presents the similarity of a researcher to all other researchers.

3.5 Finding Representatives of Communities

Usually clustering solution can be summarized by introducing a representative member for each cluster. In our work, since each cluster represents a researcher community, the representative member of a cluster is in fact a researcher who summarizes that community in terms of the knowledge, experience, and expertise carried by its members. To find a cluster representative, we have decided to use a centrality measure, called *eigenvector centrality*, which is widely used in social network analysis. According to the *eigenvector centrality*, a node is central

to the extent that its neighbors are central. Members who are connected to many otherwise isolated individuals will have a much lower score in this measure then those that are connected to groups that have many connections themselves. In our domain, the *eigenvector centrality* follows that a researcher well-connected to well-connected researchers can carry on valuable types of knowledge and experience much more widely than one who only has connections to lesser important researchers in a community. Researchers with higher scores of *eigenvector centrality* are more favorable when it is needed to find the right people whom we may ask a specific question and who will answer that question for us.

4 Experimental Results

The experimental domain of this research is dedicated to analyze a semantic-based social network of academic researchers in the field of computer science. Therefore, we have chosen 315 individuals from this field of study. These researchers have been selected from program committee members of the 16th *ACM SIGKDD*[1] conference. In addition, 62 keywords listed under the "conference topics" have been used as information items for which the program chair is seeking for relevant researchers. This set of keywords covers a wide range of scientific topics in the field of knowledge discovery and data mining. The main goal of this experiment is to assess the effectiveness of proposed semantic-based social network of academic researchers in the task of assigning papers to members of the program committee for the review process of the conference.

In the first stage of the experiment, a profile has been created for each researcher. A crawler programmed in C# programming language automatically collects profile information from online sources. 315 profiles have been constructed in this manner each of which corresponds to an individual researcher. Next profiles should be enriched by adding external knowledge extracted from *Wikipedia*. To extract the semantic knowledge from *Wikipedia* articles, we have automatically extracted *Wikipedia* pages and a tree-like structure of *Wikipedia* thesaurus has been constructed. In this structure, concept pages are located in tree leaves while internal nodes indicate category pages. In addition, it should be mentioned that since the number of *Wikipedia* articles in the area of *Computer Science* is too large, not only they need much space and computation to be processed, but also their relevance to this area is decreased by further proceeding in the tree structure. To cope with these problems, we cut the tree rooted in *Computer Science* category at the level of 7.

In the clustering phase of the experiment, we have employed *Weka*[2], a data mining and machine learning tool. Various clustering algorithms are implemented in this tool. According to the type of data features that were semantic similarity between every pair of nodes, we used the k-means clustering algorithm to group social network entities into different clusters. To choose the best clustering solution in this stage, two aforementioned criteria (homogeneity and

[1] `http://www.kdd.org/kdd2010`
[2] `http://www.cs.waikato.ac.nz/ml/weka/`

separation) have been used. In addition, for detecting representative individuals in each community, *ORA*[3], which is a dynamic meta-network assessment and analysis tool, was used. We first applied *ORA* to form the social network communities. Later, this tool was used to calculate and report the eigenvector centrality of all community members. For each community, the member with the highest eigenvalue was reported as the representative of that community. In our experiment, we have evaluated the results from two perspectives: first we report the k-means clustering results that lead to choose the best clustering solution, and second we discuss the effectiveness of the proposed semantic-based social network in the review process of the academic conferences.

4.1 Clustering Experiment

The community detection accuracy has been evaluated based on two criteria; homogeneity and separation such that the best clustering has maximum homogeneity and minimum separation. There are different clustering algorithms, but according to the nature of our data features, k-means clustering has been chosen for our experiment. We examine various clustering solutions, generated by the algorithm using different values of k in the range of 10 to 40. Then we choose a clustering solution which is an acceptable trade off between maximizing homogeneity and minimizing separation as the best solution among others. The range of k is chosen based on the number of researchers as well as the number of information items (conference topics) such that the average number of researchers in each cluster varies in a reasonable range. The results of homogeneity and separation for the Semantic-based Social Network (SSN) constructed based on concept match scheme and the Social Network (SN) constructed without considering the semantic-based relations are shown in Fig. 3(a) and Fig. 3(b), respectively. In these figures, the number of clusters in horizontal axis is plotted against the average value of homogeneity and separation in vertical axes that are distinguishable according to the shape and color of feature by which they are represented. The best clustering solution in the clustering analysis of the semantic-based social network using above mentioned scheme is the one that includes 12 clusters while in the social network in which semantic-based relationship are neglected, the best solution includes 10 clusters .

4.2 Paper Assignment Using Researchers Social Network

In order to evaluate the effectiveness of the proposed model, we have utilized it to find researchers who are experts in a given research topic. In fact, one of the most beneficial aspects of identifying communities in this model is that researchers within the same community are semantically similar regarding their expertise and each representative member can represent the knowledge and expertise of all members within the same community better than any other member in his/her community since his/her similarity to mate elements is the highest

[3] `http://www.casos.cs.cmu.edu/projects/ora/`

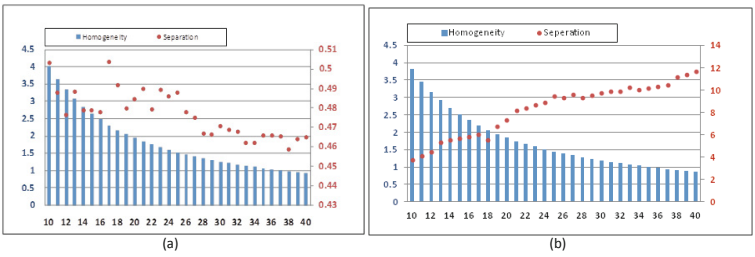


Fig. 3. Results of homogeneity and separateness for different clustering solutions

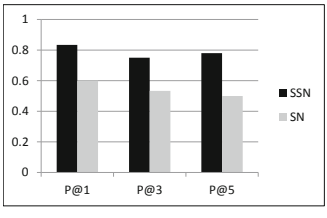


Fig. 4. Performance of SSN model compared with SN model in expert finding experiment

among all other mates. Thus, whenever an expert who has relevant expertise to a specific information domain is needed, a reliable choice is to trust the representative of the community whose members are experts in that field. In addition, if more than one expert is needed, other community members can be recommended according to their importance indicated by eigenvector centrality measure; community members with higher eigenvector centrality are more reliable in that specific domain. In this experiment, representative members of all communities in both Semantic-based Social Network (SSN) and Social Network (SN) without considering semantic relations are considered as test set. To evaluate the accuracy of the assigned items, a metric called precision at n or $P@n$ was used. This precision is defined as the fraction of retrieved instances that are relevant. Precision takes all retrieved items into account, but it can also be evaluated at a given cut-off rank, considering only the topmost results returned by the system. We consider k-top most relevant items that the system assigns to researchers and investigate how many of them are actually relevant considering the researchers real interests collected through questionnaires. In using of $P@n$, we set n to 1, 3 and 5. For example, $P@1$ indicates the percentage of researchers who are assigned relevant information when only one information item (research topic) is considered. Fig. 4 demonstrates the results of this set of experiments. As can be seen, a significant improvement obtained by considering semantic relationships among individuals. In other words, this experiment shows the effectiveness of the semantic-based social network of researchers when there is a need to assign papers to relevant researchers for the review process.

5 Conclusion

We presented a formwork for building a semantic-based social network of academic researchers in which social relationships are formed based on content-based profiles of researchers. Researchers profiles, once enriched with semantic knowledge, are used to discover hidden semantic-based relationships among pairs of researchers. We have utilized the semantic relationships to build a social network of researchers. A clustering technique has been employed to detect communities. We conducted experiments to show the effectiveness of the constructed social network in academic conferences, when there is a need to assign papers to relevant researchers for the review process, is investigated.

References

1. Carley, K.M., Gasser, L.: Computational organization theory, pp. 299–330. MIT Press, Cambridge (1999)
2. Flake, G.W., Lawrence, S., Giles, C.L.: Efficient identification of web communities. In: Proceedings of the Sixth ACM SIGKDD 2000, New York, USA, pp. 150–160 (2000)
3. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM 46, 604–632 (1999)
4. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99(12), 7821–7826 (2002)
5. Knoke, D., Yang, S.: Social network analysis, 2nd edn. Quantitative applications in the social sciences, vol. 154. Sage, Los Angeles (2008)
6. Hotho, A., Staab, S., Stumme, G.: WordNet improves text document clustering. In: Semantic Web Workshop of the 26th ACM SIGIR 2003, Toronto, Canada (2003)
7. Tang, J., Zhang, D., Yao, L.: Social Network Extraction of Academic Researchers. In: Proceeding of the 2007 IEEE International Conference in Data Mining, Omaha NE, USA, October 28-31, pp. 292–301. IEEE Computer Society (2007)
8. Davoodi, E., Afsharchi, M., Kianmehr, K.: A Social Network-Based Approach to Expert Recommendation System. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part III. LNCS, vol. 7208, pp. 91–102. Springer, Heidelberg (2012)
9. Wang, P., Hu, J., Zeng, H.-J., Chen, L., Chen, Z.: Improving text classification by using encyclopedia knowledge. In: 7th IEEE International Conference on Data Mining, Washington, DC, USA, pp. 332–341 (2007)
10. Everitt, B.S., Landau, S., Leese, M., Stahl, D.: Cluster Analysis. 5th edn. John Wiley & Sons Ltd press (2011)
11. Miller, G.A., Beckwith, R., Fellbaum, C.D., Gross, D., Miller, K.: WordNet: An online lexical database. Int. J. Lexicograph. 3(4), 235–244 (1990)

Heuristic Resource Discovery in P2P Network

Zheng Chen, Lei Wu, Jiansong Zhang, Xuemei Hu, and Yang Xu

University of Electronic Science and Technology of China,
No. 4, Section 2, North Jianshe Road
610054 Chengdu, China
Chenzheng.prc@gmail.com, Xuyang@uestc.edu.cn

Abstract. How to discovery resources effectively in a distributed system without centralized index services has been gained great concerns. In this paper, we proposed a modified chord algorithm, built a semantic P2P network. In research, we generated the resource's index by its semantic knowledge, and dynamically established routes between peers by the accumulated knowledge from their received search results. Experiment results showed that our prototype system has improved efficiency than the classic Chord. Furthermore, we also proposed a method of information sharing to accelerate the route establishing process.

Keywords: P2P network, semantic, heuristic, chord, H-chord.

1 Introduction

With the development of information technology, the needs of large-scale computation and storage are imperative. As a cheap, flexible and efficient computing infrastructure, distributed computing system based on P2P network has been paid more and more attention[1].

In P2P network, there is a major challenge that how to make the resources, no matter computing resources or storage resources, effectively match the requests of end users. This challenge is composed of a series of issues, such as resource discovery, resource monitoring, resource inventories, resources provisioning and fault tolerance. But most researchers deemed the resource discovery problem as the most fundamental one[2].Gnutella and some early P2P networks, which are called non-structured network, used flooding to solve this problem, but could not achieve an acceptable efficiency. In order to identify and locate resources in P2P network more efficiently, a series of algorithms over structured P2P networks have been put forward, including Chord[3], CAN[4], Pastry[5], Tapestry[6], and etc. By using a DHT to indexthe resources, structured P2P networks offer the lookup functionality similar to a centralized database but can be used in a decentralized way, which greatly improves the availability of P2P networks.

Chord, proposed by MIT in 2001, has a DHT-based, structured, fully distributed architecture. This algorithm has some main advantages. Its performance outperforms CAN in locating resources, and its maintenance cost is less than Pastry algorithm and

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 333–342, 2012.

Tapestry algorithm when the nodes dynamically join or leave[2]. However, Chord is still under developing: 1. The resources index allocation is completely random, when many resources associated were far away from their imperative requests, the resource discovery could be very difficult. 2. Chord routing table is static and the knowledge of the processes to allocate previous resources cannot be used to improve efficiency for future resource allocation.

In this paper we proposed a novel chord modification, which is called heuristic chord (H-chord). In H-chord, other than the completely random method in classic chord, resource semantic is used to generate the index of resource, so that similar resource could be located at nearby location. Moreover, in the each running circle, any successfully allocated resource locating request could generate some knowledge contributed to update the route table, which is denoted the semantic connection, and make the future resource locating request more efficient.

We designed the new Chord algorithm with the basic assumption that people's search requests for resources are not completely random, but semantically related. That means each search is always associated with the previous search. For example, someone searches the "Steve Job's speech at the WWDC" this time, and then his next search is likely to be "The presentation of ipad's new feature" or "IOS5 SDK", because these resources are similar or related in semantics. And the user is less likely to search for a thing that has nothing to do with the previous search like "kung fu panda 2". With this assumption we improve the classic Chord algorithm as the following:

1. Add semantic assistance in the process of constructing index. In the classic Chord algorithm, the hash procedure is completely random: cannot make use of the resource's semantic knowledge. In H-Chord, we add semantics to the hash procedure. We analysis the semantics knowledge of resources at first, and then generate an index based on its semantics. The hash function we use to create the index is segmented, which makes the hash values of the resources whose semantic knowledge is similar are similar, thus the nodes (described by hash value) which store these resources' indexes are neighboring, resulting in faster positioning.
2. Add the feedback mechanisms in the search process. After a successful resource locating, the located node has to send feedback information to the node that starts the search. The start node can adjust its routing table by some rules, update items' ranks, and add located node into the routing table. After a series of operations, the semantic association between the nodes will be well reflected in the node's routing table, enabling faster response to the search requests.
3. In order to speed up the convergence process of routing table updating, we add a knowledge-sharing mechanism between the nodes, allowing the node to use its neighboring nodes' routing table information to update its own routing table to make better use of knowledge resulting by system running to make the semantic correlation between the index nodes more quickly.

In order to achieve these improvements, we have to change the data structure and processes of the classic chord algorithm. The detail will be shown in the following section.

2 Heuristic Chord

2.1 Semantic-Based Resource Index

In the establishing procedure of chord algorithm, a consistent hash function is used for all nodes to calculate a unique node ID (node identifier). In order to make the semantic similar resources be found in the same or adjacent nodes, we need to modify the hash function.

2.2 Segmented Hash Function

We propose a semantic-based segmented hash function as follows:

$$\begin{aligned} hash_{se}(\text{resource}_i) = \text{concat}(&hash(\text{cate}_{i,j_0}),\\ &hash(\text{cate}_{i,j_1}),.....,hash(\text{cate}_{i,j_k}),\\ &hash(\text{resource}_i)) \end{aligned} \tag{1}$$

Where cate_{x,i_x} are the categories that $resource_i$ belong to. When we want to calculate $hash_{se}(resource_i)$ for resource_i, we need to get its categories firstly. Its hash value is the combine of the hash value of $cate_{i,j_x}$ and its own hash value. Obviously, how to determine the category of resource $cate_{i,j_x}$ will have a significant impact. In creating index of resources, the file name is the only information that we can get. However, the literal meaning of the resource file name may not be a good representation of the semantics, such as "Pirates of the Caribbean" and "Kung Fu Panda", it is difficult to find them belong to "popular movies" just according to its literal meanings. So we add the semantics to the classification of resources. There are two steps in the classification based on semantics: 1.Extract the keywords from the resource file name, 2.Classify the keyword based on semantics.

2.3 Semantic Category Retrieval

There are three steps in the classification of resources based on semantic: 1.Extract the keyword from the resource filename. 2.Get the categories of the keyword based on semantic. 3.Scoring the categories set.

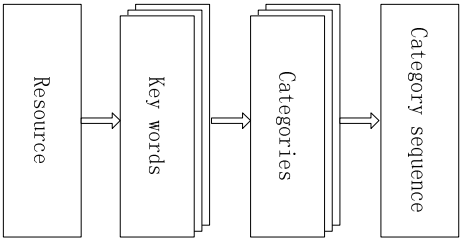


Fig. 1. Four steps of generating resources' classification

In order to establish the classification of keyword based on semantic, there are two main approaches. One is based on previous knowledge accumulated ontology[7]. The other is based on latent semantic[8]. Since the ontology has an irreplaceable advantage in syntactic ambiguity, resolving the semantic ambiguity, information retrieval and machine translation, it is widely used. There are a variety of general common-sense ontology library systems[9], such as WordNet, DBpedia, Cyc, HowNet, Frame Ontology and DublinCore. and a large number of domain ontology library systems which are different from the ontology that based on human knowledge. As we don't want to give our algorithm a domain limitation, and our work mainly focuses on the heuristic method, the classification is just the pre-procedure of heuristic locating, so how to generate the resources' classification will not our main concern. In our future work, we may focus on a specific domain, such as music sharing, to discuss this part in detail.

2.4 Heuristic Search of Resources

In the ideal case, with our modified hash function and the process of classification of resources, the index of resources whose semantic are similar will be stored in the neighboring nodes. The next issue to consider is how to use the information to make the search faster. Specific to the chord, each node needs to maintain a routing table that store a number of location information of the nodes, so the problem changes to how to generate and maintain a routing table for each node, to meet the following requirements: 1. quickly locate the nearby nodes; 2. quickly locate the relevant nodes.

2.5 Establish the Routing Table

As we introduced, classic Chord model can support rapid random resource locating by binary search. Since our modified resources indexing method makes semantic similar resources indexed by neighboring nodes, so we need to modify the routing table to locate the neighbor nodes. We design a two-way routing table instead of the classic one-way routing table to make the neighbor searching possible.

H-Chord routing table is established by the following rules: Suppose chord space size is 2^m, the number of nodes is $N \le 2^m$, and the routing table of each node contains m items. The *i*th (i is even) record the index of the -2^i offset of the successor node, and the *i*th (i is odd) items record the index of the -2^i successor node. Accordingly, the interval and the successor of each item are modified too. The $\text{interval}_i = [mid_i, mid_{i+1})$, where $mid_i = \lfloor (start_i + start_{i+1}) / 2 \rfloor \bmod 2^m$, It is noteworthy that $mid_0 = n+1$ and $mid_m = n$, where n is the node ID. successor_i is also modified to successor of mid_i.

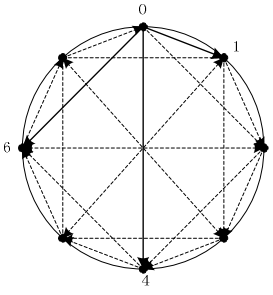


Fig. 2. A full H-chord ring with 8 nodes

A full Chord ring of $m = 2$ is shown in Fig.3, the arrow represents the routing table items. The routing table of $node_0$ has three items. The $item_0$ point to the $node_1$, where $n = 0 + 2^0 = 1$ .The $item_1$ point to the node $node_6$, where $n = (0 + 2^1) \bmod 2^3 = 6$. And the $item_2$ point to the node n ($n = 0 + 2^2 = 4$).
A typical eight-node chord ring is shown in Fig.4:

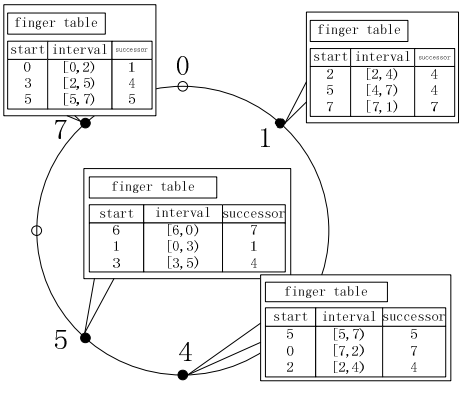


Fig. 3. A typical H-chord ring with 4 nodes in chord space 8

Through this chord ring, we can better positioning to the neighboring nodes, no matter it's before or after it. Thus making the semantics of search results with previous similar resources can be searched quickly.

2.6 Routing Table Updates

H-Chord algorithm does resource locating with the previously defined routing table. We did not change the process of classic chord resource locating, which is a simple but efficient process we discussed in section2.2, just do two small modifications:

1. Each resource locating request will start on the node that the last request ends;
2. After the end of each query, the node where the resources were found will seed a feedback to the original one.

If there are search requests that start at $node_{n1}$ are always ended at $node_{n2}$, we can get that the resources indexed by $node_{n1}$ and $node_{n2}$ are in a kind of semantic relevance. In order to use this knowledge, the $node_{n1}$ can update its routing table by adding $node_{n2}$ on it, to make the latter search faster.

To update the routing table, we give a variable *rank* to each item of routing table. The initial value of *rank* is r. After a successful search, which started at $node_{n1}$, and end at $node_{n2}$, the process of updating routing table of $node_{n1}$ is shown as follows:

1. Find out two items in the $node_{n1}$'s routing table, whose 'start' is M_1 and M_2, M_1 is larger than N_1, and M_2 is smaller than N_1.
2. $rank_{M_1}$ and $rank_{M_1}$ will be updated as follow rules:

$$rank_{M_1} = p \bullet (m-2) \bullet ((N_2 - M_1) / (M_2 - M_1)) \tag{2}$$

$$rank_{M_2} = p \bullet (m-2) \bullet ((M_2 - N_2) / (M_2 - M_1)) \tag{3}$$

Where parameter p is a nonnegative integer, depended on the category number and network size;

3. The other items of $node_{n1}$'s routing table, the rank value is updated by:

$$rank_i = rank_i - p \tag{4}$$

4. If $(rank_{M_1} + rank_{M_1}) / 3 \geq r$, need to carry out further processing which we called routing item division:
   (a) Add the $node_{n2}$ on the $node_{n1}$'s routing table, the $rank_{n2}$ set as r;
   (b) Find the item with a lowest rank, assuming that its rank is r';
   (c) Removed it from the routing table, and update $rank_{M_1}$ and $rank_{M_1}$ as follow rules:

$$rank_{M_1} = rank_{M_1} - (r - r') / 2 \tag{5}$$

$$rank_{M_2} = rank_{M_2} - (r - r') / 2 \tag{6}$$

2.7 Routing Table Sharing

The routing table will be stabilized after a period of running. If it meets our assuming, the routing table will reflect the semantic association between nodes. However, the

P2P networks is a rapidly changing network, if the convergence of the update algorithm costs too much time, the routing table cannot achieve stabilization before the network structure changing. So we propose a routing table sharing method to make this procedure faster.

When a node is running a routing item division procedure, it will send its new routing table item to adjacent nodes.

When a node receives the routing table item information of a neighbor node, e.g. $node_{n1}$ receives routing table $item_{node_{n2}}$ from its neighbor, then $node_{n1}$ will update its own routing table as follow:

1. Find out two items in the routing table, whose 'start' is M_1 and M_2, M_1 is larger than N_1, and M_2 is smaller than N_1;
2. $rank_{M_1}$ and $rank_{M_1}$ will be updated as follow rules:

$$rank_{M_1} = p \bullet q \bullet (m-2) \bullet ((N_2 - M_1)/(M_2 - M_1)) \tag{7}$$

$$rank_{M_2} = p \bullet q \bullet (m-2) \bullet ((M_2 - N_2)/(M_2 - M_1)) \tag{8}$$

Where parameter q is a nonnegative integer, depend on the distance of $node_{n1}$ and its neighbor;

3. The other items of $node_{n1}$'s routing table, the rank value is updated by:

$$rank_i = rank_i - p \bullet q \tag{9}$$

4. If $(rank_{M_1} + rank_{M_1})/3 \geq r$, need to carry out routing item division process too:
   (a) Add the $node_{n2}$ to $node_{n1}$'s routing table, the $rank_{n2}$ set as r;
   (b) Find the item with a lowest rank, assuming that its rank is r';
   (c) Removed it from the routing table, and update $rank_{M_1}$ and $rank_{M_1}$ as follow rules:

$$rank_{M_1} = rank_{M_1} - (r - r')/2 \tag{10}$$

$$rank_{M_2} = rank_{M_2} - (r - r')/2 \tag{11}$$

The essence of the method is to make use of the knowledge of neighboring nodes, enabling faster to routing table stabilization.

3 Experiment

We design a prototype system to verify the validity of the H-Chord algorithms. The size m of chord space is 8, that is, the chord space is 256, and there are N = 192

nodes, and 512 resources stored in these nodes. As we already discussed, the resource hash procedure depends on the domain knowledge, so we will only focus on the heuristic searching algorithm.

Suppose we use a resource classification algorithm to divide the resources into 64 categories, then the hash value of a resource is:

$$hash_{se}(resource) = concat(hash(cate), hash(resource)) \tag{12}$$

The outcome of $hash_{se}(resource)$ is 8bit, where the $hash(cate)$ is 6bit, and the $hash(resource)$ is 2bit.

In the experiment we add the192 nodes into the chord space at first, and then add 512resources to initial the chord ring. To make the problem more focus, we did not consider the probability of nodes or resources' joining and quitting.

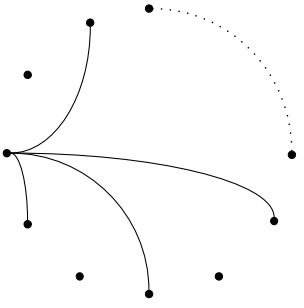


Fig. 4. Semantic map of 64 nodes and 128 edges

A series of search requests are needed to simulate the system operation. To correspond to the assumptions in this article, the resources search requests generated by the user will not independent, but semantically related. We generate the semantic map at first, denote a category by a point, and the relations between categories by the edges. A semantic map of 64 nodes and 128 edges is shown as Figure.5. With the above semantic map, search sequence of resources can be generated as follows:

If $resource_i \in N_i, whereN_i \in cate_i$, and $resource_{i+1} \in N_{i+1}, whereN_{i+1} \in cate_{i+1}$, then $cate_{i+1}$ is connected with $cate_i$. Suppose there are several $cate_j$ connected with $cate_i$, then:

$$P(cate_{i+1} = cate_j) = \frac{edge(cate_i, cate_j)}{\sum edge(cate_i, cate_x)} \tag{13}$$

3.1 Experiment

Based on the above prototype system, we generate a sequence of 2000 search request, test it on both Chord and H-Chord, write down its average steps at each 100 request, and the comparison is shown as figure 6.

We can get from the figure 6 that, at the beginning, H-Chord and Chord have almost the same results. After a while, the average length of H-Chord is decrease rapidly. Because at the running procedure, the node is updating its routing table, to fit the semantic relation of resources, which makes the search sequence much easier to locate its related resources.

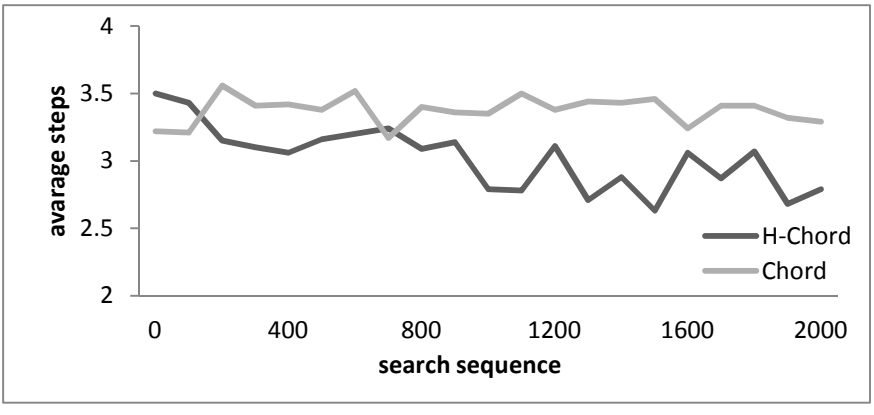


Fig. 5. Comparison of H-Chord to Chord

Furthermore, we test the impact of the network size to the algorithm efficiency. Three test networks are built, whose size is 7, 8 and 9, means these networks can contain 2^7, 2^8 and 2^9 nodes. We add about 70% nodes into these networks, other parameters are the same with the above experiment. We still tests the average steps, results are shown in the figure.8.It seems that no matter how big the network size is, the H-Chord could achieve stabilization after about 800 requests.

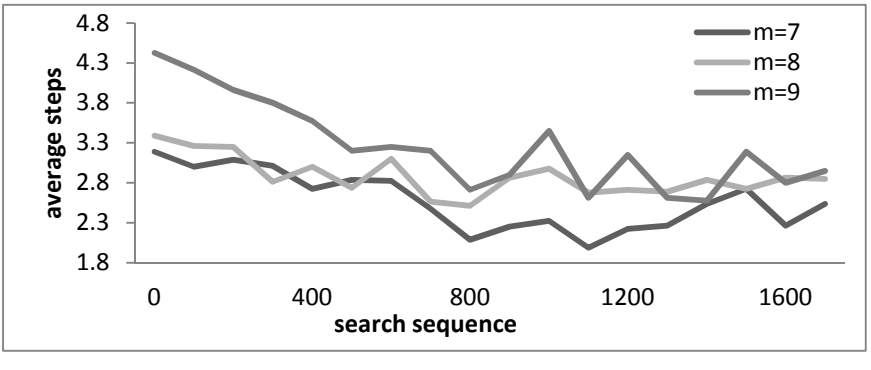


Fig. 6. Comparison of network size of H-Chord

4 Conclusion

This paper describes a novel modification of chord algorithm, called H-chord. The algorithm brings the semantic into the resources indexing procedure, and uses a

heuristic rules to establish semantic links between resources, making the resources locating requests are responded more quickly. A prototype system is built to verify our algorithm, and the results shows that H-Chord can achieve much better performance than the classic chord.

There are also lots of works to do with H-Chord in the future. Our work is based on the assumption that resources search request are always semantically related, the experiments are also built base on this assumption. The actual resource requests sequence should be study to find out how strong semantic relevance is. For application in different areas, design an efficient resource classification algorithm is also a tough job, which will be one of our major works in the future.

References

1. Parameswaran, M., Susarla, A., Whinston, A.B.: P2P networking: an information sharing alternative. Computer 34(7), 31–38 (2001)
2. Sarmady, S.: A Survey on Peer-to-Peer and DHT. Arxiv preprint arXiv:1006.4708 (2010)
3. Stoica, I., et al.: Chord: A scalable peer-to-peer lookup service for internet applications. ACM SIGCOMM Computer Communication Review 31(4), 149–160 (2001)
4. Ratnasamy, S., et al.: A scalable content-addressable network. ACM (2001)
5. Rowstron, A., Druschel, P.: Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems (2001)
6. Zhao, B.Y., Kubiatowicz, J., Joseph, A.D.: Tapestry: An infrastructure for fault-tolerant wide-area location and routing. Computer 74(11-20), 46 (2001)
7. Wang, X.H., Da Qing Zhang, T.G., Pung, H.K.: Ontology based context modeling and reasoning using OWL (2004)
8. Deerwester, S., et al.: Indexing by latent semantic analysis. Journal of the American society for information science 41(6), 391–407 (1990)
9. Khattak, A.M., et al.: Ontology Evolution: A Survey and Future Challenges. U-and E-Service, Science and Technology, pp. 68–75 (2009)

The Use of a Local Histogram Feature Vector of Classifying Diffuse Lung Opacities in High-Resolution Computed Tomography

Yoshihiro Mitani[1], Yusuke Fujita[2],
Naofumi Matsunaga[3], and Yoshihiko Hamamoto[1]

[1] Department of Intelligent System Engineering, Ube National College of Technology,
2-14-1,Tokiwadai, Ube, Yamaguchi, 755-8555, Japan
[2] Graduate School of Medicine, Yamaguchi University,
2-16-1, Tokiwadai, Ube, Yamaguchi, 755-8611, Japan
[3] Graduate School of Medicine, Yamaguchi University,
1-1-1, Minami-Kogushi, Ube, Yamaguchi, 755-8505, Japan

Abstract. The classification of diffuse lung opacities in high-resolution computed tomography(HRCT) images is an important step for developing a computer-aided diagnosis(CAD) system. In designing the CAD system for classifying diffuse lung opacities in HRCT images, a histogram feature has been shown to be effective. In order to improve further the classification performance of the CAD system, we have proposed the use of a local histogram feature vector. The experimental results show that the proposed method leads to clear improvement of the classification performance.

Keywords: HRCT images, CAD system, Diffuse lung opacities, Histogram feature vector, Classification.

1 Introduction

A large number of studies concerning the design of a computer aided diagnosis(CAD) system have been made [1][2]. The classification of diffuse lung opacities in high-resolution computed tomography(HRCT) images is an important step for developing the CAD system [3]. The HRCT image(512×512 pixels, 12 bits / pixel, 0.35 mm / pixel) were obtained from 70 patients with a variety of diffuse lung diseases. In HRCT images, we selected 500 regions of interest(ROIs) with 4 classes of typical diffuse lung opacities(reticular, nodular, ground-glass, and consolidation) and one normal class. Fig. 1 shows ROI images with 4 classes of typical diffuse lung opacities and one normal class. The diffuse lung opacities, which have a variety of patterns, tend to be distributed uniformly diffused in lungs. Thus, these may be considered as a kind of texture. It has been believed that a Gabor-filter [4] based feature approach is promising for texture recognition [5][6]. However, in our previous study [7] [8], a histogram feature based method has been shown to be more effective than a Gabor filter-based feature

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 343–350, 2012.

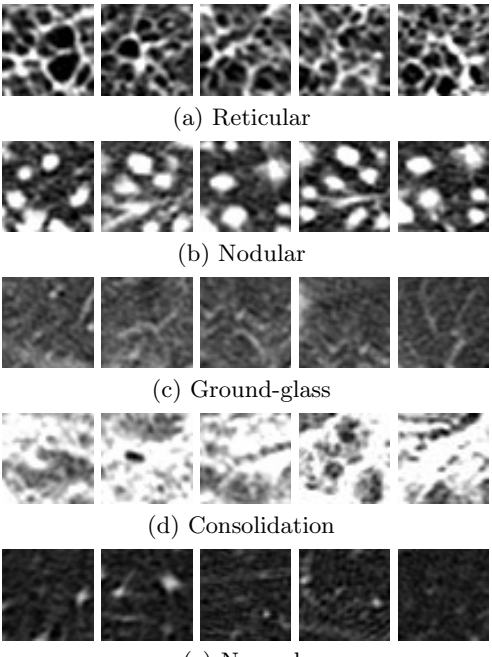
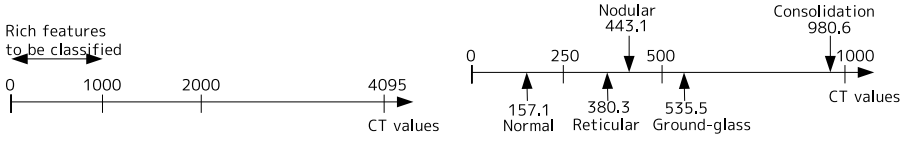


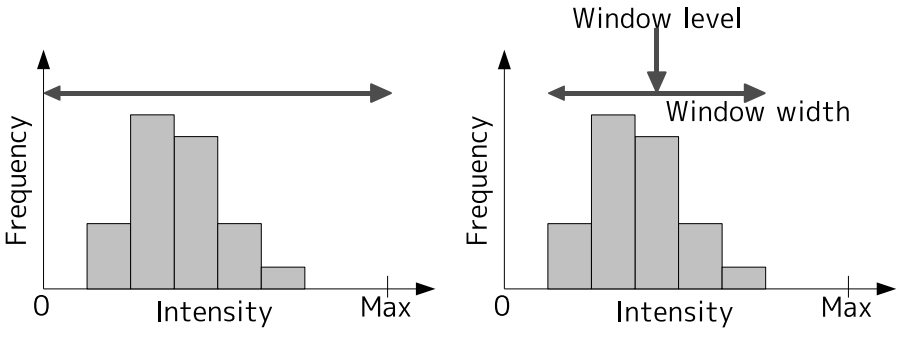
Fig. 1. ROI images with 4 classes of typical diffuse lung opacities and one normal class



(a) Overall of CT values. (b) Detail under 1,000 CT values.

Fig. 2. Distribution of average CT values with five classes of lung opacities

approach. The use of the CT values seems to reflect clearly in classifying lung opacities. Furthermore, we investigated each of average CT values for lung opacities with 100 ROIs per class, and found these comparatively clearly separable. In this paper, we use a 15×15 ROI size. Fig. 2 shows distribution of average CT values with five classes of lung opacities. Figs. 2 (a) and (b) represent the overall and the detail of the average CT values with five classes of lung opacities, respectively. In the HRCT images, the gray value of 12 bits per pixel varies from 0 to 4,095. The average CT values for normal, reticular, nodular, ground-glass, and consolidation opacities are 157.1, 380.3, 443.1, 535.5, and 980.6, respectively. The average values with five classes of lung opacities concentrate in the range under 1,000 CT values. In this range, there may be rich features to be classified. Therefore, we have considered making a local histogram in this range. It is known that in order clearly to examine findings of the HRCT images, a window level



(a) A conventional histogram. (b) A proposed histogram.

Fig. 3. Difference between the conventional and proposed histograms

and a window width are adjusted in display devices. In the proposed method, a window level and a window width are changed not for displaying the HRCT images but for classifying the lung opacities. In this paper, in order to improve further the classification performance of the CAD system for classifying diffuse lung opacities in HRCT images, we have proposed the use of a local histogram feature vector. Experimental results show that the proposed method leads to improvement of the classification performance.

2 A Proposed Method

First, we describe a conventional method using a histogram feature vector [7][8]. A histogram, which consists of x- and y- axes, is obtained from the gray ROI image. The x- and y- axes show the CT value and its frequency, respectively. The histogram feature vector is obtained by arranging the frequency of a corresponding CT value from the histogram. The ROI image has 4,096(=12 bits/pixel) gray levels. Thus, the number of bins, i.e., the dimensionality of the histogram feature vector is 4,096. Since the dimensionality is large, we quantize the gray ROI image. Here, we want to reduce the histogram bin size as much as possible. For simplicity, in quantizing the gray ROI image, the threshold levels are equally divided.

Second, we propose a local histogram feature vector approach. From the distribution of average CT values with five classes of lung opacities, we make a local histogram in the range under 1,000 CT values. Fig. 3 shows difference between conventional and proposed histograms. A conventional histogram covers all range of CT values in general. On the other hand, a proposed histogram covers the only selected range of CT values. The range is decided by setting a window. The window is defined by a window level and a window width as shown in fig. 3 (b). Note that when the window level and window width are 2,048 and 4,096, the proposed method is equal with the conventional one. Moreover, we can select the number of bins in the local histogram. In fig. 3 (b), the bin size

Table 1. Influence of the dimensionality of a conventional method on the error rate(%)

Dimensionality	Error rate
4	52.08 51.69, 52.46
8	16.80 16.43, 17.17
16	6.73 6.39, 7.08
32	5.21 4.95, 5.46
64	5.70 5.45, 5.95
128	6.34 6.06, 6.62
256	6.54 6.27, 6.81
512	6.87 6.62, 7.13
1,024	7.60 7.35, 7.86
2,048	8.91 8.62, 9.20
4,096	9.60 9.24, 9.96

Upper: average, Lower: 95% confidence interval

or dimensionality is 5. For simplicity, the bin width is equally divided. All the bin widths are the same.

3 Experimental Results

The error rate is the most effective measure of the classification performance of a CAD system. In order for the estimated error rate to be reliable in predicting the future classification performance of the CAD system, the training and test samples must be statistically independent [9]. In the error estimation literature, the holdout method has been widely used, since it maintains the statistical independence between the training and test sets [10]. The average and 95% confidence interval of the error rate were calculated as follows:

Step1. Divide 500 available samples into 250 training and 250 test samples at random.
Step2. Estimate the error rate by using the local mean-based classifier.
Step3. Repeat Steps 1 and 2, 100 times independently.
Step4. Compute the average and 95% confidence interval of the error rate.

The local mean-based classifier [11][12] is known to be robust for outliers. In the experiments, we used the local mean-based classifier with $r = 5$.

Table 2. Influence of the dimensionality of a proposed method on the error rate(%)

Dimensionality	Error rate	Window level Window width
4	5.78 5.52, 6.03	400 1,200
8	4.34 4.11, 4.56	500 1,700
16	4.46 4.25, 4.68	300 1,400
32	4.47 4.25, 4.68	300 1,300
64	4.58 4.34, 4.81	400 1,100
128	4.58 4.34, 4.81	500 1,100
256	4.50 4.28, 4.71	400 1,400

Upper: average, Lower: 95% confidence interval

Table 3. A confusion matrix of each of lung opacities to be misclassified on the error rate(%)

(a) A conventional method.

Test classes	Classes to be misclassified					
	Reticular	Nodular	Ground-glass	Consolidation	Normal	Total
Reticular	–	0.90	0.18	0.00	0.04	1.12
Nodular	0.47	–	0.23	0.18	1.02	1.90
Ground-glass	0.19	0.36	–	0.21	0.00	0.75
Consolidation	0.00	0.25	0.03	–	0.00	0.28
Normal	0.37	0.79	0.00	0.00	–	1.16

(b) A proposed method.

Test classes	Classes to be misclassified					
	Reticular	Nodular	Ground-glass	Consolidation	Normal	Total
Reticular	–	0.69	0.16	0.00	0.00	0.86
Nodular	0.25	–	0.06	0.17	0.77	1.25
Ground-glass	0.44	0.22	–	0.21	0.00	0.88
Consolidation	0.00	0.23	0.01	–	0.00	0.25
Normal	0.24	0.87	0.00	0.00	–	1.11

First, we show the performance of a conventional method. The error rate depends on the quantization level of the gray ROI image, i.e., dimensionality. The dimensionality was examined among $\{2^m | m = 2, 3, \cdots, 12\}$. Table 1 shows the influence of the dimensionality of a conventional method on the error rate. When the dimensionality is 4,096, the average error rate gives 9.60%. On the

Table 4. Influence of a window level and a window width of the proposed method on the error rate(%)

Window width	Window level				
	300	400	500	600	700
400	11.86 11.58, 12.15	12.55 12.15, 12.95	12.75 12.37, 13.14	10.79 10.51, 11.07	10.68 10.36, 11.00
500	9.33 9.05, 9.62	9.60 9.26, 9.95	10.26 9.93, 10.58	9.78 9.48, 10.07	9.69 9.42, 9.95
600	7.29 7.00, 7.58	8.06 7.76, 8.36	8.24 7.96, 8.53	8.66 8.36, 8.95	9.53 9.27, 9.80
700	6.58 6.33, 6.84	5.99 5.72, 6.29	7.30 7.02, 7.58	7.39 7.08, 7.71	8.42 8.12, 8.73
800	6.06 5.79, 6.33	4.77 4.54, 5.00	5.88 5.65, 6.11	7.18 6.87, 7.48	8.14 7.84, 8.73
900	6.31 6.04, 6.58	5.30 5.05, 5.55	5.32 5.04, 5.60	5.97 5.71, 6.23	8.33 8.03, 8.63
1,000	5.78 5.53, 6.03	4.85 4.61, 5.09	4.81 4.56, 5.06	5.89 5.60, 6.18	7.44 7.14, 7.73
1,100	5.04 4.80, 5.29	4.92 4.68, 5.16	4.77 4.52, 5.01	5.76 5.50, 6.01	7.02 6.73, 7.32
1,200	5.08 4.84, 5.31	4.43 4.17, 4.68	5.05 4.82, 5.28	5.03 4.75, 5.30	6.97 6.68, 7.32
1,300	4.73 4.50, 4.96	4.55 4.30, 4.80	5.66 5.37, 5.94	4.83 4.57, 5.09	6.58 6.30, 6.85
1,400	4.97 4.74, 5.20	4.84 4.58, 5.09	5.47 5.23, 5.72	4.97 4.71, 5.23	5.90 5.61, 6.19
1,500	4.79 4.55, 5.05	5.07 4.81, 5.34	5.03 4.82, 5.25	4.74 4.48, 4.99	5.41 5.17, 5.66
1,600	4.41 4.18, 4.65	6.55 6.24, 6.85	4.62 4.39, 4.85	6.70 6.42, 6.98	4.56 4.33, 4.78
1,700	4.36 4.14, 4.58	6.49 6.20, 6.78	4.34 4.11, 4.56	6.05 5.77, 6.33	4.63 4.38, 4.88
1,800	4.35 4.13, 4.57	6.66 6.36, 6.96	4.54 4.32, 4.75	5.43 5.16, 5.70	4.40 4.15, 4.65

Upper: average, Lower: 95% confidence interval

other hand, when the dimensionality is 32, the average error rate shows a minimum, 5.21%. From the results, we recommend that the dimensionality of the conventional method is 32.

Second, we show the performance of a proposed method. The error rate depends on the window level, the window width, and the number of bins or dimensionality. We changed a window level and a window width into 300-700 and 400-1,800, respectively. The dimensionality was examined among $\{2^m | m = 2, 3, \cdots, 8\}$. Table 2 shows the influence of the dimensionality of a proposed method on the error rate. The average error rates show the smallest values at each of the dimensionality. In the most right column, the optimal window level and window width are also shown. When the dimensionality is 8, the average

error rate gives a minimum value, 4.34%. In addition, 95% confidence intervals of the error rates between the proposed method and conventional method do not overlap. This means that the proposed method outperforms the conventional one. Furthermore, the proposed method is superior to the conventional one in terms of the dimensionality.

When the dimensionality was 8, we also examined the influence of a window level and a window width of the proposed method. Table 4 shows the influence of a window level and a window width of the proposed method on the error rate. From the results, the narrow window width seems to perform poorly. As we expected, the range under 1,000 CT values may have rich features.

Furthermore, we investigated the detail of causes of the error rates. Table 3 shows a confusion matrix of each of lung opacities to be misclassified on the error rate. Table 3 (a) shows the result of the conventional method when the dimensionality is 32. Table 3 (b) is the result of the proposed method when a window level, window width, and dimensionality are 500, 1,700, and 8. From the results, by the proposed method, the error rates of the reticular, nodular, consolidation, and normal classes improve 0.26, 0.65, 0.03, and 0.05(%), respectively. Above all, the effectiveness of nodular-class classification is large. On the other hand, the classification of the ground-glass seems to perform slightly poorly.

4 Conclusions

In order to improve the classification performance of the CAD system, we have proposed the use of a local histogram feature vector. The experimental results show that the proposed method is one of effective means for improving the CAD system. In this paper, in order to classify typical diffuse lung opacities, we have focused on a local histogram feature vector. In the future, to improve the classification performance, we consider using other types of feature vector. And we explore the use of a combined feature vector.

References

1. Duncan, J.S., Ayache, N.: Medical image analysis: Progress over two decades and the challenges ahead. IEEE Trans. PAMI-22(1), 85–106 (2000)
2. Doi, K.: Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. Computerized Medical Imaging and Graphics 31, 198–211 (2007)
3. Sluimer, I., Schilham, A., Prokop, M., van Ginneken, B.: Computer analysis of computed tomography scans of the lung: A survey. IEEE Trans., Medical Imaging 25(4), 385–405 (2006)
4. Gabor, D.: Theory of communication. J. Inst. Elect. Engr. 93, 429–457 (1946)
5. Jain, A.K., Farrokhnia, F.: Unsupervised texture segmentation using Gabor filters. Pattern Recognition 24, 1167–1186 (1991)
6. Turner, M.R.: Texture discrimination by Gabor functions. Biol. Cybernet. 55, 71–82 (1986)

7. Mitani, Y., Yasuda, H., Kido, S., Ueda, K., Matsunaga, N., Hamamoto, Y.: Combining the Gabor and histogram features for classifying diffuse lung opacities in thin-section computed tomography. In: Proc. 16th Int. Conf. Pattern Recognition, vol. I, pp. 53–56 (2002)
8. Mitani, Y., Yasuda, H., Kido, S., Ueda, K., Matsunaga, N., Hamamoto, Y.: Combined features for classifying diffuse lung opacities in thin-section computed tomography images. In: Damiani, E., et al. (eds.) Knowledge-Based Intelligent Information Engineering Systems & Allied Technologies. Frontiers in Artificial Intelligence and Applications, vol. 82, PartI, pp. 121–125. IOS Press (2002)
9. Devijver, P.A., Kittler, J.: Pattern recognition: A statistical approach. Prentice / Hall (1982)
10. Fukunaga, K.: Introduction to statistical pattern recognition, 2nd edn. Academic Press (1990)
11. Mitani, Y., Hamamoto, Y.: Classifier design based on the use of nearest neighbor samples. In: Proc. 15th Int. Conf. Pattern Recognition, vol. 2, pp. 773–776 (2000)
12. Mitani, Y., Hamamoto, Y.: A local mean-based nonparametric classifier. Pattern Recognition Letters 27(10), 1151–1159 (2006)

SOM^2CE: Double Self-Organizing Map Based Cluster Ensemble Framework and its Application in Cancer Gene Expression Profiles

Zhiwen Yu[1,2], Hantao Chen[1], Jane You[2], Le Li[1], and Guoqiang Han[1]

[1] School of Computer Science and Engineering,
South China University of Technology
zhwyu@scut.edu.cn
[2] Department of Computing,
Hong Kong Polytechnic University
csyjia@comp.polyu.edu.hk

Abstract. Though there exist a lot of cluster ensemble approaches, few of them consider how to degrade the effect of noisy attributes in the dataset. In the paper, we propose a new cluster ensemble framework, named as double self-organizing map based cluster ensemble (SOM^2CE) to perform clustering on noisy datasets. SOM^2CE incorporates the self-organizing map (SOM) twice into the ensemble framework to discovery the underlying structure of noisy datasets, which applies SOM to perform clustering not only on the sample dimension, but also on the attribute dimension. SOM^2CE also adopts the normalized cut algorithm to partition the consensus matrix constructed from multiple clustering solutions, and obtain the final results. Experiments on both synthetic datasets and cancer gene expression profiles illustrate that the proposed approach not only achieves good performance on synthetic datasets and cancer gene expression profiles, but also outperforms most of the existing approaches in the process of clustering gene expression profiles.

Keywords: Cluster ensemble, self-organizing map, cancer data.

1 Introduction

Recently, cluster ensemble is gaining more and more attention as one of the important research directions in the ensemble learning area. Cluster ensemble improves accuracy, stableness and robustness when compared with single clustering approach. A single clustering solution only identifies part of relationships among data points, while the cluster ensemble approaches can discover a relatively comprehensive cognition of the general structure of dataset. Since the cluster ensemble approach can decompose a complex problem into multiple subproblems which is easier to understand and solve. In general, cluster ensemble approaches are able to enhance the accuracy, stability and robustness of final results by combining different patterns obtained from the same origin dataset and

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 351–360, 2012.

achieve better performance when compared with a single clustering algorithm. Due to the advantages of ensemble methods, they have been applied extensively in different areas, such as bioinformatics, data mining, pattern recognition, and so on. Generally, the cluster ensemble approaches are divided into two stages: generation of clustering solutions and summarization of clustering solutions [1]. The objective of the first stage is to generate clustering solutions as different as possible, while the objective of the second stage is to obtain better results by summarizing the clustering solutions efficiently. The approaches in the first stage are as follows: applying different clustering algorithm, using one algorithm with different parameters [2], choosing different sets (subset or superset) of features [3] and selecting different sets (subset or superset) of sample points [4,5]. The first two approaches pay attention to clustering algorithms, while the last two approaches focus on datasets which are part of data transformation techniques [6]. The approaches in the second stage contain the voting approach [7], the consensus matrix based methods [8,9,10], the hypergraph method[4], and so on. Unfortunately, it is difficult to clearly identify the relationship among clusters from different clustering solutions in the second stage. It is still a challenge for the cluster ensemble approach.

Although there are many cluster ensemble methods today, most of them only achieve good results on 'clean' datasets, few of them consider how to obtain good performance on datasets with noisy attributes. The noisy datasets may sharply degrade the performance of traditional cluster ensemble approaches. For the sake of dealing with the challenges of the datasets with noisy attributes, we proposed the double self-organizing map based cluster ensemble, called as SOM^2CE, which integrates double self-organizing map (SOM)[11] and the normalized cut (NCUT) [12] into the ensemble framework. SOM^2CE applies SOM on both the sampling dimension and the attribute dimension, which decreases the risk of being impacted by noisy attributes and noisy samples. Experimental results show SOM^2CE works well in cancer gene expression profiles.

The remainder of the paper is organized as follows. Part II describes the frame of double self-organizing map based cluster ensemble. Part III evaluates the performance of double self-organizing map based cluster ensemble. Part IV draws a conclusion.

2 Double Self-Organizing Map Based Cluster Ensemble Framework

Figure 1 shows the overview of double self-organizing map based cluster ensemble framework SOM^2CE. SOM^2CE first adopts the self-organizing map approach (SOM) [11] to perform clustering on the attribute dimension, and obtains a set of datasets by adjusting the number of neurons in SOM. Then, it applies SOM again to cluster samples in the sample dimension, and generates a set of clustering solution. In the following, SOM^2CE constructs a consensus matrix, and adopts the normalized cut algorithm [12] to summarize the consensus matrix and obtain final results.

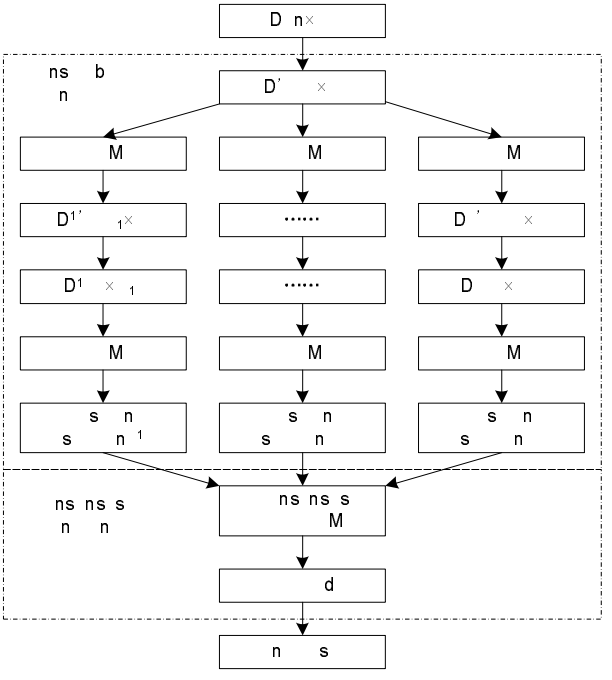


Fig. 1. Double self-organizing map based cluster ensemble framework SOM^2CE

Specifically, given a dataset $\mathbf{D}$ which consists of n samples and m attributes, SOM^2CE first transforms the original data matrix $\mathbf{D}$ to an $m \times n$ new data matrix $\mathbf{D}'$ by the transpose operator as follows:

$$\mathbf{D}^{'} = \mathbf{D}^{T} \tag{1}$$

Then, it applies SOM to perform clustering on the new data matrix $\mathbf{D}' = \{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_m\}$. A self-organizing map [11] consists of two layers: the input layer which contains a set of input vectors $\mathbf{f}_j$, and the output layer which includes a set of neurons o_h with associated weight vectors $\mathbf{c}_h$ in a low dimensional grid (where $h \in \{1, ..., m_b\}$, m_b is the number of neurons).

The learning process in SOM consists of the following steps: (i) search for the most representative weight vector to minimize the sum of the distance metric between the input vector and the weight vector and (ii) assign the input vector to the corresponding neuron. The learning algorithm first initials all the weight vectors, and considers the input vectors $\mathbf{f}_j$ one by one. The input vector $\mathbf{f}_j$ is assigned to the best matching neuron o_h, which satisfies the following condition:

$$h^* = arg\ min_{1 \le h \le m} \phi(\mathbf{f}_j, \mathbf{c}_h) \tag{2}$$

$$\phi(\mathbf{f}_j, \mathbf{c}_h) = \sqrt{\sum_{l=1}^{n} (f_{jl} - c_{hl})^2} \tag{3}$$

Then, the learning algorithm updates the weight vector $\mathbf{c}_{h^*}$ associated with the winner neuron o_{h^*} and the weight vectors of its corresponding neighborhood as follows

$$\mathbf{c}_h(t+1) = \mathbf{c}_h(t) + \beta(t)\varphi(\mathbf{P}_h, \mathbf{P}_{h^*})(\mathbf{f}_j - \mathbf{c}_h(t)) \tag{4}$$

$$where \quad \beta(t) = e^{-\alpha_1 t} + \alpha_2 \tag{5}$$

$$\varphi(\mathbf{P}_h, \mathbf{P}_{h^*}) = e^{-\frac{\|\mathbf{P}_h - \mathbf{P}_{h^*}\|^2}{2\sigma^2(t)}} \tag{6}$$

where$t$ denotes the tth iteration, $\beta(t)$ is the time-dependent learning rate, α_1 and α_2 are the parameters which control the decay of $\beta(t)$, $\mathbf{P}_h$ and $\mathbf{P}_{h^*}$ denote the grid coordinates of the h-th neuron and the h^*-th neuron respectively, $\varphi(\mathbf{P}_h, \mathbf{P}_{h^*})$ is a smooth neighborhood function, and σ is the maximum neighborhood range.

After the learning process of SOM, all the weight vectors are determined. A new dataset $D^{b'}$ is formed based on a set of weight vectors $\{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_{m_b}\}$. By adjusting the number of weight vectors in the learning process, SOM^2CE generates a set of new datasets $D^{1'}, D^{2'}, ..., D^{B'}$. The data matrix $D^{b'}$ (where $b \in \{1, ..., B\}$) with $m_b \times n$ entries is further converted to the $n \times m_b$ data matrix D^b by the transpose operator. SOM^2CE adopts SOM to perform clustering on a set of data matrices $D^1, D^2, ..., D^B$ again, and obtains a set of clustering solutions $I^1, I^2, ..., I^B$ as show in Figure 1.

In the following, SOM^2CE constructs a consensus matrix CM by combining all the adjacency matrices $\mathbf{M}^b$ obtained from a set of clustering solutions I^b as follows:

$$\mathbf{CM} = \frac{1}{B}\sum_{b=1}^{B}\mathbf{M}^b \tag{7}$$

Finally, SOM^2CE adopts the normalized cut algorithm (Ncut) [12] to partition the consensus matrix CM, and obtains the final results. Ncut first constructs a graph $\Delta = (\mathbf{D}, \mathbf{CM})$, where $\mathbf{D}$ is the vertex set which consists a set of data samples $\{\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_n\}$, and $\mathbf{CM}$ is the edge set which are the entries cm_{ij} in the consensus matrix reflecting a set of the similarity values between pairs of samples $\mathbf{s}_i$ and $\mathbf{s}_j$. The objective function $\Delta(\mathbf{U}, \mathbf{V})$ of Ncut is used to maximize the association within the cluster and minimize the disassociation between the clusters at the same time, which is defined as follows:

$$\Delta(\mathbf{U}, \mathbf{V}) = \frac{\Lambda(\mathbf{U}, \mathbf{V})}{\Upsilon(\mathbf{U}, \mathbf{D})} + \frac{\Lambda(\mathbf{U}, \mathbf{V})}{\Upsilon(\mathbf{V}, \mathbf{D})} \tag{8}$$

$$\Lambda(\mathbf{U}, \mathbf{V}) = \sum_{\mathbf{d}_i \in \mathbf{U}, \mathbf{d}_j \in \mathbf{V}} cm_{ij} \tag{9}$$

$$\Upsilon(\mathbf{U},\mathbf{D}) = \sum_{\mathbf{d}_i \in \mathbf{U}, \mathbf{d}_h \in \mathbf{D}} cm_{ih} \tag{10}$$

where $\Delta(\mathbf{U},\mathbf{V})$ is a disassociation measure between $\mathbf{U}$ and $\mathbf{V}$, cm_{ij} is the value of the entry of $\mathbf{CM}$, which is the weight of the edge between the vertices $\mathbf{d}_i$ and $\mathbf{d}_j$. The normalized cut problem can be converted to an optimization problem, and an approximate solution in the real value domain can be obtained by solving the generalized eigenvalue system. SOM^2CE recursively partition the consensus matrix and obtains the final results with the help of Ncut.

3 Experiment

3.1 Data Set and Experimental Setting

The performance of SOM^2CE is evaluated by three synthetic datasets in Table 1 and three cancer gene expression profiles in Table 2. All the synthetic datasets in Table 1 contains 50 noisy attributes among 300 attributes. As illustrated in Table 2, the breast cancer dataset assigns 98 samples into three distinct cancer types: BRCA1-mutation-positive samples, BRCA2-mutation-positive samples, and Sporadic samples. The lung cancer datasets includes 139 adenocarcinoma samples (AD), 6 small-cell lung cancer samples (SCLC), 20 pulmonary carcinoids samples (COID), 21 squamous cell lung carcinomas samples (SQ), and 17 normal tissue samples. Cancer samples in the St.Jude leukemia dataset come from pediatric patients with acute leukemia, which are divided into 6 leukemia subtypes: 43 T-lineage ALL samples, 27 E2A-PBX1 samples, 15 BCR-ABL samples, 79 TEL-AML1 samples, 20 MLL rearrangements samples and 64 *hyperdiploid* > 50 chromosomes samples.

Table 1. The outline of real datasets

Dataset	K	d	n
Synthetic1	3	300	120
Synthetic2	6	300	240
Synthetic3	5	300	250

We adopt the rand index (RI) and the purity (PU) to evaluate the quality of clusters. RI is calculated as follows:

$$RI = \frac{\tau_1 + \tau_4}{\tau_1 + \tau_2 + \tau_3 + \tau_4} \tag{11}$$

where τ_1 denotes the number of pairs of data samples $\mathbf{d}_i$ and $\mathbf{d}_j$ which are assigned to the same cluster in the ground truth partition P and the predicted partition P' at the same time. τ_2 denotes the number of pairs of data samples $\mathbf{d}_i$ and $\mathbf{d}_j$ which belong to different clusters in P and are assigned to the same cluster in P'. τ_3 denotes the number of pairs of data samples $\mathbf{d}_i$ and $\mathbf{d}_j$ which belong to the same cluster in P and are assigned to different clusters in P'. τ_4

Table 2. The summary of cancer gene expression profiles

Dataset	K	d	n
Breast cancer	3	98	1213
Lung cancer	5	203	1543
St. Jude leukemia	6	248	985

denotes the number of pairs of data samples $\mathbf{d}_i$ and $\mathbf{d}_j$ which belong to different clusters in P and are assigned to different clusters in P'.

PU is defined as follows:

$$PU(P, P') = \frac{1}{n} \sum_{j=1}^{k} \max_{h \in \{1,...,k'\}} |C_j \cap C'_h| \tag{12}$$

where n is the number of data samples, $P = \{C_1, C_2, ..., C_k\}$, and $P' = \{C'_1, C'_2, ..., C'_{k'}\}$.

The performances of AP^2CE and other approaches are measured by the average value and the corresponding standard deviation of RI and PU respectively after performing 10 times.

In the following experiments, we first explore the effect of components in SOM^2CE. Then, SOM^2CE is compared with single clustering algorithms and other cluster ensemble approaches in both synthetic datasets and cancer gene expression profiles.

3.2 The Effect of Components

In order to explore the effect of components in SOM^2CE, we compare SOM^2CE with $SOMCE$ which only performs clustering on the sample dimension instead of both the attribute dimension and the sample dimension. Table 3 shows the performance of SOM^2CE and $SOMCE$ with respect to RI on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset respectively. SOM^2CE outperforms $SOMCE$, and achieves better results. The possible reason is that SOM^2CE adopts SOM to perform clustering on the attribute dimension and selects a set of weight vectors as new attributes, which is able to degrade the effect of noisy attributes and improve the performance of SOM^2CE. If one component is missing, the ability of SOM^2CE to obtain good results will be compromised.

3.3 Comparison of Single Clustering Algorithm

SOM^2CE is compared with single clustering algorithms SOM and K-means in terms of RI and PU on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset. Table 4 illustrates the comparison results of SOM^2CE,

Table 3. The performance of SOM^2CE and $SOMCE$ with respect to RI on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset (The values in boldface indicate the better results.)

(a) The average value and the standard deviation of RI			
Approach	Synthetic1	Synthetic2	Synthetic3
SOM^2CE	**1** (**0**)	**1** (**0**)	**1** (**0**)
$SOMCE$	0.8938 (0.1170)	0.9364 (0.1145)	0.9589 (0.0577)
Approach	Breast	Lung	St. Jude leukemia
SOM^2CE	**0.7684** (**0.0407**)	**0.7319** (**0.0154**)	**0.9864** (**0.0026**)
$SOMCE$	0.6938 (0.0776)	0.6467 (0.0463)	0.9162 (0.0736)

SOM and K-means. The results obtained by SOM^2CE are better than those obtained by SOM and K-means on most of datasets. The possible reason is that SOM^2CE is able to integrate multiple clustering solutions into the ensemble framework to provide more accurate, stable and robust results.

3.4 Comparison of Cluster Ensemble Approaches

In the following experiments, SOM^2CE is compared with other cluster ensemble algorithms, which contains the cluster ensemble approach based on random subspace and self-organizing map (RS-SOMCE), the cluster ensemble approach based on random subspace and k-means (RS-KMCE), the double K-means based cluster ensemble approach(KM^2CE).

Figure 2 and Figure 3 illustrated the results obtained by SOM^2CE, RS-SOMCE, RS-KMCE and KM^2CE with respect to RI and PU on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset respectively. As show in Figure 2 and Figure 3, SOM^2CE, RS-SOMCE and RS-KMCE achieve good performance on the Synthetic1 dataset, the Synthetic2 dataset and the Synthetic3 dataset. The possible reason is that the random subspace approach is able to reduce the effect of noisy attributes and improve the performance of the ensemble approaches. The results obtained by SOM^2CE is better than those obtained by RS-SOMCE, RS-KMCE and KM^2CE on the the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset respectively. The possible reason is that SOM^2CE not only can degrade the effect of noisy attributes, but also can increase the diversity of the ensemble by adjusting the size of grid in SOM, which will improve the accuracy and stableness of the cluster ensemble approach.

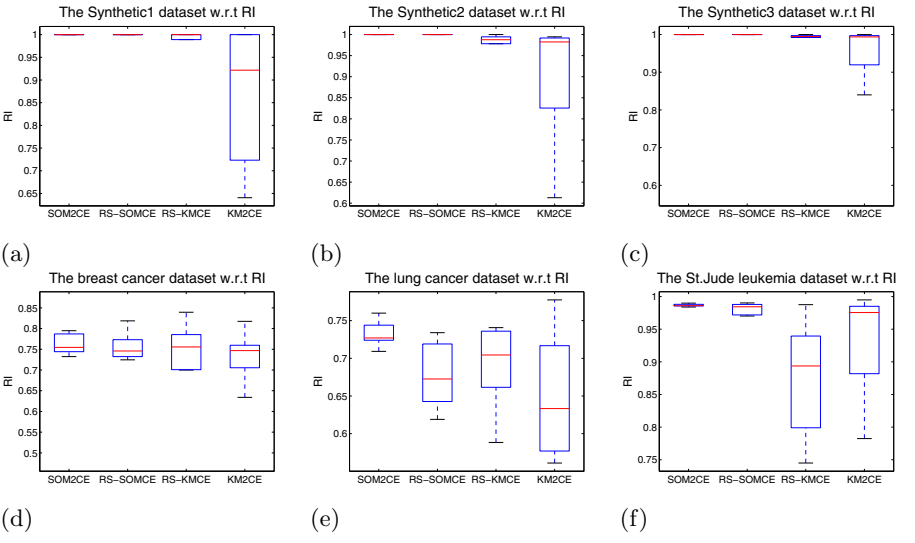


Fig. 2. The performance of SOM^2CE, RS-SOMCE, RS-KMCE and KM^2CE with respect to RI on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset respectively

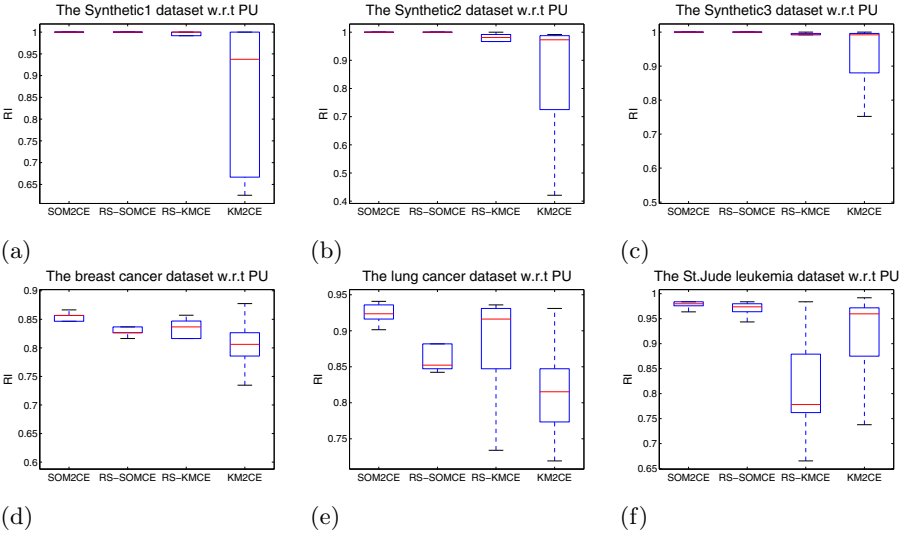


Fig. 3. The performance of SOM^2CE, RS-SOMCE, RS-KMCE and KM^2CE with respect to PU on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset respectively

Table 4. The performance of SOM^2CE, SOM and $K-means$ with respect to RI and PU on the Synthetic1 dataset, the Synthetic2 dataset, the Synthetic3 dataset, the breast cancer dataset, the lung cancer dataset and the St. Jude leukemia dataset

(a) The average value and the standard deviation of RI			
Approach	Synthetic1	Synthetic2	Synthetic3
SOM^2CE	**1** (**0**)	**1** (**0**)	**1** (**0**)
SOM	0.9667 (0.1054)	0.8845 (0.1282)	0.9857 (0.0183)
$K-means$	0.8571 (0.0545)	0.8642 (0.1245)	0.8656 (0.1190)
Approach	Breast	Lung	St. Jude leukemia
SOM^2CE	**0.7684** (**0.0407**)	**0.7319** (**0.0154**)	**0.9864** (**0.0026**)
SOM	0.7417 (0.0480)	0.6253 (0.0410)	0.9723 (0.0222)
$K-means$	0.7265 (0.0448)	0.6468 (0.0835)	0.9228 (0.0424)
(b) The average value and the standard deviation of PU			
Approach	Synthetic1	Synthetic2	Synthetic3
SOM^2CE	**1** (**0**)	**1** (**0**)	**1** (**0**)
SOM	0.9667 (0.1054)	0.7333 (0.2629)	0.9760 (0.0506)
$K-means$	0.9333 0.1405)	0.7667 (0.2250)	0.8480 (0.2047)
Approach	Breast	Lung	St. Jude leukemia
SOM^2CE	0.8578 (0.0123)	**0.9246** (**0.0129**)	**0.9774** (**0.0069**)
SOM	0.8429 (0.0216)	0.8517 (0.0805)	0.9552 (0.0482)
$K-means$	**0.8592** (**0.0280**)	0.8892 (0.0681)	0.9335 (0.0838)

4 Conclusion

In this paper, we explore the problem of performing clustering on the attribute dimension with noisy attributes by the cluster ensemble approach. The major contribution of the paper is a new cluster ensemble framework, named as double self-organizing map based cluster ensemble framework (SOM^2CE), which integrates the self-organizing map twice in the ensemble framework for discovering the underlying structure of the datasets with noisy attributes. To our knowledge, this is the first time that SOM in the cluster ensemble framework is applied to perform clustering on both the attribute dimension and the sample dimension at the same time. The experiments in both synthetic datasets and cancer gene expression profiles demonstrate that SOM^2CE works well.

Acknowledgments. The work described in this paper was partially supported by grants from National Natural Science Foundation of China (Project No. 61070090, 61003174, 60973083 and 61170080), a grant from NSFC-Guangdong Joint Fund (Project No. U1035004 and U1135004), a grant from program for New Century Excellent Talents in University (Project No. NCET-11-0165), a grant from the Fundamental Research Funds for the Central Universities (Project No. 2012ZZ0062 and 2012ZZ0064), a grant from Natural Science Foundation of Guangdong Province, China (Project No. S2011010000264 and 10451064101004233), a grant from Guangdong Major Science and Technology Project (Project No.2010A080402005), a grant from Doctoral Fund of Ministry of Education of China (Project No. 20100172120031), the grants from Hong Kong Government under its GRF scheme (5341/08E and 5366/09E) and the Hong Kong Polytechnic University Postdoctoral Fellowship (G-YX5D).

References

1. Minaei-Bidgoli, B., Topchy, A.P., Punch, W.F.: A Comparison of Resampling Methods for Clustering Ensembles. In: IJCAI 2004, pp. 939–945 (2004)
2. Weng, F., Jiang, Q., Shi, L., Wu, N.: An Intrusion Detection System Based on the Clustering Ensemble. In: 2007 IEEE International Workshop on Anti-Counterfeiting, Security, Identification, pp. 121–124 (2007)
3. Fern, X.Z., Brodley, C.E.: Random Projection for High Dimensional Data Clustering: A Cluster Ensemble Approach. In: ICML 2003, pp. 186–193 (2003)
4. Yang, Y., Kamel, M., Jin, F.: Clustering Ensemble Using ANT and ART. In: Swarm Intelligence in Data Mining 2006, pp. 243–264 (2006)
5. Topchy, A.P., Bidgoli, B.M., Jain, A.K., Punch, W.F.: Adaptive Clustering Ensembles. In: ICPR (1) 2004, pp. 272–275 (2004)
6. Yu, Z., Wong, H.-S., You, J., Yu, G., Han, G.: Hybrid Cluster Ensemble Framework based on the Random Combination of Data Transformation Operators. Pattern Recognition (2011) (to appear)
7. Dimitriadou, E., Weingessel, A., Hornik, K.: Voting-Merging: An Ensemble Method for Clustering. In: Dorffner, G., Bischof, H., Hornik, K. (eds.) ICANN 2001. LNCS, vol. 2130, pp. 217–224. Springer, Heidelberg (2001)
8. Topchy, A.P., Jain, A.K., Punch, W.F.: Clustering Ensembles: Models of Consensus and Weak Partitions. IEEE Trans. Pattern Anal. Mach. Intell. 27(12), 1866–1881 (2005)
9. Faceli, K., Ferreira de Carvalho, A.C.P.L., Pereira de Souto, M.C.: Multi-objective clustering ensemble. Int. J. Hybrid Intell. Syst. 4(3), 145–156 (2007)
10. Li, T., Ding, C.H.Q.: Weighted Consensus Clustering. In: SDM 2008, pp. 798–809 (2008)
11. Kohonen, T.: The Self-Organizing Map. Proceedings of the IEEE 78(9) (1990)
12. Jianbo, S., Jitendra, M.: Normalized Cuts and Image Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 22(8), 888–905 (2000)

A Context-Aware Enterprise Search Engine for Aviation

Hao Wang, Qingwei Liu, Tangjian Deng, and Ling Feng

Dept. of Computer Science & Technology, Tsinghua University, Beijing, China
{wanghaomails,liuqingwei2019,dengtangjian}@gmail.com,
fengling@tsinghua.edu.cn

Abstract. Compared to conventional internet search engines, the complexity of enterprise information space raises a number of challenges to enterprise search engines in the nature of unstructured search contents, task-relevance, dynamic result presentation, and multiple languages. We show how we tackle the above challenges through the development of an aviation-oriented enterprise search engine for English-Chinese bilingual MRO (**M**aintenance, **R**epair, and **O**verhaul) task cards, called TaskCardFinder. TaskCardFinder enables MRO service planners and technicians to quickly find out a list of bilingual MRO task cards which are relevant to a specific service order/request coming from airlines. The TaskCardFinder demonstrates several novel context-aware features specifically designed for aviation industry, including: topic-specific search assistance, relevant keywords suggestion, context-aware preferences support, recall-based search by time according to user's search history and/or task card viewing history, and MRO-service-relevant task cards recommendation. In addition to that, TaskCardFinder is also a bilingual (Chinese and English) search engine. It allows users to use either Chinese or English or the combination of both to search relevant technical data in both English and Chinese.

Keywords: context-aware, aviation MRO cards, bilingual search.

1 Introduction

Enterprise search is an essential part of business intelligence technologies [10]. Effective and efficient search over enterprise's massive data sources has become increasingly critical for today's enterprise information workers in carrying out their duties. However, the complexity of enterprise information space raises a number of challenges to enterprise search engines. [9,6] discussed the differences between enterprise search engines and conventional internet search engines in the nature of content, user behavior, and economic motivations, which could be summarized as follows. First, in an enterprise, a majority of information to be searched is unstructured and possibly in multi-languages. Second, search task is highly task-relevant, where search context (like user role, activity, company's regulation, etc.) shall be taken into account for personalized search. Third,

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 361–370, 2012.

domain-specific guided navigation search and refinement are desirable. Fourth, many times users may know and have previously seen the wanted information, where searching is recall-based. Fifth, the search result may be small containing the *right* information rather than the *best matching* one. Sixth, result presentation shall include summary, category, and aggregate information to enhance the usability of the search result. These challenges have led to a formidable problem but also mean enormous potential benefit.

In this paper, we show how we tackle the above challenges through the development of an aviation-oriented enterprise search engine for bilingual (English-Chinese) MRO task cards, called TaskCardFinder. The remainder of the report is organized as follows. In Section 2, we review some related work on search engines, and detail the application domain of TaskCardFinder in Section 3. The framework of context-aware searching is described in Section 4. We describe the design and implementation of a context-aware searching tool called TaskCardFinder in Section 5, and evaluate its performance in Section 6. We conclude the paper in Section 7.

2 Related Work

The first search engine in the world is called archie [11], a tool used for searching web information. From then on, search engine techniques have a sharp development. Currently, search engines can be mainly classified into five types, namely, crawler-based (traditional, common) engines, directories engines, meta search engines, vertical search engines, and pay per click search engines. Most of the commercial engines nowadays are based on crawler search. They collect information from web sites, index the obtained information, and provide search interfaces to search users, such as the most popular search engine Google, Yahoo, Chinese search engine Baidu, Bing, Ask, and AOL. Directories engines are not real search engines in a strict way, and they only just classify different web sites in hierarchical levels by humans, such as Google directories or Yahoo directories. Meta search engines [7] present results which are from several search engines. Vertical search engines focus on a special field, such as video search, shopping search, and so on. Pay per click search engines are used for advertising products and companies, namely advertise search engines.

In addition to the above popular search engines, more advanced search techniques are investigated. [3] presented a semantic search engine for XML. It has a customized simple query language, and its return results, which are ranked by both the degree of the semantic relationship and the relevance of the keywords, satisfy the users' query requests. [1] depicted an engine used to retrieve source codes, enabling users to search contents and the structural properties and relations from programming languages. [2,8] provided a search engine over images. [4] illustrated a desktop search engine over documents, e-mails, files, and programs on local personal computers. [6,9] indicated importance of enterprise searches and the challenges in these search engines.

3 Application Domain of TaskCardFinder

According to the statistics of Gupta et al., *each year aviation industry spends a whopping amount on MRO, second only to fuel. In 2007, the global MRO industry was estimated at US $ 45 billion, which is expected to reach US $ 61 billion by 2017. As per industry projections, the size of the worldwide air transport fleet will expand by nearly 50 percent in the decade to 2017, and consequently spur rapid growth of the MRO business* [5].

A MRO task card is a directive file in a word document format, containing all the information needed in repairing an airplane for a technican. A big aviation company generates thousands of task cards per day. However, managing the large volume of task cards is a challenge. Our TaskCardFinder focuses on how to organize, manage, and integrate knowledge, information, and data based on its relevant contextual background, with an ultimate goal to quickly retrieve, compose, and deliver bilingual MRO task cards at the right time, to the right users, under the specific task orders, with a higher degree of task-relevancy.

4 Context-Aware Searching Framework

Figure 1 shows the working diagram of our context-aware searching over bilingual MRO task cards. It consists of three parts, namely, *task cards storage, task cards search,* and *user interaction*, where context-awareness penetrates all the three parts.

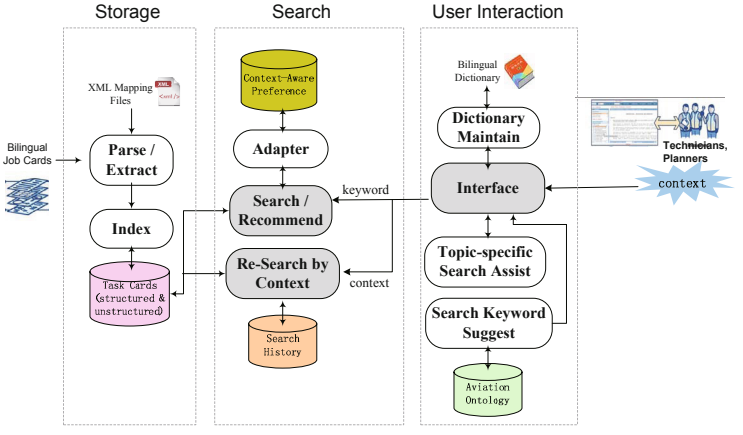


Fig. 1. Framework of context-aware searching over bilingual MRO task cards

4.1 Part I (Storage of Task Cards)

Part I prepares for searching by parsing, indexing and storing task cards into the database. It contains two modules.

- *Parse and Extraction Module* is responsible for parsing and extracting fine-grained information from bilingual task card word documents with the help of a bilingual aviation dictionary. A task card document contains several items like task card number, A/C type, service interval, critical task, general airline mechanic, work content (in both Chinese and English or in either Chinese or English), labor hours, etc. Each item is extracted as a database attribute and sent to the next data management module for further processing (storage, indexing, and search).
- *Index Module* stores the extracted fine-grained job card items, together with the complete job card documents, into a MySQL database. Each job card corresponds to a relation tuple in the database. The module also builds indexes for all different attributes.

4.2 Part II (Search over Task Cards)

Users' search requests can be formulated in either keywords, structured queries, or context. It is comprised of three modules.

- *Search and Recommend Module* performs keyword-based and structured searching upon the MRO database in response to a search request in either English or Chinese. The result returned is a ranked list of job card documents, presented in a textual, structured, or word format.
- *Adapter Module* makes users' implicit information needs explicit by adapting the original search requests with context-aware preference rules.
- *Re-Search by Context Module* supports users' re-finding requests, where search conditions are various context clues such as access histories and access time. Two re-search-by-context examples are as follows. 1) "search relevant MRO job cards which I retrieved last year when I was involved in Project XXX" (where the search condition based on two contextual information - time and user's activity in the past); 2) "search MRO job cards which I retrieved most frequently last year" (aggregate search based on the context - time).

4.3 Part III (User Interaction)

Four modules collaborate to offer a user-friendly interface.

- *Interface Module* enables simple and easy interaction between users and the MRO search engine. It also provides guidance to assist users to easily and simply identify and formulate their search requests. As users' requests may be in either Chinese or English, requests' stemming and translation are incurred at the module as well. The formulation of a user's search request can get helps from the following topic-specific search assist and relevant keyword suggest modules based on the MRO ontology.
- *Topic-Specific Search Assist Module* assists users to navigate among aviation-specific topics and raise more targeted search requests.
- *Relevant Search Keywords Suggest Module* provides other relevant keywords which users may be interested in according to users' inputs.

– *Bilingual Dictionary Maintain Module* is to dynamically and incrementally maintain the bilingual aviation lexicon by consulting the open sources on the Web during users' interaction with the search system.

5 Design and Implementation of TaskCardFinder

The major functionalities of TaskCardFinder include:

1. Keyword and structured MRO task cards search with *bilingual support*, *topic-specific search assistance*, *relevant keywords suggestion support context-aware preferences support*, *Recall-based re-search by time.*
2. Context-aware task cards recommendation.
3. Search results are a ranked list of task cards, allowing online structured view and document preview.

5.1 Context-Aware Bilingual Search in TaskCardFinder

After users' login-in TaskCardFinder, users' context-aware information are adopted to affect either keyword-based or structured search over bilingual task cards, we depicted in Figure 2 together. Topic-specific search assistance can also be provided to allow users to navigate at his/her target search domains (Figure 2), where some statistic information about the corresponding task cards is provided. These statistic information could support MRO engineers to make decisions or plan schedules.

The search result of TaskCardFinder is a list of ranked task cards with bilingual support, presented in Figure 4. For example, when we search "*cabin refitment*", the system can automatically search "客舱改装" together. The results present the documents which contain either Chinese or English contents, or both.

The structural view of of the task card for a result in Figure 4 could help users to quickly judge if this card is what the user is really searching for. Corresponding task card word documents can also be online previewed without downloading the document or downloaded to the disk for later reference.

Fig. 2. Keyword and structured search

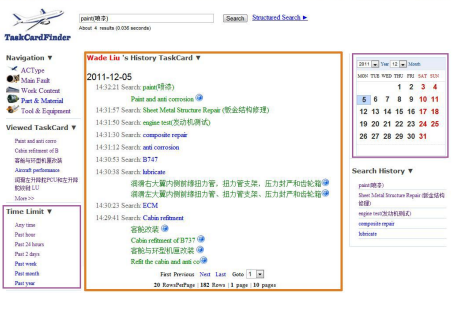


Fig. 3. Recall-based search

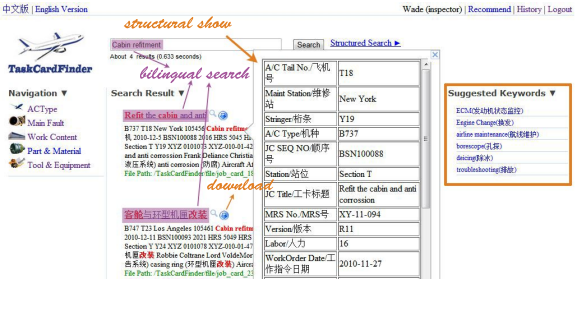


Fig. 4. Search result

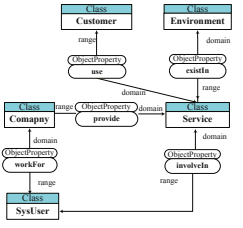


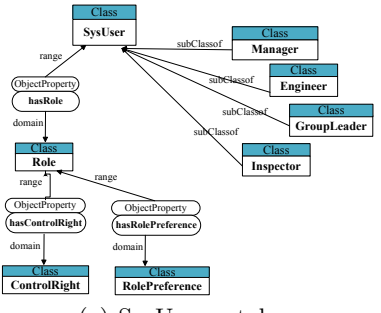
Fig. 5. Core ontology

Keyword-based search histories and task cards viewed by a user are stored for later re-search purpose. This feature could help the user re-locate what s/he searched or scanned in the past according to different search times, as is shown in Figure 3.

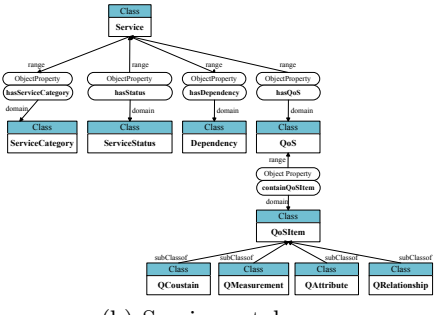
5.2 Context-Aware Preference Search and Recommendation

Adding the user's context information into the search query could enhance the relevancy of the search result remarkably. Establishing the aviation context ontology is to formalize and further standardize static and dynamic contextual elements and their relationships in offering ambient smart and context-ware aviation services to diverse customers. A context-aware aviation service is provided by the company, and used by a customer, taking into account the external environment and bi-parties' internal status. We can view and approach the aviation context from five perspectives: 1) *system user-centric*; 2) *company-centric*; 3) *customer-centric*; 4) *service-centric*; and 5) *environment*.

Figure 5 illustrates 5 basic concepts with their inter-relationships, two of which are introduced. The company provides an aviation service, while a customer uses



(a) SysUser ontology



(b) Service ontology

Fig. 6. Conceptual diagram of Sysuser&Service ontology with DL specification

the service. A service functions in a certain environment. A system user works for the company, plays a certain role, and gets involved in certain services.

$$\begin{aligned} Company &\sqsubseteq \exists provideService.Service \sqcap \forall provideService.Service \\ Customer &\sqsubseteq \exists useService.Service \sqcap \forall useService.Service \\ Service &\sqsubseteq \exists existIn.Environment \sqcap \forall existIn.Environment \\ SysUser &\sqsubseteq \exists workfor.Company \sqcap \forall workfor.Company \end{aligned}$$

Users of the ambient smart and context-aware data management system are from the aviation company. They can be the managers, group leaders, engineers, inspectors, etc, as is shown in Figure 6(a). Different roles of the users have different control rights to the system. Also different roles of users have different role preferences when using the system. Representation of role preference also follows a DL format.

$$\begin{aligned} Company &\sqsubseteq \exists hasRole.Role \sqcap \forall hasRole.Role \\ Role &\sqsubseteq \exists hasControlRight.ControlRight \sqcap \\ &\quad \forall hasControlRight.ControlRight \\ Role &\sqsubseteq \exists hasRolePreference.RolePreference \sqcap \\ &\quad \forall hasRolePreference.RolePreference \\ Manager &\sqsubseteq SysUser \quad GroupLeader \sqsubseteq SysUser \\ Engineer &\sqsubseteq SysUser \quad Inspector \sqsubseteq SysUser \end{aligned}$$

There are different categories of services offered by the company to its customers, like MRO service, modification service, training service, respectively. Dependency relationships exist among different services. Each service has a QoS requirement, detailing quality attributes, constraints, measurements, and relationships. Figure 6(b) presents the ontology diagram of Service.

$$\begin{aligned} Service &\sqsubseteq \exists hasServiceCategory.ServiceCategory \sqcap \\ &\quad \forall hasServiceCategory.ServiceCategory \\ Service &\sqsubseteq hasServiceCategory.ServiceStatus \sqcap \\ &\quad \forall hasServiceCategory.ServiceStatus \\ Service &\sqsubseteq \exists hasServiceDependency.ServiceDependency \sqcap \\ &\quad \forall hasServiceDependency.ServiceDependency \\ Service &\sqsubseteq \exists hasQos.Qos \sqcap \forall hasQos.Qos \\ QConstraint &\sqsubseteq QualityItem \quad QMeasurement \sqsubseteq QualityItem \\ QAttribute &\sqsubseteq QualityItem \quad QRelationship \sqsubseteq QualityItem \end{aligned}$$

We have presented the design of an aviation context ontology with its description logic based representation. Based on the designed aviation context ontology, we can describe system users' context-aware search preferences when they manage aviation services. Further, the search adapter module is designed to support adaptive and personalized search, using this logic-based context-aware search preference module. Here, it is interesting to note that DL concept expressions offer a natural way to represent information needs. For instance, the DL concept expression $Service \sqcap (hasServiceStatus.\{finish\})$ can be viewed as a search request on all finished services provided by the company. Therefore, in a similar fashion as context, we cam describe users' preferences using DL concept expressions as well. Formally, we can define a context-aware search preference as a tuple of the form (Context, Preference), where context and preference are DL concept expressions. For example, a context-aware search preference may like "if the system user assumes an inspector role in some aviation services, preferably returning all finished service records to them" can be specified as:

$$Context : User \sqcap (\exists hasRole.\{Inspector\}) \sqcap (\exists involvedIn.Service)$$
$$Preference : Service \sqcap (\exists hasServiceStatus.\{finish\})$$

Once a user logins in the system, the context information will be sent to the context service which is responsible for selecting the corresponding context-aware searching preferences before submitting the search request to the search module. The search adapter module augments the original request with the preference provided by the context service into the final search request, and submits it to the search service. The search module returns the result list sorted by the relevancy, taking user's preference into consideration. As a result, for different users, the order of the search result under the same keyword is different.

Both context and preferences are represented by description logic and organized by the ontology technique. Through the interface of Figure 7, we can manage and maintain (add, rename, update, and delete) the aviation context ontology and context-aware preference rules, respectively.

5.3 Search Keyword Suggestion

To enable simple and easy interaction between users and the MRO search engine, TaskCardFinder assists users to identify and formulate their exact search requests based on the MRO ontology.

Utilizing the ontology, we associate and suggest some other keywords based on user's input keyword. To associate keywords on the built concept tree (Figure 8), we first locate the keyword first, and then decide the association direction. There are many directions we could take for associating keywords, such as upward to recommend many high level keywords or downward to recommend some detailed keywords and we can also recommend some keywords at the same level of the input keyword, as is shown in Figure 8. For instance, the input is cabin refitment, and the recommend result list is on the right of result list in Figure 4.

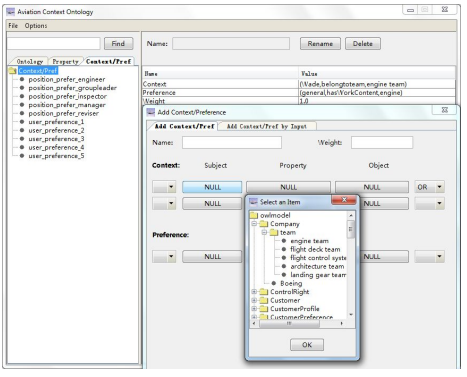


Fig. 7. Management of context-aware preferences

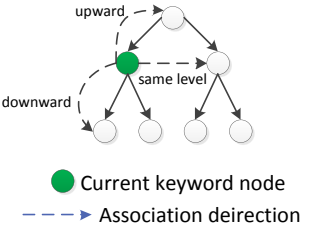


Fig. 8. keyword association

6 Evaluation

Document search is the core of TaskCardFinder. We study its scalability by varying the number of task cards from 20k to 100k. Besides, to reduce the searching time, we consider to decompose a big search task into serval small ones, and assign each of them to one computing thread to execute separately and simultaneously. The results returned are then finally combined and sent back to the user. As TaskCardFinder performs searching based on index files for the MRO task cards, we decompose task cards into a few sets, which are separately indexed. The number of computing threads determines the number of task card sets, and thus the number of index files. Each thread executes searching on its assigned index files. We investigate the effect of thread number on the searching performance.

The computer used is equipped with Intel(R) Core 2 Quad Q9650, 4GB RAM, and windows 7. We conduct an experiment by varying the number of computing threads from 1, 2, to 4. From the result presented in Figure 9, we can see that the searching time increases along with the number of total task cards. The reason is obvious. The performance of two-thread searching is consistently better than

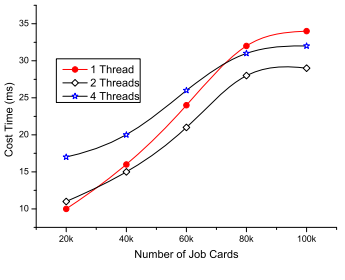


Fig. 9. Searching time versus the number of computing threads

one-thread and four-thread searching. As two-thread searching is based on two sets of index files, allowing some simultaneous handling, it is superior than one-thread searching. However, due to the overheads of results merging and CPU scheduling, more threads may not always lead to faster searching performance. On the experimental computer, four-thread searching is worse than two-thread searching.

7 Conclusion

This article reports the design, implementation, and a preliminary evaluation of an ambient smart and context-aware search engine over a collection of bilingual MRO task cards, called TaskCardFinder. It possesses several novel context-aware features specifically designed for aviation industry. In addition, TaskCardFinder is also a bilingual (Chinese and English) search engine. Two directions related to ambient smart and context-awareness could be explored in the future. Besides recall-based re-search by time, a wider scope of context (like user's activity, task, etc.) could be used as search keywords. We also envision a more intelligent question-answer functionality to be offered by the search engine in the future. To achieve this, construction of a task card warehouse and associated OLAP techniques need to be explored.

Acknowledgement. The work is supported by National Natural Science Foundation of China (60773156 , 61073004), Chinese Major State Basic Research Development 973 Program (2011CB302203-2), Important National Science & Technology Specific Program (2011ZX01042-001-002-2), and Chinese National 863 Advanced Technology Program (2008AA01Z132).

References

1. Bajracharya, S., Ngo, T., Linstead, E., Dou, Y., Rigor, P., Baldi, P., Lopes, C.: Sourcerer: a search engine for open source code supporting structure-based search. In: OOPSLA (2006)
2. Frankel, M.S.C., Athitsos, V.: Webseer: An image search engine for the world wide web. Technical Report TR-96-14 Univ. of Chicago (1996)
3. Cohen, S., Mamou, J., Kanza, Y., Sagiv, Y.: Xsearch: a semantic search engine for xml. In: VLDB, vol. 29, pp. 45–56 (2003)
4. Cole, B.: Search engines tackle the desktop. Computer 38(3), 14–17 (2005)
5. Gupta, P., Dewangan, S., Gade, S.: Time to enable mobility in aviation MRO software solutions (2009)
6. Hawking, D.: Challenges in enterprise search. In: ADC, pp. 15–24 (2004)
7. infospace, `http://www.infospace.com/`
8. Kahn, J., Charles, E., Thao, C.: Goldminer: A radiology image search engine. Am. J. Roentgenol. 188, 1475–1478 (2007)
9. Mukherjee, R., Mao, J.: Enterprise search: Tough stuff. Queue 2, 36–46 (2004)
10. Schmidt, K., Oberle, D., Deissner, K.: Taking enterprise search to the next level
11. Wiki. Internet history - search engines

Formal Analysis of Aviation Incidents

Tibor Bosse[1] and Nataliya M. Mogles[1,2]

[1] Vrije Universiteit Amsterdam, Department of Artificial Intelligence
de Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
{tbosse,nms210}@few.vu.nl
[2] National Aerospace Laboratory NLR
Anthony Fokkerweg 2, 1059 CM Amsterdam, The Netherlands

Abstract. In this paper, a formal, agent-based approach is presented to analyse the dynamics of accidents and incidents in aviation. The approach comprises a number of steps, which include development of domain ontology, formalisation of scenarios, formal specification of dynamic properties, identification of interlevel relations between dynamic properties at different aggregation levels, and automated verification of dynamic properties. The approach is illustrated by means of a case study on a runway incursion incident, and a number of advantages with respect to the current state-of-the-art are discussed.

Keywords: aviation, incidents, agent-based, verification, interlevel relations.

1 Introduction

Within aviation, analysing the exact causes of accidents and incidents is a nontrivial task. Even if detailed flight data from the 'black box' are available, it is usually still difficult to come up with a clear analysis, for the simple reason that the causes of incidents cannot be attributed to a single point of failure of one individual entity. Instead, most incidents in aviation are found to be caused by a complex interplay of processes at various levels of the socio-technical system, involving pilots, air traffic controllers, technical systems, and their interaction. For example, although currently still under investigation, the famous accident in 2009 of Air France Flight 447 seems to have been the consequence of a rare combination of factors, including inconsistent airspeed sensor readings, the disengagement of the autopilot, and the pilot pulling the nose of the plane back despite stall warnings [10].

For the analysis of accidents and incidents in aviation, roughly two streams can be distinguished in the literature, namely *accident analysis* and *risk analysis*. Whilst the former has the goal to determine the cause of an accident that actually took place, the latter aims to assess the likelihood of the occurrence of future accidents. Hence, although both streams have similar purposes, a main difference is that accident analysis attempts to identify one specific combination of hazardous factors, whereas risk analysis basically explores a whole range of such factors, and the associated risks. Nevertheless, most of the existing approaches are used for both streams.

When studying these existing approaches, a number of points for improvement can be identified. Traditionally, analysis of aviation incidents is done via fault and event

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 371–380, 2012.

trees. These are graphical representations of Boolean logic relations between success and failure types of events. However, although widely used, there is an increasing awareness that fault and event trees have serious limitations, especially when it comes to analysing dynamic systems with time-dependent interactions (see [6] for a more extensive argumentation). More recently, alternative approaches have been developed, such as FRAM [7] and STAMP [8]. While these approaches have proved successful in various case studies, they still have some drawbacks. In particular, FRAM lacks a formal semantics, which makes a computational analysis of complex non-linear processes impossible. STAMP does have a formal basis, but takes an aggregated, organisational perspective (based on system dynamics), which hinders an analysis at the level of individual agents (such as pilots and air traffic controllers), and their underlying mental processes.

As an alternative, the current paper presents an approach for analysis of aviation incidents that takes a multi-agent perspective, and is based on formal methods. The approach is inspired by an existing approach for formal analysis of incident management scenarios [1] and the multi-agent dynamic risk modelling (DRM) methodology [2, 9] for the evaluation of air traffic risk, and extends these with a mechanism to establish *interlevel relations* between dynamic properties at different aggregation levels of the system under analysis. The approach is illustrated by a case study on a runway incursion incident at a large European airport in 1995.

The remainder of this paper is structured as follows. In Section 2, a high level overview of the proposed methodology is presented. In Section 3, the scenario used within the case study is described. Section 4 explains in a number of steps how this scenario was formalised, and Section 5 describes how it was formally analysed. Section 6 concludes the paper with a discussion.

2 Methodology

The methodology used in this paper is inspired by the methodology for analysis of cognitive processes put forward in [3]. This methodology assumes that real world scenarios (e.g., aviation incidents) can be described in terms of the evolution of world states over time, and that their dynamics can be studied by formulating certain dynamic statements that are (or are not) expected to hold for the scenarios. A simple example of such a dynamic statement would be 'at all time points, the pilot is aware of the altitude of the aircraft'.

Moreover, to enable automated analysis, the scenarios and dynamic statements need to be expressed in some formal language. To this end, the Temporal Trace Language [4] is used. This predicate logical language supports formal specification and analysis of dynamic properties, covering both qualitative and quantitative aspects. TTL is built on atoms referring to *states* of the world, *time points* and *traces*, i.e. trajectories of states over time. In addition, *dynamic properties* are temporal statements that can be formulated with respect to traces based on the state ontology Ont in the following manner. Given a trace γ over state ontology Ont, the state in γ at time point t is denoted by state(γ, t). These states can be related to state properties via the formally defined satisfaction relation denoted by the infix predicate |=, comparable to the Holds-predicate in the Situation Calculus: state(γ, t) |= p denotes that state property p holds in trace γ at time t. Based on these statements, dynamic

properties can be formulated in a formal manner in a sorted first-order predicate logic, using quantifiers over time and traces and the usual first-order logical connectives such as $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\forall$, $\exists$. A special software environment has been developed for TTL, featuring both a Property Editor for building and editing TTL properties and a Checking Tool that enables formal verification of such properties against a set of (simulated or empirical) traces. More details of TTL can be found in [4].

Based on TTL, the current paper proposes the following 5-step methodology for formal analysis of aviation incidents:

1) *Development of formal ontology*: to develop the state ontology Ont introduced above, all relevant sorts, constants, functions and predicates have to be specified for the domain under investigation, enabling the modeller to describe the relevant aspects of the world (e.g., pilots, controllers, aircraft, actions, communications, mental states, and so on).
2) *Formalisation of real world scenarios in terms of traces*: for each scenario, express the different events using the formal ontology developed in step 1), and allocate a time stamp to them.
3) *Specification of dynamic properties at different levels*: identify dynamic properties that are relevant for the domain, and express them in TTL.
4) *Specification of interlevel relations between dynamic properties*: the dynamic properties identified in step 3) may be classified according to different levels of aggregation of the aviation domain. For instance, some properties may apply to the air traffic organisation as a whole (e.g., 'no incident will occur'), whereas others apply to the level of individual agents (e.g., 'agent A will only communicate correct information'). In this step, logical relationships between dynamic properties at different levels are established, to ensure that conjunctions of properties at one level imply properties at higher levels.
5) *Verification of properties against traces*: using the TTL Checking Tool mentioned above, dynamic properties at different levels are checked automatically against the traces generated in step 2), allowing the analyst to find out what exactly went wrong in the scenarios under investigation.

In subsequent sections, these steps are illustrated by means of the runway incursion case study. In particular, step 1) and 2) are applied in Section 4, and step 3), 4) and 5) are applied in Section 5.

3 Case Study

One of the possible approaches in analysing the behaviour of complex systems in aviation is by identification and formal analysis of case studies. This type of analysis allows researchers to acquire possible underlying information about incidents or almost-incidents within the air traffic domain. Our interest was mainly focused on incidents where a small mistake of one or multiple actors could have led to severe consequences at the level of the whole system, but was corrected by another actor and thus the possible accident was prevented. This focus on incidents was motivated by the fact that the numerous descriptions of air traffic accidents that could be found in the published literature are one-sided, as these cases cover just a small top of the iceberg of all risky situations occurring daily in air traffic interactions. However, it is not so easy to get access to these incidents, as they are mostly company confidential and not available for broad publications, or they are not officially reported at all. To obtain such a case study, it was decided to perform a semi-structured interview with an available expert, a two years retired pilot of a European civil aviation company.

The following subsections provide the overview of an interview that was performed with the available expert and the description of the extracted incident that has been selected for the formal analysis.

3.1 Interview

The interview took place on May 12, 2011 and lasted approximately 1 hour and 15 minutes. It was a semi-structured interview with a predefined set of questions concerning the incidents that the pilot or any of his colleagues had experienced during his flight career. In the beginning of the interview it was clearly announced to the interviewee that we are interested in the cases within air traffic where a small local mistake could have led to severe global consequences, but was corrected before an actual accident would occur. The interviewee was asked to recall such incidents. This question contained the following subquestions:

1. Who was involved in the incident?
2. What was the cause of the problem?[1]
3. How was the problem solved?
4. What were the consequences?
5. Was the situation familiar to you from trainings or procedures?

During the interview a case study was identified that describes an incident where, due to the mistake of a pilot of one taxiing aircraft, two aircraft were taking off almost simultaneously from crossing runways. After the correct interference of the air traffic controllers from the ATC Tower, and adequate decision making, coordination and action of the pilots of one of the aircraft, a collision was prevented. This incident is described (in an anonymised manner) in the following section.

3.2 Runway Incursion Incident Description

The incident took place in 1995, during the departure of an Airbus A310 of a civil aviation company from one large airport in Europe, and is summarised below:

The Airbus was preparing for the departure: the pilot-in-command was sitting on the left and the co-pilot on the right seat in the cockpit and they were ready to start taxiing. They were supposed to taxi to runway 03 in the north-east direction. The Airbus received permission to taxi and started taxiing to its runway. Approximately at the same time, a military Hercules aircraft that was ready for the departure as well received permission to taxi in the north-west direction from its parking gate. The Hercules was supposed to take off from runway 36 that crossed with runway 03 that was designated for the Airbus. Both aircraft were taxiing to their runways. During the taxiing, the Airbus received its flight route from the air traffic controllers. Some time tater, when the Airbus was near the runway designated for taking off, it switched from the taxiing radio frequency to the frequency of the Tower and received permission to line up on the assigned runway. The Hercules was still at the taxiing radio frequency and also

[1] Possible causes that might be relevant include failure of technical systems, miscommunication, fatigue, high or low workload (restricted Situation Awareness or decreased vigilance), strong positive or negative emotions, power influences, (dis)trust in colleagues or computer systems, little experience, negligence of the existing procedures, organisational management etc.

received permission to line up, while at the same time the Airbus received permission to take off at the radio frequency of the Tower. However, due to unknown reasons[2]*, the Hercules pilot interpreted his permission for lining up as permission for taking off and started taking off on runway 36. As a result of this mistake of the pilot of the Hercules, two aircraft were taking off simultaneously on crossing runways, and none of the crews were aware of that. The air traffic controllers in the Tower observed the conflicting situation and communicated a 'STOP' signal to the pilot-in-command of the Airbus, while the Airbus was still on the ground (but at high speed). The pilot had to make a quick decision about the termination of the take-off as there is a point in this process that one cannot safely do this anymore. After having analysed the situation, the pilot-in-command of the Airbus gave a command to the co-pilot (who controlled the aircraft) to abort the take-off and start braking on the runway. During braking, the crew of the Airbus saw the Hercules flying close in the air above their own aircraft at a distance of about 5 meters. The serious collision was prevented.*

4 Formalisation

This section provides a description of the formal ontology of the scenario and the formal trace based on this case study.

4.1 Formal Ontology

As the first step towards the formalisation of the incident identified during the interview, a formal domain ontology was developed in TTL. In Table 1 an overview of the ontology elements is shown, including the relevant sorts and subsorts relations, elements (constants) of sorts, and logical predicates over sorts.

Table 1. Domain Ontology

PREDICATES	DESCRIPTION
	Communication
communicate_from_to(A:AGENT, B:AGENT, I:INFO_EL)	agent A communicates information I to agent B
	Internal states of agents
observation(A:AGENT, I:INFO_EL)	agent A observes information element I from the world
belief(A:AGENT, I:INFO_EL)	agent A beliefs that information element I from the world is true
world_state(I:INFO_EL)	information element I holds in the world
	Actions of agents
move_from_to(R1:ROADWAY, R2:ROADWAY)	action of moving from roadway R1 to roadway R2
start_take_off(R:RUNWAY)	action of starting taking off on runway R
stop_take_off	action of ceasing taking off
	Information elements used within predicates
is_at_position (A:AGENT, P:POSITION)	agent A is at a certain position P
performed(A:AGENT, C:ACTION)	agent A performs action C
start_taxiing(T:TAXIWAY)	start taxiing on taxiway T
permission_line_up(R:RUNWAY)	permission to line up on runway R
permission_take_off(R:ROADWAY)	permission to take off on roadway R
crossing_ways(R1:ROADWAY,R2:ROADWAY)	roadways R1 and R2 are crossing
request(A:ACTION)	request to perform action A
conflict_situation(A1:AGENT, A2:AGENT)	conflict situation between agent A1 and agent A2

[2] This misinterpretation might be explained by the fact that the pilot of the Hercules got used to the routine procedure of taxiing from the same military parking place at this airport and perhaps also of taking off from the same runway. And in many past cases, the line up procedure was often immediately followed by taking off, as permissions for lining up and taking off were sometimes given simultaneously.

Table 1. (*continued*)

SORT	ELEMENTS
AGENT	sub-sorts: PILOT, OTHER
PILOT	{airbus_pilot, airbus_co-pilot, hercules_pilot}
OTHER	{tower, hercules, airbus}
ROADWAY	sub-sorts: RUNWAY, TAXIWAY
RUNWAY	{runway_03, runway_36}
TAXIWAY	{taxiway_1, taxiway_2}
POSITION	{startingpoint_1, startingpoint_2, before_critical_point, after_critical_point, close_in_air}

4.2 Formal Trace

The informal scenario described in Section 3 was formalised using the ontology presented in the previous subsection. A time point was assigned to each event of the case study under consideration.

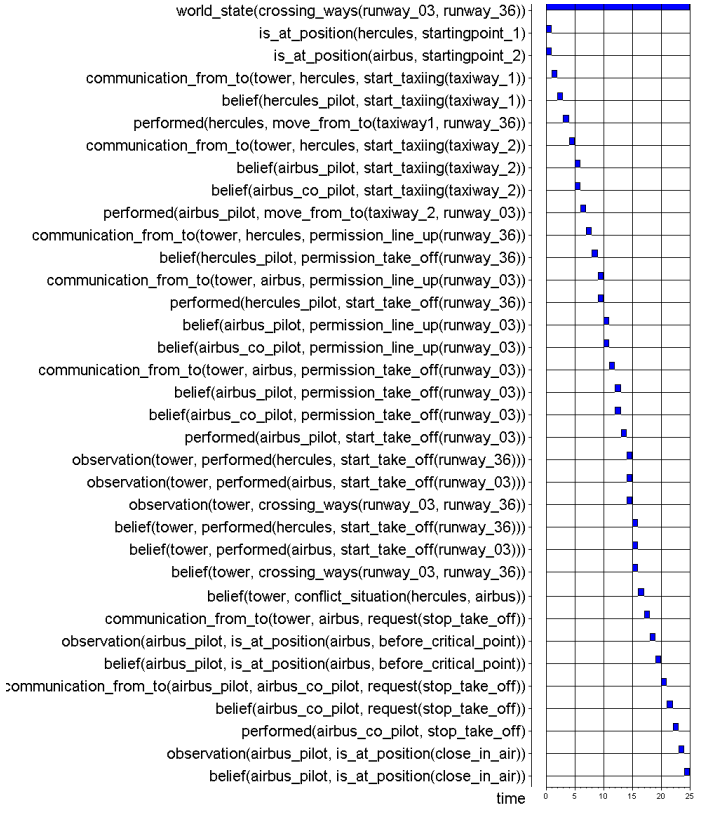


Fig. 1. Formalised Empirical trace of the Runway Incursion Incident in LEADSTO

The time points in the trace indicate the relative timing of the events. The trace was formalised in the LEADSTO software environment [5]. An example of a visualisation

of the trace in this environment is shown in Figure 1. The states that hold in the world are represented on the vertical axis and the time line on the horizontal axis. The dark lines on the right indicate time intervals within which the given states are true.

5 Formal Analysis

This section addresses automated analysis of the scenario by specification and verification of dynamic properties at different levels of aggregation. Section 5.1 addresses specification of dynamic properties, and Section 5.2 address specification of interlevel relations and verification of properties.

5.1 Dynamic Properties

Various dynamic properties for the aviation domain have been formalised in TTL. Below, a number of them are introduced, both in semi-formal and in informal notation. Note that these properties address processes at different aggregation levels, thereby distinguishing global properties about the scenario as a whole (indicated by GP), intermediate properties about input and output states of individual agents (indicated by IP), and local properties about mental processes of individual agents (indicated by LP):

GP - No simultaneous take-off at crossing runways
There are no trace m, time points t1 and t2, agents a1 and a2, and runway r1 and r2 such that
agent a1 performs a take-off on runway r1 at time t1
and agent a2 performs a take-off on runway r2 at time t2
and runway r1 and r2 are crossing runways
and the difference between t1 and t2 is smaller than or equal to some constant d.

```
¬ [∃m:TRACE ∃t1,t2:TIME ∃a1,a2:AGENT ∃r1,r2:RUNWAY
    state(m, t1) |= performed(a1, start_take_off(r1)) &
    state(m, t2) |= performed(a2, start_take_off(r2)) &
    state(m, t1) |= world_state(crossing_ways(r1, r2)) &
    | t1 - t2 | ≤ d ]
```

IP1 - No simultaneous permissions to take off at crossing runways
There are no trace m, time points t1 and t2, agents a1 and a2, and runway r1 and r2 such that
the tower gives agent a1 permission for take-off on runway r1 at time t1
the tower gives agent a2 permission for take-off on runway r2 at time t2
and runway r1 and r2 are crossing runways
and the difference between t1 and t2 is smaller than or equal to some constant d.

```
¬ [∃m:TRACE ∃t1,t2:TIME ∃a1,a2:AGENT ∃r1,r2:RUNWAY
    state(m, t1) |= communicate_from_to(tower, a1, permission_take_off(r1)) &
    state(m, t2) |= communicate_from_to(tower, a2, permission_take_off(r2)) &
    state(m, t1) |= world_state(crossing_ways(r1, r2)) &
    | t1 - t2 | ≤ d ]
```

IP2 - Each take-off is preceded by a corresponding permission
For all traces m, time points t1, agents a, and runways r
if agent a performs a take-off on runway r at time t
then there was a time point t2 with t1-d ≤ t2 ≤ t1 (where d is a constant) on which
the tower gave agent a permission for take-off on runway r.

∀m:TRACE ∀t:TIME ∀a:AGENT ∀r:RUNWAY
state(m, t1) |= performed(a, start_take_off(r)) ⇒
[∃t2:TIME state(m, t2) |= communicate_from_to(tower, a, permission_take_off(r)) &
t1-d ≤ t2 ≤ t1]

LP1 - Each take-off is preceded by a corresponding belief
For all traces m, time points t1, agents a, and runways r
if agent a performs a take-off on runway r at time t
then there was a time point t2 with t1-d ≤ t2 ≤ t1 (where d is a constant) on which
agent a believed that it had permission for take-off on runway r.

∀m:TRACE ∀t:TIME ∀a:AGENT ∀r:RUNWAY
state(m, t1) |= performed(a, start_take_off(r)) ⇒
[∃t2:TIME state(m, t2) |= belief(a, permission_take_off(r)) &
t1-d ≤ t2 ≤ t1]

LP2 - Each belief is preceded by a corresponding communication
For all traces m, time points t1, agents a, and runways r
if agent a believes that it has permission for take-off on runway r at time t
then there was a time point t2 with t1-d ≤ t2 ≤ t1 (where d is a constant) on which
the tower gave agent a permission for take-off on runway r.

∀m:TRACE ∀t:TIME ∀a:AGENT ∀r:RUNWAY
state(m, t1) |= belief(a, permission_take_off(r)) ⇒
[∃t2:TIME state(m, t2) |= communicate_from_to(tower, a, permission_take_off(r)) &
t1-d ≤ t2 ≤ t1]

5.2 Interlevel Relations

A number of logical relationships have been identified between properties at different aggregation levels. An overview of all identified logical relationships relevant for GP is depicted as an AND-tree in Figure 2.

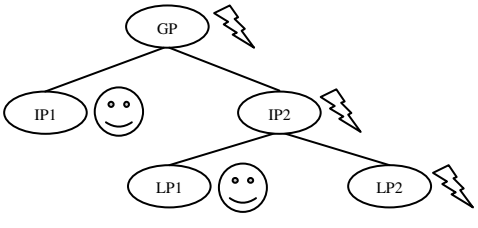


Fig. 2. AND-tree of Interlevel Relations between Dynamic Properties

The relationships depicted in this figure (while for the moment ignoring the symbols next to the ovals) should be interpreted as semantic entailment relationships. For example, the relationship at the highest level expresses that the implication IP1 & IP2 => GP holds, whereas the relationship at the lower level expresses that LP1 & LP2 => IP2 holds. A sketch of the proof for the first implication is as follows (for simplicity reasons abstracting from time constrains):

Suppose that IP1 and IP2 hold. Then, according to IP1, no two permissions to take off at crossing runways will be communicated simultaneously. Moreover, since take-offs are only performed immediately after a corresponding permission has been communicated (as guaranteed by IP2), no simultaneous take-offs are performed at crossing runways. This confirms GP.

Such logical relationships between dynamic properties can be very useful in the analysis of incident scenarios, especially when used in combination with the TTL Checking Tool mentioned earlier. For example, for the runway incursion case study, checking GP pointed out that this property was not satisfied (indicated by the lightning symbol in Figure 2). As a result, by a refutation process it could be concluded that either IP1 or IP2 failed (or a combination of them). When, after further checking, IP2 was found to be the cause of the failure, the analysis could proceed by focusing on LP1 and LP2. Eventually, LP1 was found satisfied, whereas LP2 failed. Thus, (part of) the source of the incident could be reduced to failure of LP2, i.e., there was an agent (namely the pilot of the Hercules) that believed to have the permission to take off, whilst this was not communicated by the tower. A discussion with our domain expert confirmed that this was indeed the case.

Note that the example provided here is mainly meant as an illustration of the approach. In addition to this relatively simple case, similar trees of interlevel relations are being constructed that involve more properties at multiple levels. For such more complex cases, the diagnostic process is economic in the sense that, when a certain property holds, the entire subtree under this property does not have to be examined.

6 Discussion

In this paper, a formal approach was presented to analyse the dynamics of accidents and incidents in aviation. The approach comprises a number of steps, including development of a domain ontology, formalisation of scenarios, formal specification of dynamic properties, identification of interlevel relations between dynamic properties, and automated verification of dynamic properties. The approach was illustrated by means of a case study on a runway incursion incident. This case was considered appropriate for our purposes because it involved an interesting combination of factors (involving pilots, controllers, and technical systems), and a domain expert was available to provide detailed information about the incident.

Besides the fact that the approach is formal (thus enabling a computer-supported analysis, with the usual advantages such as computational power and reduction of errors), it has as advantage over approaches like [8] that it addresses not only at an aggregated, organisational level, but also focuses on more local, individual processes, possibly even up to the level of cognitive processes involving beliefs, intentions, and

so on. Specification of the dynamic properties is still a time consuming task, but once specified, they have a generic nature, allowing for them to be reused for various case studies. Moreover, the possibility to express interlevel relations allows the analyst to gain more insight in how certain global phenomena relate to local mechanisms of individual components of the system, and (via computational checks) to reduce the causes of global incidents to more local failures.

Note that this diagnostic process has an added value not only for analysing existing scenarios (as in accident analysis), but also for analysing potential future scenarios (as in risk analysis). For this latter direction, an interesting step would be to apply the idea of interlevel relationships and checking of properties to the results of (Monte Carlo) simulations (e.g., as in [9]). This idea will be further explored in follow-up research.

Acknowledgements. This work was performed under the auspices of the SESAR WP-E research network ComplexWorld. It is co-financed by Eurocontrol on behalf of the SESAR Joint Undertaking. The authors are grateful to the retired airline pilot who participated in the interview for his useful input on the case study, and to Jan Treur for a number of fruitful discussions.

References

1. Abbink, H., Dijk, R., van Dobos, T., Hoogendoorn, M., Jonker, C.M., Konur, S., Maanen, P.P., van, Popova, V., Sharpanskykh, A., Tooren, P., van, Treur, J., Valk, J., Xu, L., Yolum, P.: Automated Support for Adaptive Incident Management. In: van de Walle, B., Carle, B. (eds.) Proceedings of the First International Workshop on Information Systems for Crisis Response and Management, ISCRAM 2004, pp. 69–74 (2004)
2. Blom, H.A.P., Bakker, G.J., Blanker, P.J.G., Daams, J., Everdij, M.H.C., Klompstra, M.B.: Accident risk assessment for advanced air traffic management. In: Donohue, G.L., Zellweger, A.G. (eds.) Air Transport Systems Engineering, pp. 463–480. AIAA (2001)
3. Bosse, T.: Analysis of the Dynamics of Cognitive Processes. Ph.D. Thesis, Vrije Universiteit Amsterdam (2005)
4. Bosse, T., Jonker, C.M., van der Meij, L., Sharpanskykh, A., Treur, J.: Specification and Verification of Dynamics in Agent Models. International Journal of Cooperative Information Systems 18, 167–193 (2009)
5. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by SimulaTiOn. International Journal of AI Tools 16(3), 435–464 (2007)
6. Everdij, M.H.C.: Review of techniques to support the EATMP Safety Assessment Methodology. Report for EEC Safety Methods Survey project, vol. I and II (2004), http://www.eurocontrol.int/eec/public/standard_page/DOC_Report_2004_001.html
7. Hollnagel, E.: Barriers and accident prevention. Ashgate, Aldershot (2004)
8. Leveson, N.: A new accident model for engineering safer systems. Safety Science 42, 237–270 (2004)
9. Stroeve, S.H., Blom, H.A.P., Bakker, G.J.: Systemic accident risk assessment in air traffic by Monte Carlo simulation. Safety Science 47, 238–449 (2009)
10. http://en.wikipedia.org/wiki/Air_France_Flight_447

Common Sensorimotor Representation for Self-initiated Imitation Learning

Yasser Mohammad[1], Yoshimasa Ohmoto[2], and Toyoaki Nishida[2]

[1] Assiut University, Egypt
yasserm@aun.edu.eg
[2] Kyoto University, Japan
nishida@i.kyoto-u.ac.jp

Abstract. Internal representation is an important design decision in any imitation learning system. Actions and perceptual spaces were separate in classical AI due to the standard sense-process-act loop. Recently another representation that combines the two spaces into what we call a common sensorimotor space was inspired by the discovery of mirror neurons in animals and humans. The justification of this move is usually biological plausibility. This paper reports on a series of experiments comparing these two alternatives for self-initiated imitation tasks. The results of these experiments show that using a common sensorimotor representation allows the system to achieve higher accuracy and sensitivity. This is shown to be true (for our scenarios) even when the dimensionality of the common sensorimotor representation is higher than the dimensionality of the separate perceptual space. It also allows for an easier behavior generation mechanism and ensures reproducibility of learned behavior by the learner.

1 Introduction

Representation issues loom large in AI research specially since the work of Brooks [5]. In this paper, we do not engage in the discussion about advantages and disadvantages of representation in general but we discuss the advantages and disadvantages of two specific representations for a very specific task. This way, we believe the discussion can be made quantitative and more applicable to real world problems.

Our task is learning from demonstration (or imitation learning) where the imitator (learner) watches another agent (demonstrator/teacher/imitatee) performing some behavior and the goal of the imitator is to replicate that behavior. We are interested in the self-initiated case in which the demonstrator does not signal the boundaries of the behaviors to be learned in the perceptual stream of the imitator [11]. This paper focuses on comparing two different ways in which the behavior to be learned is represented/encoded by the imitator. In the first approach, the imitator has two separate perceptual and action spaces and the behavior is modeled in the perceptual space (we call this the two-spaces approach). In the second approach, a common sensorimotor space is defined for

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 381–390, 2012.

the imitator and the behavior is encoded directly in this space (we call this the single-space approach).

An important advantage of this task for representation comparison is that the accuracy of the system can be measured by the accuracy of boundary detection without the need to model the learned behavior. In this paper, we show that, in a series of experiments in different tasks, the single-space approach provided superior performance to the two-spaces approach in self-initiated imitation. We then discuss these results and their implications for future work in imitation learning.

The first forms of imitation learning research in robotics can be traced back to 1980s [6]. Demonstrations were done by hand on the imitator itself and the behavior was stored in the form of a graph connecting states to actions. Deliberative reasoning was then used to combine these graphs into behaviors. Here the learning is done in the action space directly and because the demonstrator and the imitator are the same robot, there is no need for the representational distinction we outlined in the previous paragraph.

More recent research usually assumes that the demonstrator and the imitator are different agents (with possible different form factors). Some researchers simplify the problem by assuming that the pose/state of the demonstrator is directly accessible by the imitator (e.g. using motion-capture devices) [3]. A more interesting situation arises when the imitator has access only to the behavior generated by the demonstrator [4] [1]. In such cases the distinction between single-space and two-spaces representations becomes critical.

In [1], the authors present a Central Pattern Generator model that is based on using a nonlinear window to modify an oscillator to reproduce smooth periodic behavior perceived by the imitator (as demonstrated by the imitatee). The imitator records the path followed by a robotic marionette controlled by a human demonstrator. A hand-eye coordination process is used to learn mapping between the perceived path of the marionette robot and the action space of the controller. CPG is used to learn a model of the motion performed by the demonstrator in the perception-space (utilizing both forward and backward mapping functions). This is a case of the two-spaces approach. One limitation of this approach is that it learns only periodic behavior. Another more important problem in our current research context is that the boundaries of the motion to be learned from the demonstrator is explicitly set by the experimental setup.

In [11], the authors present a system for solving the self-initiated imitation problem that utilizes the single-space approach and is based on a combination of change point discovery and constrained motif discovery algorithms. This is the basis of our analysis in this paper.

The rest of this paper is organized as follows: In section 2, the self-initiated imitation engine that will be used in all experiments is introduced. Section 3 provides a series of experiments to compare the two representations in different tasks. Section 4 provides a final discussion of the results reported in section 3 and directions of future research. The paper is then concluded.

2 Self-initiated Imitation

Self-initiation imitation is the problem of segmenting interesting behaviors for imitation from the continuous actions of the demonstrator utilizing no information from that demonstrator. Self-initiated imitation learning will be our task in this paper for comparing the common sensorimotor representation to the traditional separate action and perception spaces approach.

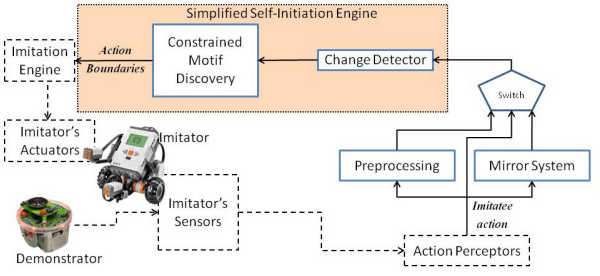


Fig. 1. A simplified version of the self-initiation engine showing the two possible representation

Fig. 1 shows a simplified version of the self-initiation engine proposed in [11] after removing high-level cognition components and input streams that does not affect the analysis in this paper. The input to the system is calculated by projecting the behavior of the demonstrator into one of two possible spaces using either the mirror-system or the preprocessing components.

Let's consider a demonstrator going around doing different actions. Even in absence of any knowledge about the domain or any intrinsic goals for the learner (i.e. no high-level cognition), it is still possible to detect interesting (salient) patterns of behavior based on the assumption that *the saliency of any behavior increases with every change in its measured features*. For example, it is well-known that care-givers use very specific types of motion when teaching their children that they do not use when interacting with adults. These special motion patterns (called motionese [13]) were shown to increase the saliency of objects and behavior by moving the first and exaggerating the later [13]. In this case the only saliency feature would be a change in one of the measured signals in the *imitatee-action* stream and self-initiated imitation will reduce to discovering the patterns recurring around these changes. In [8], a solution to this problem was developed using CMD by utilizing a change point discovery algorithm [9]. The advantage of this two-steps solution over traditional motif discovery algorithms for imitation learning is that it provides a natural way to incorporate more constraints from high-level cognition as well as changes in the state of the environment as discussed in [11].

The focus of this paper is on the utility of using a mirror system to drive the engine which results in a single-space/common sensorimotor representation.

The mirror system can be modeled mathematically by a map from the perceptual space of the imitator to a common sensorimotor representation that may or may not correspond to the action space. This same mirror system will be activated when the imitator later executes the action. Because multiple robot configurations may result in the same state in the perceptual space, this map is not unique. Moreover, the general problem of finding this map is an inverse-kinematics problem that is highly nonlinear in most cases and difficult to solve in closed form. In this paper, we use a simple technique that can overcome these two problems in the same time. The perceived behavior of the demonstrator is a multidimensional time series that we call $\theta(t)$. The first step is to build a forward kinematic model of the imitator in the form of a Jacobian matrix J. This is a simple problem that can be solved in closed form for many manipulators [14]. If not, it can easily be learned in a motion babbling stage as proposed in [1]. Once we have this Jacobian, the mirror system can be expressed by its pseudo-inverse:

$$M = \left(JJ^T\right)^{-1} J^T$$

Because the distance (in the perceptual space) of consecutive points on the path performed by the demonstrator is small, this simple approach generates a smooth path in the common sensorimotor space. Notice that we used the action space of the imitator as the common sensorimotor space but this needs not be the case. Any space that can be easily mapped to both the action and perceptual spaces of the imitator can be used. Future research will analyze using different candidates for this space.

The first component of the engine is the change detector. Change detection is the basic saliency estimation technique that constructs the constraints needed to drive the constraint motif discovery algorithm. There are many change point discovery algorithms that can be used with our system [2] [12] [8]. In this paper we use a an algorithm presented first in [10].

Assume that we have a single dimension time series $X = \{x(t) : 1 \leq t \leq T\}$. The essence of all CPD algorithms (including the one presented here) is to find for every point $x(i)$ the difference between a representation of the dynamics of the few points before it (i.e. $x(i-p) : x(i)$) and the few points after it (i.e. $x(i+g) : x(i+f)$) where p, g and f are all integers and; p and f are greater than zero. This difference is the estimate for $c(t)$. The proposed algorithm has two parameters (w and n) corresponding to the height and width of the Hankel matrices used to represent the dynamics before and after any time step (t). A subspace is generated to represent both the past and future subsequences of the time series at every point and the change score is calculated as:

$$C(t) = \frac{1 - \cos\theta_A(t)}{1 - \cos\theta_B(t)} \tag{1}$$

where $\theta_A(t)$ is the angle between the subspace representing the past and a similar representation of the whole time-series, while $\theta_B(t)$ is the angle between the subspace representing the future and the same representation of the whole

time series. Due to lack of space, details of this algorithm will not be given in this paper. For more details please refer to [10].

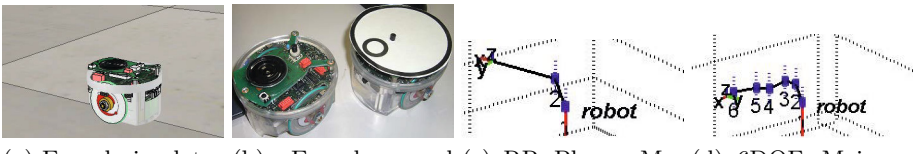
The second component of the engine is the constrained motif discovery algorithm. CMD algorithms find recurrent patterns in single-dimension time-series subject to user defined constraints. The main advantage of CMD algorithms compared to other motif discovery algorithms is the ability to utilize domain knowledge in the form of constraints that inform the mining process. We use the *MCFull* algorithm introduced in [8] for solving CMD. Again, any other algorithm for solving CMD problems can be utilized. The only modification we applied to *MCFull* was to extend it to multiple dimensions by first projecting the input into a single dimension using PCA. Please refer to [8] for details of this algorithm.

3 Evaluation

The focus of this paper is on the utility of using the mirror system described in section 2 to drive the engine which results in a single-space/common sensorimotor representation (this is called the *Mirror* condition hereafter).

Two other control systems were used to evaluate the utility of the mirror system (both corresponding to two-spaces representations.

In the first condition the demonstrator's behavior as perceived by the sensors of the imitator are just passed directly to the engine (*Direct* condition hereafter).

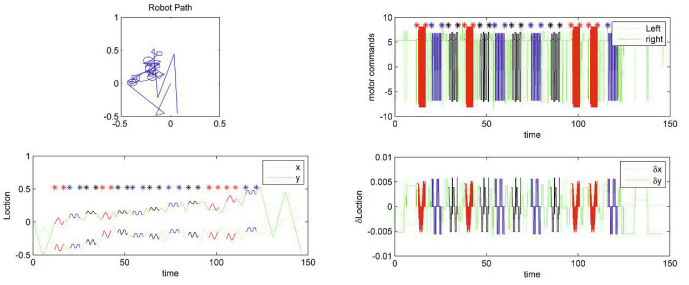


(a) E-puck simulator (b) E-puck real robot (c) RR Planar Manipulator Simulator (d) 6DOF Mainpulator Simulator

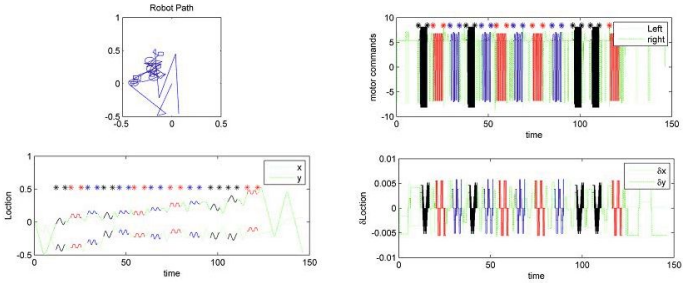
Fig. 2. The Robots used in the Evaluation Experiments

In the second condition, the signal is pre-processed by finding its first difference before passing it to the engine (*Preprocessed* condition hereafter). The justification of the *Preprocessed* condition is that this first difference effectively maps the information in the perceived demonstrator's action to its own egocentric frame of reference.

We evaluate the system using four statistics considering the complete occurrences in the ground truth and the ones found by the system. To measure these statistics we construct a time series of the same length as the demonstrator-action input that represent the ground truth occurrence location for all the patterns that the learner is expected to learn. This time series will be one in all points covered by occurrences of any pattern and zero otherwise. We define true-positives (TP) as the locations that are one in this time series and covered by learned behaviors. true-negatives (TN), false-positives (FP) and false-negatives



(a) Ground Truth



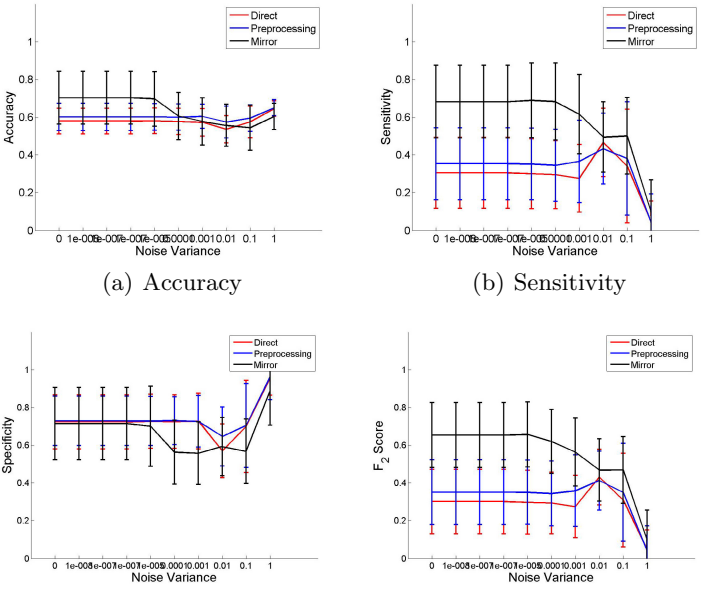
(b) Learned Patterns

Fig. 3. Ground Truth and learned patterns (Mirror Condition) in the original data in the first experiment (Differential Drive). Different Motifs are shown in different colors.

(FN) are defined using a similar method. For evaluation purposes we use four measurements based on these values:pattern accuracy ($\frac{TP+TN}{TP+TN+FP+FN}$), pattern sensitivity ($\frac{TP}{TP+FN}$), pattern specificity ($\frac{TN}{TN+FP}$) and pattern F_2 score ($F_\beta = \frac{(1+\beta^2)\times TP}{(1+\beta^2)\times TP+\beta^2\times TN+FP}$ for any value of *beta*).

The system was also evaluated using similar statistics applied to each discovered pattern alone. Due to lack of space these results will not be presented here but they were very similar to the ones reported.

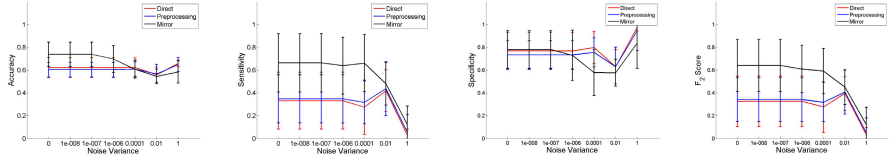
In the first experiment the imitator and demonstrator are both differential-drive e-puck robot simulator (Fig. 3). The demonstrator moves around an empty arena randomly but with some small probability it does one of three predefined motion patterns (Circle, Square and Triangle). The session is continued until we have eight occurrences of each pattern. The imitator receives the location of the demonstrator at every time step and this is the imitatee action stream in Fig. 1. We run the imitation engine with the three conditions defined earlier in this section by setting the switch to its three possible positions in Fig. 1. This experiment was conducted 1000 times for each condition (100 different paths with 10 different added synthetic noise for each case). Fig. 3-a shows the path of the robot as well as the inputs to the imitation engine in the three conditions during one of the sessions with the ground truth locations of patterns marked. Fig. 3-b shows the learned patterns over the signal when the mirror system is enabled.



(a) Accuracy (b) Sensitivity

(c) Specificity (d) F_2

Fig. 4. Differential Drive → Differential Drive

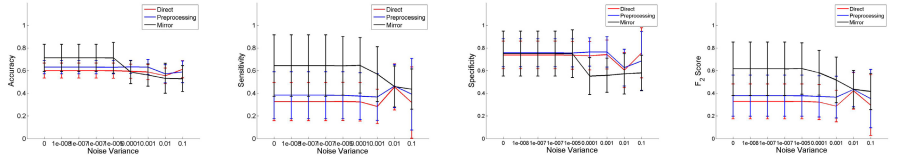


(a) Accuracy (b) Sensitivity (c) Specificity (d) F_2

Fig. 5. RR Serial Manipulator → RR Serial Manipulator

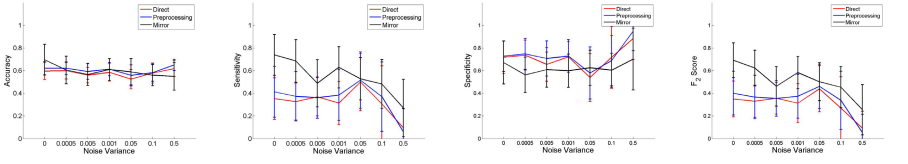
As clearly visible, the system is able to accurately localize the boundaries. Fig. 4 shows the pattern accuracy, specificity, sensitivity, and F_2 score as a function of the standard deviation of the noise. It is clear that in all noise levels, the common sensorimotor (single-space) representation achieved using the mirror system was superior to the other two conditions in terms of accuracy, sensitivity and F_2 score (The difference is statistically significance according to paired t-tests with single tail and a p-value less than 0.001). The three conditions had the same specificity. We can also see that in most (but not all cases), the proposed preprocessing improved the performance of the two-spaces representation.

The same experiment was carried out using a real e-puck robot for 10 sessions with a vision based localization system that utilized a marker on the robot (see Fig. 3) and the results were similar to the ones in Fig. 4. Due to lack of space, these results will not be reported here.



(a) Accuracy (b) Sensitivity (c) Specificity (d) F_2

Fig. 6. RR Serial Manipulator → Differential Drive



(a) Accuracy (b) Sensitivity (c) Specificity (d) F_2

Fig. 7. RR robot → 6DoF Serial Manipulator

The third experiment used a RR 2D planner simulated robot (Fig. 3) that was randomly moving and with a small probability drew three predefined random shapes until eight occurrences of each patterns are conducted. Again, a single session of data was fed to our system and the perceived behavior of the robot was the location of the end effector in the plane. This experiment was conducted 1000 times for each condition (100 different paths with 10 different synthetic noise levels).

Fig. 5 shows the pattern accuracy, specificity, sensitivity, and F_2 score as a function of the standard deviation of the noise. It is, again, clear that in all noise levels, the common sensorimotor (single-space) representation achieved using the mirror system was superior to the other two conditions in terms of accuracy, sensitivity and F_2 score with not statistically significant difference in specificity.

The fourth experiment tried to evaluate the system when the form factor of the two robots is very different. In this case, we used the differential drive robot as the imitator and the 2DoFs RR planner robot as the demonstrator. Fig. 6 shows the results of this experiments which is the same as the previous three except that in this case there is no difference in accuracy.

In all these cases, the dimensionality of the space used for learning was the same in the three conditions. To evaluate the system under a harsher condition, we used a redundant 6DoF serial manipulator as the imitator (Fig. 3) while keeping the RR robot as the demonstrator. In this case, the dimensionality of the common sensorimotor approach (using the mirror system) is higher than the dimensionality of the perceptual space used in the other two conditions. Fig. 7 shows the results of this experiments. Now the difference between the three conditions in accuracy disappeared. The single-space representation is still superior in terms of sensitivity and F_2 score while the two-spaces conditions are now superior in terms of specificity at some noise levels.

4 Discussion

Neuroscience investigations have demonstrated physiological mechanisms of mirroring at single-cell and neural-system levels that support the cognitive and social psychology constructs [7]. Moreover, the ideomotor framework of human actions assumes a common representational format for action and perception that facilitates imitation [7]. This gives theoretical support for the common sensorimmotor representation.

Practically, what is most important in self-initiated imitation is having high accuracy in marking the boundaries of behaviors to be learned. Our analysis presented in section 3 for different tasks with one and two form factors (as well as other tasks not reported for lack of space) reveals that in all cases, the single-space approach had two major advantages: Firstly, it requires no preprocessing of the signal to remove the dependence on the frame of reference and initial state during different demonstrations. The two-spaces approach with its dependence on the perceptual space, will require such pre-processing. Secondly, the single-space approach provides higher sensitivity and accuracy compared with the two-spaces approach (even with pre-processing).

This superior performance of the common sensorimotor representation may be attributed to the fact that this representation is by definition ego-centric and though it does not depend on the choice of the reference frame. Even if the two-spaces representation is preprocessed to provide an ego-centric perceptual input, the mapping between this space and the action space will generally be one to many and the choice of a smooth trajectory in the configuration space will still be a problem. In general, the common sensorimotor representation corresponds to modifying the learning problem as if the demonstrator was the imitator but with its joints moved by a teacher which is the simple problem targeted in early imitation learning literature [6].

These results though are to be taken with caution because they only apply to the boundary detection problem in self-initiated imitation learning and even in this domain, more tests with different tasks will be required.

5 Conclusion

This paper presented a series of experiments to evaluate the accuracy, sensitivity and specificity of unsupervised behavior boundary discovery in self-initiated imitation using two approaches for behavior encoding: single-space representation employing a common sensorimotor space and two-spaces approach employing separate perceptual and action spaces. The results reported in the paper, show that for self-initiated imitation learning in different tasks, the single-space approach provides higher accuracy and sensitivity and in the same time requires no pre-processing of the input streams to remove the effect of the frame-of-reference choice and initial state of different demonstrations. These results provide a computational reason to use a common sensorimotor representation for imitation tasks which augments the biological plausibility of this approach in light of recent results in neuroscience including mirror neurons.

References

1. Ajallooeian, M., Ahmadabadi, M.N., Araabi, B.N., Moradi, H.: An Imitation Model based on Central Pattern Generator with application in Robotic Marionette Behavior Learning. In: IEEE IROS, pp. 4199–4205 (2009)
2. Alippi, C., Roveri, M.: An adaptive CUSUM-based test for signal change detection. In: 2006 IEEE International Symposium on Circuits and Systems, p. 4 (2006)
3. Alissandrakis, A., Nehaniv, C.L., Dautenhahn, K.: Correspondence Mapping Induced State and Action Metrics for Robotic Imitation. Cybernetics 36(3), 1–9 (2006)
4. Antonelo, E.A., Schrauwen, B., Stroobandt, D.: Imitation Learning of an Intelligent Navigation System for Mobile Robots using Reservoir Computing. In: 10th Barazelian Symposium on Neural Networks, pp. 93–98 (2008)
5. Brooks, R.A.: Intelligence without representation. Artificial Intelligence 47, 139–159 (1991)
6. Dufay, B., Latombe, J.C.: An approach to automatic robot programming based on inductive learning. International Journal of Robotics Research 3(4), 3–20 (1984)
7. Iacoboni, M.: Imitation, empathy, and mirror neurons. Annual Review of Psychology 60, 653–670 (2009)
8. Mohammad, Y., Nishida, T.: Constrained motif discovery in time series. New Generation Computing 27(4), 319–346 (2009)
9. Mohammad, Y., Nishida, T.: Robust Singular Spectrum Transform. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 123–132. Springer, Heidelberg (2009)
10. Mohammad, Y., Nishida, T.: On comparing SSA-based change point discovery algorithms. In: IEEE/SICE SII 2011, pp. 938–945 (2011)
11. Mohammad, Y., Nishida, T.: Fluid imitation: Learning from unplanned demonstrations. International Journal of Social Robotics (in press, 2012)
12. Moskvina, V., Zhigljavsky, A.: An algorithm based on singular spectrum analysis for change-point detection. Communications in Statistics.Simulation and Computation 32(4), 319–352 (2003)
13. Nagai, Y., Rohlfing, K.J.: Can Motionese Tell Infants and Robots. What to imitate? In: 4th Inter. Symp. on Imitation in Animals and Artifacts, pp. 299–306 (2007)
14. Spong, M.W., Hutchinson, S., Vidyasagar, M.: Robot Modeling and Control. Wiley (2005)

Near-Optimal Evaluation of Network Survivability under Multi-stage Attacks

Frank Yeong-Sung Lin[1], Pei-Yu Chen[1,2,*], and Quen-Ting Chen[1]

[1] Department of Information Management, National Taiwan University
Taipei, Taiwan, R.O.C.
[2] CyberTrust Technology Institute, Institute for Information Industry
Taipei, Taiwan, R.O.C.
{yslin,d96006,r98043}@im.ntu.edu.tw

Abstract. In this paper, a multi-stage attack-defense model is proposed. We consider cyber attackers and network defenders with complete understanding of the information about each other. In general, in the strategic interaction between cyber attackers and network defenders, both parties repeatedly interact with each other. These interactions should thus not be one-stage but multi-stage. From the network defenders' view, this model is used to support network operators and to predict all the likely strategies used by both cyber attacker and network defender. As a result, the Average Degree of Disconnectivity (Average DOD) is provided as a survivability metric for evaluating the residual network after malicious attacks. To solve the problem, a gradient method and game theory is adopted to find the optimal resource allocation strategies for both cyber attackers and network defenders.

Keywords: Average Degree of Disconnectivity (Average DOD), Gradient Method, Network Survivability, Optimization, Game Theory, Resource Allocation, Network Attack and Defense.

1 Introduction

Network survivability is a concept to evaluate the ability of a network to provide services even in the presence of disturbances of natural or man-made disasters, which is becoming an important issue in network security technology. However, a clear definition of survivability is still being debated. Numerous researchers have been devoted to defining the meaning of network survivability and estimating the impact of external and internal factors on the network survivability [1][2]. How to evaluate the network survivability is a critical issue in the attack-defense model. Traditionally, the Degree of Disconnectivity (DOD) metric which was proposed in [3] is used to measure the damage degree of a network. However, the DOD metric is used under the assumption that the attack is either successful or unsuccessful, which ignores the possibility that the attack might neither be 100% successful nor 100% unsuccessful. Therefore, a

* Corresponding author.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 391–399, 2012.

novel metric which is called Average Degree of Disconnectivity (Average DOD), proposed in [4], is adopted in this paper. Average DOD consists of the concept of attack success probability calculated by contest success function [5] and the concept of DOD metric. The larger the Average DOD value, the smaller the network survivability. This metric carries out the precise description and formal analysis of the network context through the attack-defense scenario.

An attack-defense scenario here is composed of cyber attackers and network defenders, whose interaction is like information warfare. The cyber attackers usually aim to minimize the degree of network survivability, whereas the network defenders always expect to maximize the degree of network survivability. Hence, the attack-defense situation becomes a min-max or max-min problem, which has been treated in recent years by researchers as a kind of attack-defense problem of network security by mathematical programming approaches, such as Game Theory [6][7] and the Lagrangean Relaxation Method [8][9][10].

However, to enhance or reduce network survivability, both network defenders and cyber attackers usually need to invest a fixed number of resources in the network; this is a significant issue concerning how to efficiently allocate scarce resources to the network for cyber attackers and network defenders. Traditionally, the attack-defense resource allocation problem usually discusses only one stage [6][7][8][9]. However, as the interaction frequency between cyber attackers and network defenders is usually more than one time in practice, an increasing number of researchers have begun to discuss multi-stage attack-defense resource allocation issues [10][11][12]. Unfortunately, most existing multi-stage attack-defense resource allocation solutions are not suitable to the field of the network security due to their almost exclusive focus on the attack-defense problems of the parallel systems [11][12] or serial systems [13]. The topology of networks, on the other hand, is usually more complicated than the topology of the parallel, serial or even serial-parallel systems. With these research gaps in mind, a new multi-stage attack-defense model to solve resource allocation problems for both cyber attackers and network defenders is developed in this paper.

2 Problem Description and Formulation

In the attack-defense problem, both cyber attackers and network defenders employ strategies to attain their goals. From the perspective of network defenders, the defenders usually look to minimize the degree of network damage. On the other hand, the cyber attackers try to maximize the damage degree of the network. However, both cyber attackers and network defenders are always limited by the invested resources. It is imperative for both cyber attackers and network defenders to make the decision to efficiently allocate resources to each node. In this light, a new mathematical model to support both cyber attackers and network defenders to make these kinds of optimal decisions needs to be developed.

2.1 Problem Description

A multi-stage attack-defense problem in a mathematical model is considered, since there are multiple interactions between cyber attackers and network defenders in practice. In addition, the damage degree of network is evaluated by the Average DOD value in that the larger the value of the Average DOD, the greater the damage degree of the network. Both cyber attackers and network defenders usually use various strategies to attain their goals. Since both participants are usually constrained by the allocated resources in each stage, they need to determine how to optimally allocate resources to defend or attack targeted network. Details of the model's notations are described in the next section.

2.2 Problem Formulation

The network attack-defense problem is formulated as a mathematical model as follows. Both cyber attackers and network defenders have complete information about the targeted network topology and a budget allocation is assumed. For simplicity, since the targeted network is at the AS level in this model, attackers cannot simply attack any node directly. The notations used in this paper and problem formulation is defined in Table 1.

Table 1. Given Parameters and Decision Variables

Given parameter	
Notation	Description
V	Index set of nodes
R	Index set of stages in the attack and defense actions
$\hat{A}$	Total budget of attacker
$\hat{B}$	Total budget of defender
Decision variable	
Notation	Description
$\vec{a}$	Attacker's budget allocation, which is a vector of attack cost a_1, a_2 to a_i, where $i \in V$
$\vec{b}$	Defender's budget allocation, which is a vector of defense cost a_1, a_2 to a_i, where $i \in V$
a_i	Attacker's budget allocation on node i, where $i \in V$
b_i	Defender's budget allocation on node i, where $i \in V$
A_r	Attacker's attack budget in stage r, where $r \in R$
B_r	Defender's defense budget in stage r, where $r \in R$
$\bar{D}(\vec{a},\vec{b})$	The Average DOD, which is considered under the attacker's and defender's budget allocation, are $\vec{a}$ and $\vec{b}$

The problem is then formulated as the following problem:

Objective function:

$$\min_{\vec{a}} \max_{\vec{b}} \overline{D}(\vec{a}, \vec{b}), \tag{IP 1}$$

Subject to:

$$\sum_{i \in V} b_i \leq B \tag{IP 1.1}$$

$$\sum_{i \in V} a_i \leq A \tag{IP 1.2}$$

$$\sum_{i \in V} A_r \leq \hat{\mathrm{A}} \qquad r \in R \tag{IP 1.3}$$

$$\sum_{i \in V} B_r \leq \widehat{\mathrm{B}} \qquad r \in R. \tag{IP 1.4}$$

The objective function is to minimize the maximum sum of the product of Average DOD and weight in each stage. IP 1.1 describes that the sum of the allocated defense budgets in each node should not exceed the sum of the new allocated and reallocated budgets in that stage. IP 1.2 calculates that the sum of the allocated attack budgets in each node should not exceed the attack budgets in that stage. IP1.3 and IP 1.4 describe that the sum of the allocated defense and attack budgets in each stage should not exceed the total budget of themselves.

3 Solution Approach

In this paper, we combine game theory with the gradient method [14] to find the optimal resource allocation strategy for each node in each stage for both cyber attacker and network defender. The Gradient Method is used to calculate the Average DOD value and to find the optimal resource allocation strategy for each node. To optimally allocate resources in each stage is another issue. Hence, complete and imperfect game theory is adopted to determine the optimal strategy allocation in each stage. The detailed process flow is demonstrated in Figure 1.

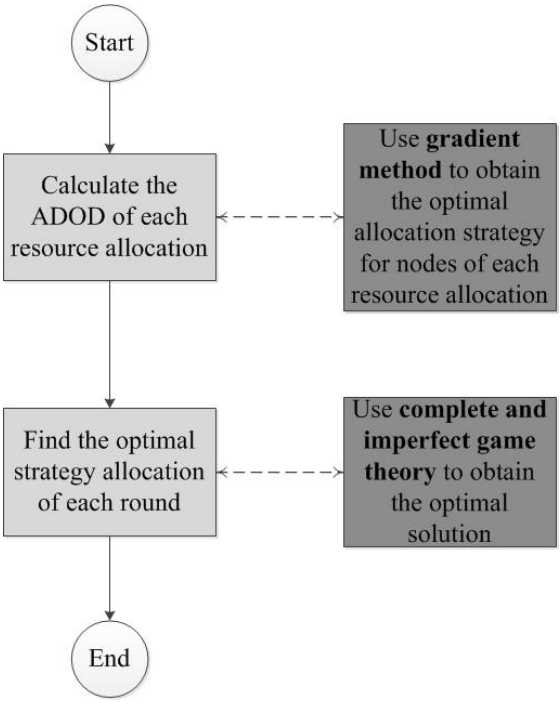


Fig. 1. The Solution Procedure of the Model

3.1 The Gradient Method

The Gradient Method is a general framework used to solve the optimization problems of how is to maximize or minimize functions of continuous parameters. This problem is a min-max formulation and both cyber attacker and network defender are assumed to be able to allocate continuous resources to each node. Here, the gradient method is adopted to solve this problem.

The Gradient Method can usually be categorized into two types, one is gradient descent and the other one is gradient ascent [15]. The gradient descent method can be used to solve the optimal minimization problem. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or of the approximate gradient) of the function at the current point. If instead one takes steps proportional to the positive of the gradient, one approaches a local maximum of that function; the procedure is then known as gradient ascent. The concepts of gradient descent and gradient ascent are extremely similar. We adopt them both in the proposed heuristic. The detailed process flow of the gradient method is also described as Table 2.

Table 2. The Algorithm of the Gradient Method

```
Step1.an initial  point
Step2.Determine a positive or negative direction
Step3.Determine a step size
Step4.Do{
      Find the most impact of all dimensions
      Move a step of the most of all dimensions
      Update an initial  point
      }While( a Given Stop Criterion )
```

3.2 Game Theory

Complete information is a term used in economics and game theory to describe an economic situation or game in which information about other participants is available to all participants [15]. In other words, every participant knows the payoffs and strategies available to the other participants. But each participant may still not be aware of the motions of all the other participants that have already taken place, making it impossible to thus predict the other participants' actions.

Here, the normal form is introduced to solve this problem in this model, which is represented by a matrix which shows the participants, strategies, and payoff values [16]. For two participants, the strategy of one is represented in a column whereas the strategy of the other is represented in a row of a matrix. For example, both participants have five different strategies (S_{11} to S_{15} and S_{21} to S_{25}). The combination of the two participants with different strategies results in 25 (U_{11} to U_{55}) values (the Average DOD values).

According to the above paragraph, cyber attacker and network defender strategies, which involve different percentages of resource allocation in each stage, can be narrated in a matrix. The payoff of all resource allocation strategies of each participant is calculated by the Average DOD. The analysis of the complete and imperfect information game is conducted via heuristics. The solution procedure of the complete and imperfect information game is shown in the following:

Step1. Dominant strategy elimination. The dominant strategy means that no matter what kind of strategy the opponent takes, it is better than other strategies.
Step2. If only one strategy is left for each participant, it is the optimal strategy. Otherwise, go to step 3.
Step3. Use the min-max strategy to find the optimal strategy of each participant. If the min-max strategy still cannot find the optimal strategy, go to step4.
Step4. Use the mixed strategy (Linear programming) to find the optimal strategy for each participant.

4 Computational Experiments

Because of the complexity of this problem, the number of network nodes considered in the experiments is a 9-node grid network. The topology is showed in Figure 2. The number of attack-defense scenarios is discussed as two stages. The experiment is demonstrated in a grid network. The feature of the grid network is that it is an extremely regular network.

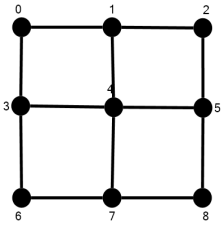


Fig. 2. Grid Network

There are several different policies that the attackers and defenders can take in scenarios presented in this paper, as both participants have 20 units. Their strategies are shown in Table 3. In this experiment, the total stage number is set as 2, with the Grid Network being adopted in this experiment. The experiment results are discussed hereafter.

Table 3. The Experiment under the Grid Network

Strategy		**Attacker**				
		(0,20)	(5,15)	(10,10)	(15,5)	(20,0)
Defender	(0,20)	0.749355	2.979505	2.99328	2.995515	2.996635
	(5,15)	0.854925	1.870855	2.305035	2.448675	2.449975
	(10,10)	0.99734	1.632309	1.872685	2.0907	2.148695
	(15,5)	1.197075	1.624254	1.821465	**1.872965**	1.84314
	(20,0)	1.496635	1.798021	1.997645	2.13829	1.49871

According to the experiment results, the equilibrium point is on both participants' allocating 15 units on stage 1 and 5 units on stage 2. In this two-stage scenario, it is observed that both cyber attackers and network defenders focus on not only stage 1 or stage 2, but both stages. Since the result of two stages may mean they hold the same

importance to them, they may not allocate all resources to just one particular stage. For the network defenders, they intend to reserve some resources for stage 2, in case the cyber attackers still attack the target network in the next stage. As for the cyber attackers, they keep some attack resources in the next stage, in order to prevent the network defenders spending all their resources on the first stage. In that situation, the attack effectiveness is ideal. As a result, the cyber attackers also keep partial attack resources for stage two.

5 Conclusion

In this paper, a model is proposed to solve a multi-stage network-defense problem. We introduce this problem with a mathematical technique and optimally solve it under a two-stage scenario. The complexity of this problem is the evaluation of the entire configurations and probabilities. The problem is optimally solved with the proposed heuristic, with the solution obtained via the gradient method and game theory analysis.

Our results show that network operators should consider each stage of their network planning. The model provides a basis for network operators to decide how to distribute their resources. However, considering the diversity of network problems, this model still makes a number of assumptions. For example, network operators in practice would repair the dysfunction nodes of the network or they could reallocate resources. Moreover, in order to limit the complexity of the problem, the model is demonstrated in only two stages. In a more general case, the experiment could be expanded to n-stages. Hence, we are still working on developing more practical models for this network-defense problem.

Acknowledgments. This research was supported by the National Science Council of Taiwan, Republic of China, under grant NSC-100-2221-E-002-174.

References

1. Westmark, V.R.: A Definition for Information System Survivability. In: Proc. HICSS 12 (2004)
2. Ellison, R.J., Fisher, D.A., Linger, R.C., Lipson, H.F., Longstaff, T., Mead, N.R.: Survivable Network Systems: An Emerging Discipline. Technical Report CMU/SEI-97-TR-013 (November 1997)
3. Lin, F.Y.S., Yen, H.H., Chen, P.Y., Wen, Y.F.: Evaluation of Network Survivability Considering Degree of Disconnectivity. In: Corchado, E., Kurzyński, M., Woźniak, M. (eds.) HAIS 2011, Part I. LNCS (LNAI), vol. 6678, pp. 51–58. Springer, Heidelberg (2011)
4. Lin, F.Y.S., Chen, P.Y., Chen, Q.T.: Resource Allocation Strategies to Maximize Network Survivability Considering of Average DOD. In: Proc. DCAI 2012 (2012)
5. Skaperdas, S.: Contest Success Functions. Economic Theory (1996)
6. Jiang, W., Fang, B.X., Zhang, H.L., Tian, Z.H.: A Game Theoretic Method for Decision and Analysis of the Optimal Active Defense Strategy. In: Proc. CIS 2007 (2007)

7. Jiang, W., Fang, B.X., Zhang, H.L., Tian, Z.H.: Optimal Network Security Strengthening Using Attack-Defense Game Model. In: Proc. ITNG 2009 (2009)
8. Lin, Y.S., Tsang, P.H., Chen, C.H., Tseng, C.L., Lin, Y.L.: Evaluation of Network Robustness for Given Defense Resource Allocation Strategies. In: Proc. ARES 2005 (2005)
9. Lin, F.Y.-S., Yen, H.-H., Chen, P.-Y., Wen, Y.-F.: Evaluation of Network Survivability Considering Degree of Disconnectivity. In: Corchado, E., Kurzyński, M., Woźniak, M. (eds.) HAIS 2011, Part I. LNCS (LNAI), vol. 6678, pp. 51–58. Springer, Heidelberg (2011)
10. Lin, F.Y.S., Chen, P.Y., Wang, Y.S., Chang, Y.Y.: Network Recovery Strategies for Maximization of Network Survivability under Malicious Attacks. In: Proc. IEEE IWCMC 2011 (2011)
11. Levitin, G., Hausken, K.: Parallel Systems under Two Sequential Attacks. Reliability Engineering and System Safety 94(3), 763–772 (2009)
12. Levitin, G., Hausken, K.: Resource Distribution in Multiple Attacks against a Single Target. Risk Analysis 30(8), 1231–1239 (2010)
13. Levitin, G., Hausken, K.: Protection vs. False Targets in Series Systems. Reliability Engineering and System Safety 94(5), 973–981 (2009)
14. Hassoun, H.: Fundamentals of Artificial Neural Networks. MIT Press (1995)
15. Owen, G.: Game Theory, 3rd edn. Academic Press (2001)
16. Fudenberg, D., Tirole, J.: Game Theory. MIT Press (1995)

Solving Manufacturing Cell Design Problems Using Constraint Programming

Ricardo Soto[1], Hakan Kjellerstrand[2], Juan Gutiérrez[1], Alexis López[1], Broderick Crawford[1], and Eric Monfroy[3]

[1] Pontificia Universidad Católica de Valparaíso, Chile
[2] Independent Researcher, Sweden
www.hakank.org
[3] CNRS, LINA, Université de Nantes, France
{ricardo.soto,broderick.crawford}@ucv.cl, hakank@gmail.com,
{juan.gutierrez.q,alexis.lopez.e}@mail.pucv.c,
eric.monfroy@univ-nantes.fr

Abstract. A manufacturing cell design problem (MCDP) consists in creating an optimal production plant layout. The production plant is composed of cells which in turn are composed of machines that process part families of products. The goal is to minimize part flow among cells in order to reduce production costs and increase productivity. In this paper, we focus on modeling and solving the MCDP by using state-of-the-art constraint programming (CP) techniques. We implement different optimization models and we solve it by using two solving engines. Our preliminary results demonstrate the efficiency of the proposed implementations, indeed the global optima is reached in all instances and in competitive runtime.

1 Introduction

Manufacturing cell design consists in dividing a plant in a set of cells, that process part families, in such a way the part flow among cells in minimized. A part family corresponds to a group of parts requiring similar and often identical operation processes, materials, and tools. It is well known that manufacturing cells provide considerable cost and productivity benefits to practical manufacturing environments [17]. A main problem in cell formation is the identification of machine and component groups. This identification process requires an effective approach to form part families so that similarity within a part family can be maximized. According to [17], clustering analysis is one of the most used methods for manufacturing cell design. However, since the cell formation problem is known to be NP-complete, there is still the challenge of creating an efficient clustering method.

This paper focuses on the efficient solving of manufacturing cell formation problems (MCDP). We introduce constraint programming (CP) as powerful techniques to tackle this problem. We implement different models by using different CP-based global optimizers. Those implementations demonstrate the

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 400–406, 2012.

efficiency and feasibility of using this technology for tackling cell formation problems. Indeed, all implementations are able to reach the global optimum in all instances and most of them in excellent runtime. This paper is organized as follows: Section 3 gives an overview of CP. MCDP is described and modeled in Section 4. Experimental results are presented and discussed in Section 5. Finally, we conclude and we give some directions for future work.

2 Related Work

The research work devoted to MCDP has been focused on different techniques, mainly coming from global optimization and approximate methods. For instance, in the context of approximate methods, Aljaber et al. [2] and Lozano et al. [10] presented tabu search approaches. Simulated annealing and genetic algorithms approaches for MCDP can be found in [19] and [18], respectively. In [6], Durán et al. proposed a particle swarm optimization algorithm enhanced with a data mining technique. Hybridizations also exists, e.g. genetic algorithms combined with simulated annealing [19] and with local search [8]. In the global optimization field, experiments have been mostly centered on mathematical programming, some examples are the works on linear programming [14,12] and on quadratic programming [3,9]. A hybridization can also be found in [4], which proposes a combination of genetic algorithms with branch-and-bound techniques. In this paper, we focus on unexplored global optimization techniques –in the context of MCDP– namely constraint programming, which to the best of our knowledge, have not been applied yet to MCDP.

3 Constraint Programming

Constraint Programming is a powerful programming paradigm devoted to the efficient resolution of constraint-based problems. In CP, a problem is formulated as a Constraint Satisfaction Problem (CSP). Formally, a CSP $\mathcal{P}$ is defined by a triple $P = \langle \mathcal{X}, D, C \rangle$ where: $\mathcal{X}$ is an n-tuple of variables $\mathcal{X} = \langle x_1, x_2, \ldots, x_n \rangle$. $\mathcal{D}$ is a corresponding n-tuple of domains $\mathcal{D} = \langle d_1, d_2, \ldots, d_n \rangle$ such that $x_i \in d_i$, and d_i is a set of values, for $i = 1, \ldots, n$. $\mathcal{C}$ is an m-tuple of constraints $\mathcal{C} = \langle c_1, c_2, \ldots, c_m \rangle$, and a constraint c_j is defined as a subset of the Cartesian product of domains $d_{j_1} \times \cdots \times d_{j_{n_j}}$, for $j = 1, \ldots, m$.

A solution to a CSP is an assignment $\{x_1 \rightarrow a_1, \ldots, x_n \rightarrow a_n\}$ such that $a_i \in d_i$ for $i = 1, \ldots, n$ and $(a_{j_1}, \ldots, a_{j_{n_j}}) \in c_j$, for $j = 1, \ldots, m$. In a CP context, an optimization problem is simply an extension of a CSP, which is called COP (Constrained Optimization Problem). Formally, a COP $\mathcal{P}$ is defined by a 4-tuple $P = \langle \mathcal{X}, D, C, f \rangle$, where f is the objective function to be maximized or minimized.

The basic CP idea for solving CSPs is to build a tree-data structure holding the potential solutions by interleaving two main phases: enumeration and

Algorithm 1
Input: $\mathcal{C}$, $\mathcal{D}$
1 **While** ¬success **or** failure **do**
2 **Variable_Selection**($\mathcal{D}$)
3 **Value_Selection**($\mathcal{D}$)
4 **Propagate**$_{\mathcal{C}}$($\mathcal{D}$)
5 **If** empty_domain_in_future_var
6 **Shallow_Backtrack**()
7 **If** empty_domain_in_current_var
8 **Backtrack**()
9 **End While**

Algorithm 2
Input: f, $\mathcal{C}$, $\mathcal{D}$
1 $m \leftarrow +\infty$
2 **While** ¬success **or** failure **do**
3 **Variable_Selection**($\mathcal{D}$)
4 **Value_Selection**($\mathcal{D}$)
5 **Propagate**$_{\mathcal{C}\cup\{f(\mathbf{x})\leq m\}}$($\mathcal{D}$)
6 $m \leftarrow$**Update**($\mathcal{D}$, f)
7 **If** empty_domain_in_future_var
8 **Shallow_Backtrack**()
9 **If** empty_domain_in_current_var
10 **Backtrack**()
11 **End While**

Fig. 1. Left: The branch and prune algorithm. Right: The branch and bound algorithm.

propagation. In the enumeration phase, a variable and a value from its domain are chosen to create a tree branch. In the propagation phase, a consistency property is enforced to prune the tree, i.e., the values that do not lead to any solution are temporarily deleted from domains (see [5] for a detailed description of constraint propagation). In this way, the exploration does not inspect unfeasible instantiations, which speeds-up the whole process.

Algorithm 1 illustrates a general procedure for solving CSPs. The goal is to iteratively generate partial solutions, backtracking when an inconsistency is detected, until a result is reached. The algorithm has as input the set of constraints and domains. Then, a while loop encloses a set of actions to be performed until success (i.e. a solution is reached) or a failure is detected (i.e. no solution is found). The first two enclosed actions correspond to the variable and value selection, which are driven by the variable and value ordering heuristics, respectively. The third action is a call to a propagation procedure, which is responsible for attempting to prune the tree. Finally two conditions are included to perform backtracks. A shallow backtrack corresponds to try the next value available from the domain of the current variable, and the backtracking returns to the most recently instantiated variable that has still values to reach a solution.

Algorithm 2 is a CP-based branch and bound algorithm for handling Constraint Optimizations Problems (COPs), in particular minimization problems (maximization problems are handled similarly). It is a slight modification of algorithm 1. It includes a cost function as an additional input and maintains an upper bound on the global minimum in the variable m (which is initialized to $+\infty$ at the beginning of the search). This upper bound is used to discard parts of the search space whose cost is larger than it by adding $f(x) \leq m$ to the set of constraints. The idea is to propagate involving the cost function (line 5). Finally, the upper bound is updated whether a better solution has been found (line 6).

4 Problem Statement

The problem is represented by the following mathematical model:

$$\texttt{minimize} \quad \sum_{k=1}^{C} \sum_{i=1}^{M} \sum_{j=1}^{P} a_{ij} z_{jk} (1 - y_{ik})$$

subject to

$$\sum_{k=1}^{C} y_{ik} = 1 \quad \forall_i, \qquad \sum_{k=1}^{C} z_{jk} = 1 \quad \forall_j, \qquad \sum_{i=1}^{M} y_{ik} \leq M_{max} \quad \forall_k,$$

where:

- M is the number of machines,
- P, the number of parts,
- C, the number of cells,
- i, the index of machines ($i = 1, \ldots, M$),
- j, the index of parts ($j = 1, \ldots, P$),
- k, the index of cells ($k = 1, \ldots, C$),
- M_{max}, the maximum number of machines per cell.
- a_{ij}, the $M \times P$ machine-part matrix, where:

$$a_{ij} = \begin{cases} 1 \text{ if } j_{th} \text{ part visits } i_{th} \text{ machine;} \\ 0 \text{ otherwise.} \end{cases}$$

- y_{ik}, the $M \times C$ machine-cell matrix, where:

$$y_{ik} = \begin{cases} 1 \text{ if machine } i \in \text{cell } k; \\ 0 \text{ otherwise;} \end{cases}$$

- z_{jk}, the $P \times C$ part-cell matrix, where:

$$z_{jk} = \begin{cases} 1 \text{ if part } j \in \text{family } k; \\ 0 \text{ otherwise;} \end{cases}$$

5 Experiments

We have performed a set of experiments in order to exhibit the efficiency of the proposed implementations. We have designed and launched five different MCDP models in two CP-based solvers (Gecode [16] and Ecl^{i}ps^e [13], using MiniZinc [11] as modeling language) considering four enumeration strategies, coming from the combination of two variable ordering heuristics (anti-first-fail[1] and occurrence) and two value ordering heuristics (min, max). The description of models is given in table 1 and the source code of models is available in [1].

[1] Let us note that this problem owns a Boolean domain for all variables, so first-fail and anti-first-fail act equally.

We have tested 50 problems (10 instances considering 5 values of M_{max}, see table 2) on a 2.80GHz Intel Core i7 930 with 12Gb RAM running Ubuntu Linux. The global amount of experiments is about 2000, considering 5 models, 2 solvers, 4 enumeration strategies, and 50 problems. Such 50 problems have been taken from Boctor's experiments [3] in order to compare it with previous work. Table 2 contrasts the optimum value reached by using different techniques for the 50 problems. For each M_{max}, column 1 (Op) depicts the optimum value for the given problem, column 2 (CP) the optimum value reached using our CP implementations, column 3 (SA) the optimum value using Simulated Annealing, and column 4 (PSO) the optimum value using Particle Swarm Optimization. Here, it is possible to see that our approach is unique in reaching the global optimum for all instances.

Table 1. Different models implemented

MCDP1	it exactly corresponds to the mathematical model presented in section 4.
MCDP2	it is an extension of MCDP1 that includes a dual representation of the problem. Two additional arrays are added in order to identify the machine assignments (i.e. which machines have been assigned to the $y_{i,k}$ matrix) as well as the part assignments (i.e. which parts have been assigned to the $z_{j,k}$ matrix).
MCDP3	this model skips the initial matrix representation of MCDP1 and MCDP2. Decision variables are now completely handled via lists including the dual representation of MCDP2. This allows one to introduce an extra heuristic for the constraints that ensure that a part should be in the same cell as one of its machines, and vice versa.
MCDP4	this model is an extension of MCDP3 that includes symmetry breaking [7] in order to reduce the search space. The idea is to force using cells in order e.g., if there is a machine assigned to cell k, then there must be a machine assigned to cell $k-1$.
MCDP5	this model implements set instead of single decision variables. It maintains the dual representation of MCDP2 and the extra heuristic of MCDP3. Additionally, it involves the use of the `partition`[2] and `all_different`[3] global constraints for speeding up the solving process.

Now, considering solving times, the best time reached by Ecl^{i}ps^e for solving the complete set of 50 problems was 817.02 seconds, that means an average of 16,34 seconds per problem. The best time for Gecode was 181.99 seconds, achieving an excellent average of 3,63 seconds per problem. Best times for both solvers were reached by using the MCDP4 model, the occurrence variable-ordering, and the max value-ordering.

[2] The `partition` constraint forces to partition a given universe into disjoint sets.

[3] The `alldifferent`$(X_1, ..., X_n)$ constraint specifies that the values assigned to the variables $X_1, ..., X_n$ must be pairwise distinct [15].

Table 2. Optimum values for CP (CP), Simulated annealing (SA), and Particle swarm optimization (PSO)

	$M_{max} = 8$				$M_{max} = 9$				$M_{max} = 10$				$M_{max} = 11$				$M_{max} = 12$			
Instance	Op	CP	SA	PSO	Op	CP	SA	PSO	Op	CP	SA	PSO	Op	CP	SA	PSO	Op	CP	SA	PSO
1	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
2	7	7	7	7	6	6	6	6	4	4	10	5	3	3	3	4	3	3	3	4
3	4	4	5	5	4	4	4	4	4	4	4	5	3	3	4	4	1	1	4	3
4	14	14	14	15	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
5	9	9	9	10	6	6	6	8	6	6	6	6	5	5	7	5	4	4	4	5
6	5	5	5	5	3	3	3	3	3	3	3	3	3	3	3	4	2	2	3	4
7	7	7	7	7	4	4	4	5	4	4	4	5	4	4	4	5	4	4	4	5
8	13	13	13	14	10	10	20	11	8	8	20	10	5	5	11	6	5	5	7	6
9	8	8	13	9	8	8	8	8	8	8	8	8	5	5	8	5	5	5	8	8
10	8	8	8	9	5	5	5	8	5	5	5	7	5	5	5	7	5	5	5	6

6 Conclusions

In this paper, we have modeled and solved the MCDP by using state-of-the-art constraint programming technology. We have implemented five different models and we have tested them in Ecl^{i}ps^e and Gecode, being both well-known solvers within the CP community. The results demonstrated the feasibility of using CP for solving cell design problems. Indeed, the global optimum was reached in all instances and in excellent runtime.

The results illustrated here correspond to ongoing work, and they can clearly be extended for instance by testing new solving technology, e.g. SAT-based solvers or hybrids of CP+SAT. Autonomous search should also be an interesting approach to experiment.

References

1. Manufacturing Cell Design Models, `http://www.inf.ucv.cl/~rsoto/MCDP` (visited 11/2011)
2. Aljaber, N., Baek, W., Chen, C.: A tabu search approach to the cell formation problem. Computers and Industrial Engineering 32(1), 169–185 (1997)
3. Boctor, F.: A linear formulation of the machine-part cell formation problem. International Journal Production Research 29(2), 343–356 (1991)
4. Boulif, M., Atif, K.: A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem. Computers and Operations Research 33, 2219–2245 (2006)
5. Bessière, C.: Constraint Propagation. In: Handbook of Constraint Programming, pp. 29–84. Elsevier (2006)
6. Durán, O., Rodriguez, N., Consalter, L.A.: Collaborative particle swarm optimization with a data mining technique for manufacturing cell design. Expert Syst. Appl. 37(2), 1563–1567 (2010)
7. Gent, I.P., Smith, B.M.: Symmetry breaking in constraint programming. In: Proceedings of the 14th ECAI, pp. 599–603. IOS Press (2000)
8. James, T., Brown, E., Keeling, K.: A hybrid grouping genetic algorithm for the cell formation problem. Computers and Operations Research 34(7), 2059–2079 (2007)

9. Kusiak, A., Chow, W.S.: Efficient solving of the group technology problem. Journal of Manufacturing Systems 6, 117–124 (1987)
10. Lozano, S., Díaz, A., Eguía, I., Onieva, L.: A one-step tabu search algorithm for manufacturing cell design. Journal of the Operational Research Society 50(5) (1999)
11. Nethercote, N., Stuckey, P.J., Becket, R., Brand, S., Duck, G.J., Tack, G.: MiniZinc: Towards a Standard CP Modelling Language. In: Bessière, C. (ed.) CP 2007. LNCS, vol. 4741, pp. 529–543. Springer, Heidelberg (2007)
12. Olivia-Lopez, E., Purcheck, G.F.K.: Load balancing for group technology planning and control. International Journal of MTDR 19, 259–268 (1979)
13. Puget, J.F.: A C++ implementation of CLP. In: SCIS, Singapore (1994)
14. Purcheck, G.F.K.: A linear - programming method for the combinatorial grouping of an incomplete set. Journal of cybernetics 5, 51–58 (1975)
15. Régin, J.-C.: A Filtering Algorithm for Constraints of Difference in CSPs. In: Proceedings of the 12th AAAI, pp. 362–367 (1994)
16. Schulte, C., Tack, G.: Views and Iterators for Generic Constraint Implementations. In: van Beek, P. (ed.) CP 2005. LNCS, vol. 3709, pp. 817–821. Springer, Heidelberg (2005)
17. Selim, H., Askin, R., Vakharia, A.: Cell formation in group technology: review, evaluation and directions for future research. Computers and Industrial Engineering 34(1), 3–20 (1998)
18. Venugopal, V., Narendran, T.T.: A genetic algorithm approach to the machine-component grouping problem with multiple objectives. Computers and Industrial Engineering 22(4), 469–480 (1992)
19. Wu, T., Chang, C., Chung, S.: A simulated annealing algorithm for manufacturing cell formation problems. Expert Syst. Appl. 34(3), 1609–1617 (2008)

A Pragmatic Approach to Grounding Data Computation for Modal Statements

Wojciech Lorkiewicz, Grzegorz Skorupa, and Radoslaw Katarzyniak

Institute of Informatics
Wroclaw University of Technology
{wojciech.lorkiewicz,grzegorz.skorupa,radoslaw.katarzyniak}@pwr.wroc.pl

Abstract. In this paper we focus on the problem of grounding modal statements in an agent system. We follow the Grounding Theory, formally defining an approach to grounding based on empirical data collected by a cognitive agent. Performed analysis is done in a context of incomplete observations, where partial empirical data should influence agent's cognitive attitude. In particular, we focus solely on modal conjunctions and formulate a set of common-sense requirements for incorporating incomplete (partial) empirical data. We further propose a coherent and unified extensions to the Grounding Theory that compensates for the existence of partial observations, i.e., influencing the cognitive stance of the agent. Finally, we outline a few examples to illustrate the underlying modifications of the existing theory.

Keywords: multi-agent system, cognitive agent, language grounding, modal statements.

1 Introduction

Here we incorporate the Grounding Theory model [5,6,7], where the process of grounding of modal statements follows the Dennett's idea that multiple 'exposure to x – that is, sensory confrontation with x over suitable period of time – is the normally sufficient condition for knowing (or having true beliefs) about x'[1]. This approach investigates common-sense requirements for uttering different levels of certainty, i.e. possibility, belief and knowledge, and encapsulates them in a well-defined and self-contained system. In particular, the Grounding Theory defines a mechanism for an agent to fill in the unobserved parts of the environment with cognitive schema extracted from its past empirical experiences. For instance, every observation in which an agent observed an object exhibiting a property (not exhibiting a property) makes a corresponding cognitive scheme stronger (weaker) in relation to all complementary schemes. These strengths, along with a system of modal levels, serve as means for grounding adequate, i.e., in compliance with common-sense, modal statements.

In this paper we address the problem of partial observations, i.e., observations that involve neither a complete nor fully incomplete observation of a particular object. We claim that such observations should affect the cognitive stance of

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 407–416, 2012.

an individual agent. In particular, we present a short example pinpointing that such a neglection is counter-intuitive and can results in grounding of unreasonable modal statements, i.e., not adequate to the gathered empirical knowledge. Our main goal is to compensate for experiences gathered from the partial observations. We present a study of the influence of these type of observations on the agent's cognitive stance. In particular, we introduce a set of common-sense postulates that are required for a proper grounding of modal statements in the presence of partial observations. In essence, we extend the classical model defined in the Grounding Theory by incorporating a simple modification to the process of calculating the strengths of cognitive schemes.

2 Grounding of Modal Statements

Symbol grounding [3] is a task of finding the embodied meaning of symbols. For a symbol to have a meaning, it has to be recognised by an observer and related to its embodied knowledge (i.e. empirical experiences). In particular, an intelligent agent forms the so called semiotic triangle [9] by setting up a proper relation between objects in the environment and recognised symbols (representing them). Such relation can be build based on past perceptions of the environment that are gathered by an agent and form its empirical knowledge. Multiple perceptions of a particular object are fundamental for forming beliefs and intentions related with it [1,2]. Such beliefs form generalised knowledge, that is further extrapolated to mental models of agent's current and future perceptions of the environment. Such an approach significantly differs from the traditional solutions, where the interpretations of symbols are parasitic to the system, i.e., unknown to the system, and where formal logical theories are used to transform and analyse them.

Works [5,6,7] propose a solution to the aforementioned problem for a chosen class of modal statements. In particular, statements with intentional stances understood as modal operators of possibility (Pos), belief (Bel) and knowledge ($Know$). The presented theory can be classified as a cognitive and phenomenological approach to building the true meaning of statements. As such, agent's perceptions of the external world are used to construct a cognitive stance of the currently observed situation. This cognitive stance is autonomously developed by the agent and is entirely private. Further, such a cognitive stance supplies basis for building proper relations between objects in the environment and recognised statements, i.e., forming a particular semiotic triangle.

Table 1. Exemplary statements and their semantics

formal statement	intuitive semantics
$Bel(p(o))$	I believe, that object o is p.
$Know(p(o) \wedge q(o))$	I know, that object o is p and q.
$Pos(p(o) \wedge q(o))$	It is possible, that object o is p and q.

Moreover, the theory presented in [5,6,7] explores a set of both, simple and complex modal statements - some of them have been presented in tab. 1. All of the incorporated statements should be understood as referring to the currently observed situation. Crucially, semantics of these modal statements is consistent with the natural language understanding of modal operators [4] and conjunctions. In particular, it should not be misunderstood with the formal modal logics based on the possible worlds [8] semantics or classical truth tables.

2.1 Agent and It's Environment

Agent is situated in an environment and it's perceptive abilities are limited to recognition of objects and recognition of their binary properties. Let O denote a set of perceptually available objects and P denote a set of perceptually recognisable properties of objects. As the environment is not-static the objects change their properties over time. In particular, at each discrete moment of time the agent observes some part of the environment. The observation for moment t is stored in agent's embodied structures as a tuple:

$$BP(t) = (O, P_1^+(t), P_1^-(t), P_2^+(t), P_2^-(t), ..., P_K^+(t), P_K^-(t)) \tag{1}$$

Within $P_k^+(t) \subseteq O$ agent holds all objects' perceptions having a property $p_k \in P$ at time t and $P_k^-(t) \subseteq O$ holds objects' perceptions not having a property p_k at t. Agent is unable to directly observe all properties, as such if an object is neither in $P_k^+(t)$ nor in $P_k^-(t)$, then the agent could not measure feature p_k of that object at a moment t. Such a situation can happen due to physical limitations of an agent or bad quality of the received data. Set $P_k^\pm(t)$ holds all objects' perceptions where property p_k was unknown to the agent at a moment t. We assume that sets $P_k^+(t), P_k^-(t), P_k^\pm(t)$ meet constraints in equations 2.

$$P_k^+(t) \cup P_k^-(t) \subseteq O, \quad P_k^+(t) \cap P_k^-(t) = \emptyset, \quad P_k^\pm(t) = O \setminus (P_k^+(t) \cup P_k^-(t)) \tag{2}$$

Observations from every time moment are gathered together within agent's memory as a set $ST(t)$ (see eq. 3). The set $ST(t)$ forms agent's empirical knowledge about the environment gathered until the moment t.

$$ST(t) = \{BP(\hat{t}) : \hat{t} \leq t\} \tag{3}$$

2.2 Agent and It's Cognitive Organisation

We analyse cognitive approach to communication presented in [5,6,7]. In particular, we are considering statements in a form of modal conjunctions: $Pos(p(o) \wedge q(o)), Bel(p(o) \wedge q(o)), Know(p(o) \wedge q(o))$, where $p, q \in P$.

The theory proposed in [5,6,7] introduces a model of environment as described in sec. 2.1. Agent's physical experiences (perceptions) gathered from the environment provide an empirical material crucial for grounding of modal statements. Agent, as an autonomous entity, evaluates empirical data and forms a subjective cognitive stance of a situation. Model of cognitive stance C contains the

empirical material involved in building the relation between an object, from the physical environment, and a modal statement of the considered language.

$$C = C^{++} \cup C^{+-} \cup C^{-+} \cup C^{--} \tag{4}$$

$$C^{xy} = \{BP(t) : o \in P^x(t) \wedge o \in Q^y(t)\} \subseteq ST(t), \text{ where } x, y \in \{+, -\} \tag{5}$$

Cognitive stance is divided into four partitions $C^{++}, C^{+-}, C^{-+}, C^{--}$ (see eq. 5). Sets $P^x(t)$ and $Q^x(t)$ denote the perceptions of properties p and q, respectively. Each partition, called grounding set, contains empirical material gathered by the agent and supporting one of the four possible situations. For example the set C^{++} contains perceptions where an object o exhibited both, p and q, features in the past.

Grounding sets are used to measure the subjective strength of each experience type. The stronger the $\lambda^{xy} \in [0, 1]$ (see eq. 6) the more influential the situation reflected by C^{xy} is. Measure λ^{xy} is used to extrapolate the previous perceptions of the environment to the current observation.

$$\lambda^{xy} = \frac{card(C^{xy})}{card(C)}, \text{ where } x, y \in \{+, -\} \tag{6}$$

Later the theory [5,6,7] introduces a formal criteria defining the relation (called epistemic satisfaction relation) between cognitive stance and various modal statements. These formal criteria result from common sense constraints that are characteristic for natural language understanding and usage[1]. Definitions 1-3 taken from [5,6,7] present such a criteria for modal conjunctions.

Definition 1. *Epistemic relation* $ST(t) \models^E Know(p(o) \wedge q(o))$ *holds iff*

$$o \in P^+(t) \wedge o \in Q^+(t) \quad or \quad o \in P^{\pm}(t) \wedge o \in Q^{\pm}(t) \wedge \lambda^{++} = 1$$

Definition 2. *Epistemic relation* $ST(t) \models^E Bel(p(o) \wedge q(o))$ *holds iff*

$$o \in P^{\pm}(t) \ \wedge \ o \in Q^{\pm}(t) \ \wedge \ \lambda_{minBel} \leq \lambda^{++} < \lambda_{maxBel}$$

Definition 3. *Epistemic relation* $ST(t) \models^E Pos(p(o) \wedge q(o))$ *holds iff*

$$o \in P^{\pm}(t) \ \wedge \ o \in Q^{\pm}(t) \ \wedge \ \lambda_{minPos} < \lambda^{++} < \lambda_{minBel}$$

In laymans terms the epistemic relation for $Know$ operator is met when either, the agent observed o being p and q at moment t, or the properties have not been observed and the object always exhibited these properties in the past. Epistemic relation for Bel operator is met when properties have not been observed at moment t and they happened often in the past. Epistemic relation for Pos operator is met when properties have not been observed at moment t and they happened sometimes in the past.

Parameters $0 \leq \lambda_{minPos} < \lambda_{minBel} < \lambda_{maxBel} \leq 1$ are fixed and should be set according to the limitations presented in [5,6,7][2].

[1] As opposed to absolute truths or formal modal logics.

[2] Additionally it is assumed $\lambda_{maxPos} = \lambda_{minBel}$.

3 Pragmatic Limitations

Grounding the strongest belief of knowledge requires a substantial evidence. In particular, expressing such a certainty should involve a specific, strictly restricted, cognitive stance of the speaker. These fundamental confinements maintain a precise understanding of basic statements like 'I know that object o exhibits property p' ($Know(p(o))$) or 'I know that object o exhibits property p and property q' ($Know(p(o) \wedge q(o))$).

In order to maintain clarity, we further define the notions of complete, partial and incomplete observations:

Definition 4. *At a time point $t \in T$, given properties $p, q \in P$, and a given base profile $BP(t)$ we call the observation:*

- *a p-defined observation, iff $o \in P^x(t)$, where $x \in \{+, -\}$*
- *a p-undefined observation, iff $o \in P^{\pm}(t)$*
- *a pq-complete observation, iff it is p- and q-defined observation*
- *a pq-partial observation, iff it is p-defined and q-undefined or p-undefined and q-defined*
- *a pq-incomplete observation, iff it is p-undefined and q-undefined*

Focusing on modal conjunctions, the Grounding Theory specifies two underlying stances that allow a proper grounding of the belief of knowledge (e.g. $Know(p(o) \wedge q(o))$). First, reflecting the *pq*-complete observation of the properties in a given object o and at a particular point of time t. Second, reflecting the *pq*-incomplete observation with all past *pq*-complete observations supporting *only* the appropriate model[3]. The former situation is obvious, as an agent that directly perceives a particular state of the external environment can, without any doubt, utter its highest certainty. However, the latter situation seems to be questionable, as it completely neglects the existence of partial observations, i.e. observations in which one of the properties is not directly observed ($o \in P^{\pm} \veebar o \in Q^{\pm}$).

To visualise the problem of incomplete observations let us consider a simple example (See Table 2). In particular, focusing on experience limited to 10 observations of an environment comprised of a single object o with two properties p and q. Further, let us assume that both features p and q are directly observed only in $t = \{1, 3, 7, 8\}$ and in these observations object o always exhibited properties p and q. At $t = 11$, the agent can conclude that it lacks a direct complete observation of o ($o \in P^{\pm}(t)$ and $o \in Q^{\pm}(t)$) coinciding with past complete observations ($\{1, 3, 7, 8\}$) that all involve o exhibiting both, p and q. Consequently, according to the Grounding Theory the set C equals C^{++} and $\lambda^{++} = 1$ (See Section 2), allowing the agent to properly ground $Know(p \wedge q)$.

However, a deeper analysis discloses that in partial observations at $t = \{2, 6\}$ object is known not to exhibit property q (the state of p is unknown). As such,

[3] For instance, grounding $Know(p(o) \wedge q(o))$ in time point t requires that $\forall_{\hat{t}<t}\, o \notin P^{\pm}(\hat{t}) \wedge o \notin Q^{\pm}(\hat{t}) \Rightarrow o \in P^{+}(\hat{t}) \wedge o \in Q^{+}(\hat{t})$.

Table 2. Exemplary experience history

t	1	2	3	4	5	6	7	8	9	10	11
p	+		+	+	+		+	+			
q	+	-	+			-	+	+	+	+	

the agent has a direct observation of o not exhibiting property q. Moreover, despite the underlying unobserved realisation of property p in the external world, the agent already has a past experience that explicitly contradicts the statement $Know(p \wedge q)$. This directly leads to the conclusion that the agent (with experience defined in Table 2) should not state it's certainty considering the property q, and the epistemic satisfaction relation for statement $Know(p \wedge q)$ should not be met.

In the following section, we address the identified problem of partial observations in the process of grounding modal statements.

4 Pragmatic Extension

Despite the fact that partial observations do not provide a direct evidence of a particular state of the external world, they are still a valuable source of information (See Sec.3). Moreover, certain situations of their neglection result in grounding of counter-intuitive modal statements, in particular, statements that are not adequate to the gathered empirical experience (See Table 2).

As aforementioned, partial observations *should* affect the cognitive stance of an individual agent. Following this idea we introduce a set of common-sense postulates against partial observations that are required for a proper grounding mechanism. Further, we extend the classical model defined in the Grounding Theory by incorporating a few basic modifications to the definition of the epistemic satisfaction relation of $Know$, extension to the cardinality of grounding sets C^{xy}, and modification to the calculation of relative grounding strengths λ^{xy}.

Whenever the agent (focused on a pair of properties p and q) registers a only p-defined (or q-defined) observation, it automatically activates the mental models of possible realisations of the q-undefined (p-undefined) state. The strength of such activation depends on the gathered experience from the pq-complete observations, as they serve as the only means of precise knowledge.

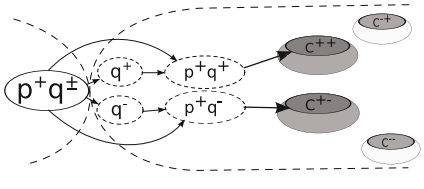


Fig. 1. Partial observation influence on the cognitive stance of the agent

In general, we further extend the procedure of grounding of modal conjunctions to compensate for the existence of partial observations, both in current, and past

experiences. Proposed approach assumes that the extension of the classical grounding process of modal conjunctions should follow a fundamental set of guidelines. We propose five postulates, that underlie the essences of the intended changes:

P1: Having a past experience that explicitly contradicts the statement of knowledge, e.g. $Know(p \wedge q)$, should result in not grounding this statement.
P2: Having a current pq-partial observation should allow the agent to ground it's beliefs concerning the conjunction of p and q.
P3: Past pq-partial observations should affect agent's cognitive stance, such that the relative strength of a model should be lowered by opposing observations.
P4: Lack of pq-complete observations should result in the inability of the agent to ground statements that relate to conjunctions of properties p and q.
P5: Lack of pq-partial observations and pq-incomplete observations should result in classical definitions of grounding of modal conjunctions.

First of all it should be stressed, that in the classical model of grounding the agent registering a pq-partial observation is not able to ground a modal conjunction statement (See P2). However, due to the co-activation of mental models for modal conjunctions (See Fig.1) we claim that even though the agent registered the state of one of the observed properties it is still able to reason about the state of both properties. Second, in a situation that the agent lacks pq-complete observations, consequently lacks the adequate mental models, any of the partial observations is unable to trigger any of the C^{xy} (See P4). Third, postulates 1, 3 and 5 are a direct consequence of the proposed approach (See Fig.1).

4.1 Modification

In the proposed model we assume a uniform distribution of probability of observation, i.e., *only observing* a particular property in a given object. Consequently, having enough observations allows the agent to properly estimate the underlying tendencies in the external world.

The cardinalities of the grounding sets depend strictly on the current situation. As such, the strength of a particular mental model triggered in the agent is shaped by the collected experience and modulated by the current observation:

Definition 5. *For $BP(t)$ being the current observation we can define the strength ϕ of grounding sets as follows:*

$$\phi^{++} = \begin{cases} card(C^{++}) + \mu_p^+ card(C^{+\pm}) + \mu_q^+ card(C^{\pm+}) & o \notin P^-(t) \wedge o \notin Q^-(t) \\ 0 & otherwise \end{cases}$$

$$\phi^{+-} = \begin{cases} card(C^{+-}) + \mu_p^- card(C^{+\pm}) + \mu_{\neg q}^+ card(C^{\pm-}) & o \notin P^-(t) \wedge o \notin Q^+(t) \\ 0 & otherwise \end{cases}$$

$$\phi^{-+} = \begin{cases} card(C^{-+}) + \mu_{\neg p}^+ card(C^{-\pm}) + \mu_q^- card(C^{\pm+}) & o \notin P^+(t) \wedge o \notin Q^-(t) \\ 0 & otherwise \end{cases}$$

$$\phi^{--} = \begin{cases} card(C^{--}) + \mu_{\neg p}^- card(C^{-\pm}) + \mu_{\neg q}^- card(C^{\pm-}) & o \notin P^+(t) \wedge o \notin Q^+(t) \\ 0 & otherwise \end{cases}$$

Value $\mu_{\phi}^{x} \in [0,1]$ is the influence from the ϕ-defined observation on the possible positive (+) or negative (-) realisation $x \in \{+,-\}$ of the unobserved property. The influence μ_{ϕ}^{x} of a particular pq-partial observation relates to the observed tendency in the collected empirical knowledge. In particular, utilising the past observations the agent is able to estimate the conditional probabilities of a particular realisation of the ϕ-undefined observation. Consequently, it uses equations given by Definition 6.

Definition 6. *Conditional probability estimators are defined as:*

$$\mu_{p}^{x} = \frac{card(C^{+x})}{card(C^{++} \cup C^{+-})} \quad \mu_{\neg p}^{x} = \frac{card(C^{-x})}{card(C^{-+} \cup C^{--})}$$
$$\mu_{q}^{x} = \frac{card(C^{x+})}{card(C^{++} \cup C^{-+})} \quad \mu_{\neg q}^{x} = \frac{card(C^{x-})}{card(C^{+-} \cup C^{--})}$$

in case the delimiters from the aforementioned fractions are 0 then absolute probability estimators are calculated:

$$\mu_{p}^{x} = \mu_{\neg p}^{x} = \frac{card(C^{+x} \cup C^{-x} \cup C^{\pm x})}{card(C^{++} \cup C^{-+} \cup C^{\pm +} \cup C^{+-} \cup C^{--} \cup C^{\pm -})}$$
$$\mu_{q}^{x} = \mu_{\neg q}^{x} = \frac{card(C^{x+} \cup C^{x-} \cup C^{x\pm})}{card(C^{++} \cup C^{+-} \cup C^{+\pm} \cup C^{-+} \cup C^{--} \cup C^{-\pm})}$$

Having defined the cardinalities of the grounding sets ϕ^{xy} $(x, y \in \{+,-\})$, we can now define the relative grounding strengths that compensates for the pq-partial observations. Modified relative grounding strength of a mental model 'p^x and q^y' is:

$$\lambda^{xy} = \frac{\phi^{xy}}{\phi^{++} + \phi^{+-} + \phi^{-+} + \phi^{--}}, \quad \text{where } x, y \in \{+,-\} \tag{7}$$

Finally, we can extend the definition of the epistemic relation (See def. 1,2,3) in order to utilise the information gathered in the past pq-partial observations and possible pq-partial current situation:

Definition 7. *Epistemic satisfaction relation* $ST(t) \models^{E} Know(p \wedge q)$ *holds iff*

$$o \in P^{x}(t) \wedge o \in Q^{y}(t) \wedge \lambda^{++} = 1, \quad \textit{where } x, y \in \{+, \pm\}.^{4}$$

4.2 Relation to the Classical Definition

Presented extension introduces a straightforward modification to the definition of relative grounding strengths compensating for the existence of partial observations. Moreover, the extended model allows to maintain the underlying mechanisms of the classical grounding. In particular, the original model is a case of the extended model, i.e., without partial observations(See Theorem 1).

Theorem 1. *When sets* C^{xy}, $x = \pm \vee y = \pm$, *are empty, modified relative grounding strength (eq. 7) is equivalent to relative grounding strength (eq. 6).*

[4] Similairly for $(p \wedge \neg q) : x \in \{+, \pm\}, y \in \{-, \pm\}, \lambda^{+-} = 1$, for $(\neg p \wedge q) : x \in \{-, \pm\}, y \in \{+, \pm\}, \lambda^{-+} = 1$, and for $(\neg p \wedge \neg q) : x, y \in \{-, \pm\}, \lambda^{--} = 1$

Proof. As sets C^{xy}, $x = \pm \vee y = \pm$ are empty, then the introduced modifications $\phi^{xy} = card(C^{xy})$ (See def. 6) and λ^{xy} equal in both equations. □

Additionally, the proposed extension satisfies the postulate P1 and allows the agent to ground only the adequate modal levels, i.e., maintaining the common sense interpretations of grounded statements, despite the existence of partial observations (See Theorem 2).

Theorem 2. *When the set $C^{-\pm}$ or $C^{\pm-}$ is not empty, then $\lambda^{++} < 1$ and epistemic relation for $Know(p \wedge q)$ is not met.*[5]

Proof. As one of the sets $C^{-\pm}$ or $C^{\pm-}$ is not empty, then the introduced modification on the cardinalities of grounding sets C^{-+}, C^{--} or C^{+-} is non zero. Consequently, the relative grounding strength of the mental model for $p \wedge q$ is lower $\lambda^{++} < 1$ then the required threshold for the modal level of knowledge ($Know$). □

5 Computational Examples

To illustrate the proposed modifications we present 3 examples (table 3), each representing a set of 11 observations (similarly to the example in sec.3).

Table 3. Three exemplary experiences gathered from 11 time moments

	t	1	2	3	4	5	6	7	8	9	10	11
Example 1	p	+		+	+	-		+		+	-	
	q		+	+	+		+	+		+		
Example 2	p	+		+		+	+	+	+			+
	q	+	+	+		+		+		+	+	
Example 3	p	+		+	+	-	-	+	-		-	
	q	+	-	+	-					+		

Within Example 1, time moments $3, 4, 7$ and 9 contain observations where both properties were known. Only these moments shall be included in grounding sets (eq. 5). Within all of these moments the object possessed both p and q, hence λ^{++} (eq. 6) is equal to 1 and statement $Know(p(o) \wedge q(o))$ is allowed. On the other hand, we have two partial observations for moments 5 and 10 neglecting $p(o) \wedge q(o)$. These observations are included within the newly defined grounding strengths (def. 7), consequently the knowledge statement is not allowed.

Example 2 presents a situation where old definition (def. 1) does not support $Know(p(o) \wedge q(o))$ while new definition (def. 7) does. Agent observed the property p at moment 11 and had not observed the property q. According to the new definition this does not deny grounding $Know(p(o) \wedge q(o))$ statement.

[5] Similarly for all the other possible statements of knowledge, i.e., $Know(p \wedge \neg q)$, $Know(\neg p \wedge q)$, and $Know(\neg p \wedge \neg q)$.

Example 3 presents a situation where both properties are rarely known (at moments $1, 3$ and 4). When they were observed together, it often happened that the object exhibited p and q. On the other hand it often happened, that only one property was known. In such situations usually the object did not exhibit property p. Old definition of grounding sets results in $\lambda^{++} = 0.66$, that allows $Bel(p \wedge q)$ statement. New definition includes partial observations and changes agent's certainty significantly, giving $\lambda^{++} = 0.36$ and allowing only weaker $Pos(p \wedge q)$ statement.[6]

6 Conclusions

We proposed slight modifications to the grounding theory [5,6,7]. Modifications were inspired by the need to include incomplete observations in the process of constructing agent's cognitive stance. Our changes extend the cognitive abilities of an agent to incorporate partial observation in the grounding process. We extended the existing model of choosing cognitive material taking part in grounding of modal conjunctions and propose a set of renewed definitions that consider incomplete observations. Proposed changes comply with the introduced pragmatic and common-sense requirements. In particular, in the extended model the incomplete observations that contradict the truth of $Know(p(o) \wedge q(o))$ are included in the grounding sets and disallow the usage of such statements. The introduced modifications also influence agent's cognitive attitude in the case of modal operators of possibility and belief.

Acknowledgements. This paper was partially supported by Polish Ministry of Science and Higher Education under grant no. N N519 407437.

References

1. Dennett, D.C.: True believers: The intentional strategy and why it works. In: Stich, Warfield (eds.) Mental Representation: A Reader. Blackwell (1994)
2. Dennett, D.C.: Kinds of Minds. Basic Books, New York (1996)
3. Harnad, S.: The Symbol Grounding Problem. Physica D 42, 335–346 (1990)
4. Hintikka, J.: Knowledge and belief. An introduction to the logic of the two notions. Cornell University Press (1962)
5. Katarzyniak, R.: Grounding Atom Formulas and Simple Modalities in Communicative Agents. In: Applied Informatics 2003, pp. 388–392 (2003)
6. Katarzyniak, R.: The Language Grounding Problem and its Relation to the Internal Structure of Cognitive Agents. J. UCS 11(2), 357–374 (2005)
7. Katarzyniak, R.: On some properties of grounding uniform sets of modal conjunctions. Journal of Intelligent and Fuzzy Systems 17(3), 209–218 (2006)
8. Kripke, S.: Semantical Analysis of Modal Logic. Zeitschrift fur Mathematische Logic und Grundlagen der Mathematik 9, 67–96 (1963)
9. Richards, I.A., Ogden, C.A.: The Meaning of Meaning. Harvest/ HBJ (1989)

[6] $\lambda_{minPos} = 0, \lambda_{minBel} = 0.65, \lambda_{maxBel} = 1$ were assumed.

G-SteX: Greedy Stem Extension for Free-Length Constrained Motif Discovery

Yasser Mohammad[1], Yoshimasa Ohmoto[2], and Toyoaki Nishida[2]

[1] Assiut University, Egypt
yasserm@aun.edu.eg
[2] Kyoto University, Japan
nishida@i.kyoto-u.ac.jp

Abstract. Most available motif discovery algorithms in real-valued time series find approximately recurring patterns of a known length without any prior information about their locations or shapes. In this paper, a new motif discovery algorithm is proposed that has the advantage of requiring no upper limit on the motif length. The proposed algorithm can discover multiple motifs of multiple lengths at once, and can achieve a better accuracy-speed balance compared with a recently proposed motif discovery algorithm. We then briefly report two successful applications of the proposed algorithm to gesture discovery and robot motion pattern discovery.

1 Introduction

Motif discovery (MD) can be informally defined as efficiently finding unknown approximately recurring patterns in long time series with no prior knowledge about their locations. Constrained Motif Discovery was introduced in [13]. In CMD, the goal is to find unknown approximately recurrent patterns but utilizing a set of constraints on motif locations. One way to introduce these constraints when no prior knowledge is available is to use a change-point discovery (CPD) algorithm and then search for the motifs around the discovered change points. The rationale for this approach is that a motif occurrence is – by definition – expected to be different than the neighboring subsequences in the time series and hence change scores are expected to be high around the boundaries of motif occurrences. By utilizing this insight, motif discovery can be achieved in strictly linear worst-case time [13]. In this sense a CMD algorithm can be used in conjunction with a CPD algorithm to solve the general motif discovery problem. CMD has applications in activity detection [11], gesture recognition [16], Human-Robot Interaction [12], among other fields.

The main contribution of this paper is the introduction of a novel motif discovery algorithm that can solve the CMD problem without requiring an upper bound on motif lengths. The paper also introduces a quantitative method to compare different MD and CMD algorithms and shows that the proposed algorithm can provide a better speed-accuracy balance compared with a recently proposed CMD algorithm.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 417–426, 2012.

In this paper we define a motif as a pattern that occurs in the time series at least twice. Motif occurrences need not be exact replicas because of the probable existence of noise or outliers but the distance between these occurrences (using some predefined distance function) must be significantly higher than the distances between them and other subsequences of the time series of the same length. For a more formal definition of a motif, please refer to [13].

The rest of this paper is organized as follows: Section 2 gives the problem statement. Section 3 situates the proposed method within existing research. Section 4 introduces the proposed algorithm and section 6 evaluates it using the technique proposed in section 5. The paper is then concluded.

2 Problem Statement

The Free-Length Constrained Motif Discovery problem can be stated as follows: *Given a single dimension real-valued time series $x(t)$ of length T and an optional time series $c(t)$ of length T representing the relative probability of finding a motif at every time-step, and a distance function d, find all recurrent patterns of any length $l_{min} < l < T/2$.*

This statement requires that the algorithm should be able to utilize domain knowledge of probable motif locations if available but it should be able to discover motifs even if this knowledge is not available. In the introduction, we highlighted one way to achieve that using a change-point discovery algorithms and we use this approach in this paper. The $T/2$ limit is implicit in motif definition as a motif must have at least two disjoined occurrences. The limit l_{min} is not restrictive as it can be set to a very small number (e.g. 2) and its only purpose is to limit very short (e.g. single point) motifs that are not expected to correspond to any real patterns in the data.

The problem statement explicitly states that the input is single-dimensional but this can be easily alleviated by projecting the input to a 1-D subspace (e.g. using Principal Component Analysis), as proposed in [15] or by applying it to every dimension and then combining the results. Given that the proposed solution is linear, for time-series in which the number of dimensions is much less than the length of the time series, this later approach will be efficient because the final across-dimensions motif combination process will be carried out on a comparatively small number of motifs and will have more flexibility than the projection approach of [15].

3 Related Work

Unconstrained motif discovery is a known problem in data-mining and several algorithms have tried to solve it [2],[20],[4] [7], [9] [21]. With the exception of Gemoda [4] which is quadratic in time and space complexities, these algorithms aim to achieve sub-quadratic time complexity by first looking for candidate motif *stems* using some heuristic method and then doing exhaustive motif detection instead of motif discovery which is linear in time.

The most used method for finding these stems is based on the PROJECTIONS algorithm [2] which requires discretization of the data (usually using the SAX [5] algorithm). After discretization, a collision matrix is built that specifies a measure of similarity between every two subsequences in the time series. The size of this collision matrix is quadratic in the length of the time series. To reduce the time and space complexity, the parameters of SAX have to be selected to ensure that this matrix is sparse which leads to linear time and space complexity. Two major factors that determine the sparsity of the collision matrix are the word length (number of symbols per subsequence) and vocabulary size (number of different symbols) parameters of SAX. Unfortunately there is no way to find the optimal values of these parameters. Minnen et al. [10] suggested a simple heuristic to find acceptable values for these two parameters but this heuristic requires multiple estimations of the sparsity of the distance matrix which can be time consuming in itself.

To avoid calculating the accurate distance between every two pairs of subsequences (as done in Gemoda [4]), random projections [2] is used to approximate this distance. The subsequences with high similarity are then used as the stems for subsequent motif detection. The proposed system differs from these algorithms on requiring no discretization, and no motif-length or upper limit parameters. The proposed algorithm also avoids building the collision matrix. The collision matrix is built using a predefined motif length which makes these algorithms inadequate for free-length motif discovery without a post-processing concatenation step.

PERUSE [20] uses a different approach by casting the discovery problem in probabilistic terms and solving it directly on the real-valued data. This approach has the advantages of requiring no discretization and discovering motifs of variable sizes. It also uses a stem growing technique to find free-length motifs. One limitation of this algorithm is that it assumes dense motif occurrences in the time series. Another limitation is that the algorithm needs to compare each window of the time series with the rest of the windows which can degrade performance considerably. The proposed approach was designed to overcome both of these limitations.

VLMD [19] was recently proposed to find variable-length motifs in time-series. The algorithm uses exhaustive search of all possible motif lengths and applies an exact motif discovery algorithm (as defined in [17]) at each length. Even though the authors have shown that the algorithm can find a few number of *interesting* motif lengths, the algorithm is not expected to scale well for longer time series due to its exhaustive search strategy. Another problem of the algorithm is that it defines a motif of any length (after [17]) to be the two subsequences with minimum distance. This means that the algorithm cannot find multiple motifs of the same length. Again, the proposed algorithm overcomes all of these limitations.

Li and Lin [6] recently proposed using the Sequitur [18] algorithm for discovering motifs of variable length using grammar inference after discretization using SAX [8]. This technique can discover variable length motifs but it requires discretization and cannot utilize constraints. Another problem with this approach

is that a small burst of outliers in a single motif occurrence will result in dividing this motif into two disjoint motifs.

To increase the speed of motif discovery and overcome the need of comparing all subsequences in the time series, Catalano et al. [1] proposed an algorithm that utilizes random sampling from the time series. This algorithm was modified in [13] to utilize constraints leading to the MCFull algorithm. MCFull was shown to provide higher accuracy than the original Catalano's algorithm. One problem that both these algorithms suffer from is the need to specify an upper bound on motif lengths to be discovered.

Table 1 compares some of the aforementioned algorithms with the proposed algorithm.

Table 1. Comparison Between Different Motif Discovery Algorithms

Algorithm	Gemoda[4]	PROJECTIONS[2]	PERUSE[20]	Sequitur[6]	Catalano[1]	MCFull[13]	DGR [12]	VLMD [19]	**Proposed**
No Discretization			√		√	√	√		√
Linear Time		†		√	√	√	√		√
Linear Space		†	√	√	√	√	√	√	√
Variable-Length			√	√	√	√	√	√	√
Free-Length				√				√	√
Domain Knowledge						√	√		√
† Linear if the collision matrix is sparse, otherwise quadratic									

4 Proposed Algorithm

The proposed algorithm consists of two major stages: stem generation and finalization. During stem generation, the algorithm generates a set of candidate motifs with at least two occurrence locations for each. During finalization, the algorithm remove duplicates, stitches together overlapping motifs and removes short noise bursts around discovered motifs. The algorithm used in the first and most crucial stage is called Greedy STem eXtension (G-SteX).

If no constraint is given, the Robust Singular Spectrum Transform (RSST) [14] is used to discover change points in the time series and generate the constraint signal $c\,(t)$.

A set of candidate motif occurrence locations (*candLoc*) is generated by localizing local maxima in $c\,(t)$. Another possible approach is to sample from the probability distribution generated by normalizing $c\,(t)$ for a predefined number of times.

A set of stem locations is generated from *candLoc* by collecting all subsequences of length l_{min} around each member of *candLoc*. This set is called *sequences*. Notice that l_{min} is a small integer that is used for initialization and the stems will then be allowed to grow. In our experiments, we selected l_{min} to be $\max(10, 0.00001T)$, where T is the length of the input time series. The results presented were not dependant on the choice of this parameter as long as it was less than half the discovered motif lengths.

The distances between all members of *sequences* is then calculated and clustered into four clusters using K-Means. The largest distance of the cluster containing shortest distances is used as an upper limit of distances between *similar* subsequences ($maxNear$) and is used to prune future distance calculations.

The sequences generating the first distance clusters ($nearSequences$) are then used for finding motif stems. Each one of these distances corresponds to a pair of sequences in the *sequences* set. This is the core step in G-SteX. For each pair of sequences in the *nearSequences* set, the first of them is slid until minimum distance is reached between the pair and then a stem is generated from the pair using one of the two following two techniques:

The first extension technique is called G-SteXB (for binary/bisection) and it starts by trying to extend the motif to the nearest end of the time series and if the stopping criteria is not met, this is accepted as the motif stem. If the stopping criteria is met, the extension is tried with half of this distance (hence the name binary/bisection) until the stopping criteria is not met. At this point, the extension continues by sequentially adding half the last extension length until the stopping criteria is met again. This is done in both directions of the original sequences. By the end of G-SteXB, we have a pair of motif occurrences that cannot be extended from any direction without meeting the stopping criteria.

The second extension technique is called G-SteXS (for sequential) and it extends the sequences from both sizes by incrementally adding l_{min} points to the current pair from one direction then the other until the stopping criteria is met. At this step it is possible to implement the don't care section ideas presented in [2] by allowing the extension to continue for subsequences shorter than the predefined maximum outlier region length (don't care length) as long as at least one l_{min} points are then added without breaking the stopping criteria.

The stopping criteria used in G-SteX with both its variations combines pruning using the $maxNear$ limit discovered in the clustering step with statistical testing of the effect of adding new points to the sequence pair. If the new distance after adding the proposed extension is larger than $maxNear$, then the extension is rejected. If the extension passes this first test, point-wise distance between all corresponding points in the motif stem and between all corresponding points in the proposed extension part. If the mean of the point-wise distances of the proposed extension is less than the mean of the point-wise distances of the original stem or the increase in the mean is not statistically significant according to a t-test then the extension is accepted otherwise it is rejected.

Once G-SteXS or G-SteXB is finished, we have a set of – possibly overlapping – candidate motifs with at least two occurrences of each. The finalization step of

G-SteX involves the following four operations running until no further change is possible on the locations of the discovered motifs or a predefined maximum number of iterations is reached: Firstly, motif candidates with similar means (mean distance is less than $maxNear$) are combined into single motif candidates. Secondly, overlaps are resolved by removing partial motifs (that are totally covered by other longer ones) and cutting overlapping sections from motifs of similar lengths. Occurrences that appear in multiple motifs are then removed and a local search of optimal occurrence shifts to minimize overall motif distances is employed. Finally, extra noise regions that may have entered the candidates around the correct motifs during the extension are removed by carrying a single G-SteXS extension from the middles of motif occurrences.

The final output of the algorithm is the set of boundaries of discovered occurrences of each motif. An optional full motif detection can then be run to find all occurrences of each motif.

The full implementation of this algorithm in MATLAB is available as part of the CPMD Toolbox from the authors' web site [3]. Several test time series and evaluation routines can also be downloaded as a part of the same package. This implementation of G-SteX has the advantage of being able to utilize an optional range of motif lengths when available during the discovery process. This feature improves the speed of the algorithm when this information is available.

5 Comparing Motif Discovery Algorithms

Comparing motif discovery algorithms is not a trivial task. For example, a MD algorithm that finds a hundred occurrences of a 5-points motif with a single point shift will have 200 errors if compared directly with the ground truth. Another algorithm that finds exactly only 61 occurrences and fails to find the rest may be considered better if only false positives and negatives are counted. This may or may not be appropriate depending on the application. The inadequacy of false positive and false negative based evaluation criteria (including precision, recall and F-measure) calls for a more appropriate evaluation criteria when these measures are not adequate.

In this paper, we propose to use a mutli-dimensional criterion for evaluating MD algorithms assuming that ground-truth information about the motifs and their occurrences is available.

For each of the discovered motifs, four quantities are calculated:

- Correct Motifs: The number of discovered motifs that completely cover at least some occurrences of a single ground truth motif.
- Covering None: The number of discovered motifs that cover no parts of any ground truth motif.
- Covering Partially: The number of discovered motifs that cover only parts of some occurrences of a single ground truth motif.
- Covering Multiple: The number of discovered motifs that cover occurrences from multiple ground truth motifs.

Based on the application, one or more of these dimensions may be more important than the others. For example, in gesture discovery it is necessary to discover the complete gesture and hence motifs in the *covering-multiple* set should be treated harshly but in an imitation learning task that involves a higher-cognitive planner, these motifs may be counted as correct motifs because the boundaries between acts are not important in this task.

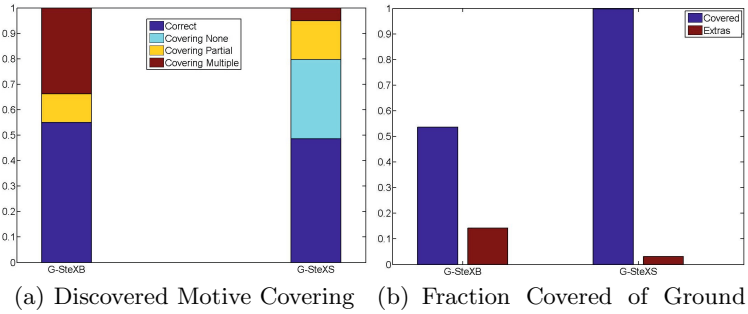
These four criteria are calculated from the view-point of the discovered motifs. We also calculate two criteria from the ground-truth motifs view point: *Fraction Covered* is the fraction of occurrences of each one of the ground-truth motifs that is covered by discovered motifs. *Extra Fraction* is the fraction of the discovered motif occurrences used in calculating the fraction covered that are not covering a part of the ground truth occurrence.

6 Evaluation

Our first evaluation experiment considered comparing G-SteXB and G-SteXS using synthetic data. We generated 500 timeseries with varying lengths (between 20000 and 50000 points each), varying noise levels (from zero to 1 P-P variance normal noise), varying outlier percentage (from 0 to 1%) and varying scoring localization errors (from 0 to 20). Within each timeseries three different motifs were embedded of lengths between 50 and 100 points. Only four occurrences of each motif were implanted.

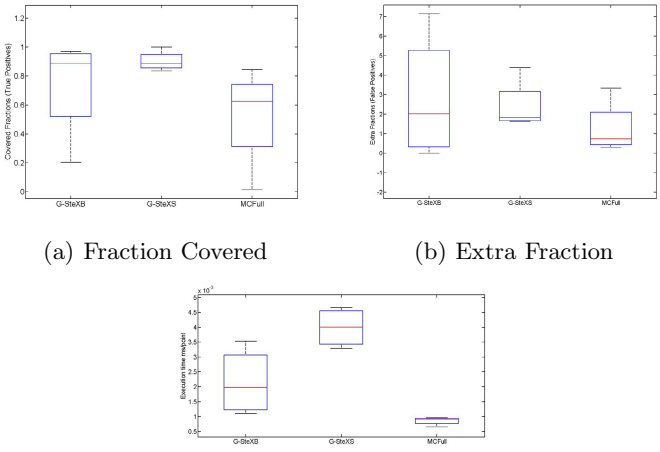
In the first experiment, both G-SteXB and G-SteXS were used to discover motifs in the first 100 timeseries of this set and the six evaluation criteria discussed in the previous section were calculated. Fig. 1 shows the average of these results over the complete set of timeseries tested. If we limit ourselves to the number of correct motifs (as defined in the previous section), G-SteXB is slightly better than G-SteXS (that this difference is not statistically significant).

Because G-SteXS is more conservative, it never generates motifs that correspond to just noise in the input but G-SteXS can sometimes generate such motifs. These results suggest that G-SteXB has lower false positives in terms of



(a) Discovered Motive Covering (b) Fraction Covered of Ground Truth Motifs

Fig. 1. Comparison between G-SteXB and G-SteXS on synthetic data



(a) Fraction Covered (b) Extra Fraction

(c) Execution Time

Fig. 2. Comparison between G-SteXB, G-SteXS and MCFull on synthetic data

complete motifs. Nevertheless, if we consider the results from the ground-truth motifs viewpoint as shown in Fig. 1-B, G-SteXS has both higher true positives and lower false positives in terms of specific motif occurrences. Depending on the application, one or the other of these two extension techniques may be more adequate. In most cases, G-SteXS will find more useful motifs and will tend to find complete motifs while G-SteXB will be less sensitive to noise (generating less non-sense motifs).

In the second experiment, both G-SteXB and G-SteXS are compared to MCFull which is shown in [13] to be competitive with existing state-of-the-art fixed length motif discovery algorithms. MCFull was selected for comparison because it is implemented using the same MATLAB environment and it is one of the fastest CMD algorithms available [13]. For MCFull, we had to provide the correct motif length upper bound. Fig. 2 shows some of the results obtained from this experiment (more results are available in the CPMD Toolbox [3]). MCFull was the fastest algorithm as expected because it utilizes a pre-specified upper motif length bound that is not available for G-SteX. G-SteXB and G-SteXS gave significantly higher motif coverage for a slight increase in the false positives added compared with MCFull (the differences are again statistically significant).

These results suggest that G-SteX algorithms can recover implanted motifs with an even better false positive/false negative balance than MCFull without needing a pre-specified upper limit on the motif length. The average execution time of G-SteXB and G-SteXS is 2 and 4 milliseconds per point on an intel Core 2 Due T9600 machine with 4GB memory running windows 7 64-bits OS. From Fig. 2, it is clear that in both evaluation criteria, G-SteXS had lower variance compared with G-SteXB which can be a useful feature for some applications.

To evaluate G-SteX on a real-world application, we utilized a dataset containing seven patterns representing seven different hand-gestures. Each gesture was executed ten times during a 20 minutes free hand motion session. Two accelerometers were attached to the back of the hand and the tip of the middle finger while the subject was executing the gestures. The data was sampled 100 times/second leading to a 120000 points 3D time-series. The time-series was converted into a single space time series using PCA as proposed in [15]. G-SteXS was applied to this time-series and it recovered 16 motifs. The longest six of them corresponded to six of the seven gestures embedded in the data (with a discovery rate of 85.7%) and five of them corresponded to partial and multiple coverings of these gestures.

As another application, motion patterns were collected from a differential drive robot simulator moving around an empty arena and executing one of three different motions at random times (a circle, a triangle and a square). Ten sessions with four occurrences of each pattern within each session were collected and G-SteXB was applied to each session after projecting the 2D time-series into a 1-D time-series as in the previous case. The algorithm discovered 6 motifs. The longest three motifs corresponded to the three motion patterns in the data with an average covering of 93.5%, two patterns corresponded to partial coverings of the circle motif with 0.07% extra noise added to it. The final discovered pattern did not correspond to any specific motion pattern but represented periods in which the robot was moving in a straight line during its random walk.

7 Conclusion

This paper presented a novel constrained motif discovery algorithm called G-SteX with two variations that differ in the way motif stems are extended to generate the final motifs. The proposed algorithm has the advantage of requiring no pre-specified motif length range for its operation. Comparison between the G-SteX (in its two variations) and the MCFull algorithm on synthetic data with known ground-truth information shows that G-SteX can achieve higher motif discovery (true positives) for only a slight increase in the false positives even though MCFull requires an upper-limit on the motif length. Applications of the proposed algorithm to gesture discovery and robot motion pattern discovery were also presented and G-SteX was able to discover 85.7% of the gestures and 93.5% of the motion patterns present in its input.

References

1. Catalano, J., Armstrong, T., Oates, T.: Discovering Patterns in Real-Valued Time Series. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) PKDD 2006. LNCS (LNAI), vol. 4213, pp. 462–469. Springer, Heidelberg (2006)
2. Chiu, B., Keogh, E., Lonardi, S.: Probabilistic discovery of time series motifs. In: KDD 2003, pp. 493–498 (2003)
3. CPMD Toolbox, `http://www.ii.ist.i.kyoto-u.ac.jp/~yasser/cpmd/cpmd.html`

4. Jensen, K.L., Styczynxki, M.P., Rigoutsos, I., Stephanopoulos, G.N.: A generic motif discovery algorithm for sequenctial data. BioInformatics 22(1), 21–28 (2006)
5. Keogh, E., Lin, J., Fu, A.: Hot sax: efficiently finding the most unusual time series subsequence. In: ICDM, p. 8 (November 2005)
6. Li, Y., Lin, J.: Approximate variable-length time series motif discovery using grammar inference. In: Proceedings of the Tenth International Workshop on Multimedia Data Mining, p. 10. ACM (2010)
7. Lin, J., Keogh, E., Lonardi, S., Patel, P.: Finding motifs in time series. In: The 2nd Workshop on Temporal Data Mining, at the 8th ACM SIGKDD International, pp. 53–68 (2002)
8. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing SAX: a novel symbolic representation of time series. Data Mining and Knowledge Discovery 15(2), 107–144 (2007)
9. Minnen, D., Essa, I., Isbell, C.L., Starner, T.: Detecting Subdimensional Motifs: An Efficient Algorithm for Generalized Multivariate Pattern Discovery. In: ICDM (2007)
10. Minnen, D., Starner, T., Essa, I., Isbell, C.: Improving activity discovery with automatic neighborhood estimation. In: Int. Joint Conf. on Artificial Intelligence (2007)
11. Miwa, H., Itoh, K., Matsumoto, M., Zecca, M., Takanobu, H., Rocella, S., Carrozza, M., Dario, P., Takanishi, A.: Effective emotional expressions with expression humanoid robot we-4rii: integration of humanoid robot hand rch-1. In: IROS 2004, vol. 3, pp. 2203–2208 (2004)
12. Mohammad, Y., Nishida, T.: Learning interaction protocols using augmented baysian networks applied to guided navigation. In: IROS 2010, pp. 4119–4126 (2010)
13. Mohammad, Y., Nishida, T.: Constrained motif discovery in time series. New Generation Computing 27(4), 319–346 (2009)
14. Mohammad, Y., Nishida, T.: Robust Singular Spectrum Transform. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS, vol. 5579, pp. 123–132. Springer, Heidelberg (2009)
15. Mohammad, Y., Nishida, T.: On comparing SSA-based change point discovery algorithms. In: 2011 IEEE/SICE International Symposium on System Integration (2011) (to appear)
16. Mohammad, Y., Nishida, T., Okada, S.: Unsupervised simultaneous learning of gestures, actions and their associations for human-robot interaction. In: Proceedings of the 2009 IEEE IROS, pp. 2537–2544 (2009)
17. Mueen, A., Keogh, E., Zhu, Q., Cash, S.: Exact discovery of time series motifs. In: Proc. of 2009 SIAM (2009)
18. Nevill-Manning, C.G., Witten, I.H.: Identifying Hierarchical Structure in Sequences: A linear-time algorithm. Journal of Artificial Intelligence Research 7(1), 67–82 (1997)
19. Nunthanid, P., Niennattrakul, V., Ratanamahatana, C.: Discovery of variable length time series motif. In: ECTI-CON 2011, pp. 472–475 (2011)
20. Oates, T.: PERUSE: An unsupervised algorithm for finding recurring patterns in time series. In: ICDM, pp. 330–337 (2002)
21. Tang, H., Liao, S.S.: Discovering original motifs with different lengths from time series. Know.-Based Syst. 21(7), 666–671 (2008)

Event-Level Textual Emotion Sensing Based on Common Action Distributions between Event Participants

Cheng-Yu Lu[1,2], William W.Y. Hsu[1,3], and Jan-Ming Ho[1]

[1] Institute of Information Science, Academia Sinica, Taiwan
[2] PIXNET
http://www.pixnet.net
[3] Dep. of CSIE, National Taiwan Normal University, Taiwan

Abstract. Automatic emotion sensing in textual data is crucial for the development of intelligent interfaces in interactive computer applications. This paper reports a high-precision, domain-independent approach for automatic emotion sensing for "events" embedded in sentences. The proposed approach is based on the common action distribution between the subject and object of an event. We have incorporated semantic labeling and web-based text mining techniques, together with a number of reference entity pairs and hand-crafted emotion generation rules to realize an event emotion detection system. Moreover, a hybrid emotion detection engine is presented by incorporating a set of predefined emotion keywords and the proposed event-level emotion detection engine. The evaluation outcome reveals a rather satisfactory result with about 73% accuracy for detecting the Happy, Sad, Fear, Angry, Surprise, Disgust, and Neutral.

Keywords: emotion sensing, web mining, natural language processing, event.

1 Emotion Sensing from Textual Data

In the past, many studies have been conducted to detect a user's affective states automatically from textual data. Some using "keyword-spotting" techniques [2], but the results are not satisfactory. Keyword-spotting approach apparently can't apply to sentences without clearly-defined affective keywords. A number of studies applied emotion theories to determine emotions of interactive agents in interactive systems [3,4].

Liu et al. [4] reported a rather interesting approach to detect sentence-level emotion based on a large-scale common sense knowledgebase, ConceptNet. The approach uses real world knowledge about the inherent affective nature of everyday situations (such as "getting into a car accident") to classify sentences into basic emotion categories.

Wu et al. [1] proposed a novel approach for sentence-level emotion detection based on the semantic labels (SLs) and attributes (ATTs) of entities of a

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 427–436, 2012.

sentence. To distinguish the emotions of happy and unhappy, the SLs are manually classified into three categories, Active SLs (e.g., obtain, reach, lost, hinder), Negative SLs (e.g., no, never), and Transitive SLs (e.g., finally, but, fortunately). ATTs of an entity are obtained automatically from a lexical resource, WordNet [4]. The results show the degree of accuracy is rather high. The proposed approach exploited modern Natural Language Processing (NLP) technologies and is one of the rare studies that dealt with sentence-level emotion detections with high precision.

However, in our opinion, the major issues that might limit the performance and applicability of this approach in wider contexts include (1) the need of affective-annotated sentences as training sample, and (2) the use of attributes (ATTs) as the sentence-level emotion-invoking ingredients.

As the first step for our endeavor towards a robust emotion sensing engine from free-texts using web mining approaches [7], this study proposes a novel approach for detecting emotion of an individual "event" embedded in English sentences such as "a student fails his exams", "a girl saw a diamond", "a cat was chased by a dog", "a mouse encounters a cat", etc. We adopt the "common actions" between the event participants as the major cue to determine the event-level emotions. In the following sections, the underlying principles of our approach will be elaborated.

2 Emotion Sensing Based on Common Action Distributions between Event Participants

In this section, we will elaborate on the underlying principles and detail processes of our methodology for automatically sensing emotions of events embedded in a textual sentence. We first present here a typical scenario to illustrate the steps to achieve our goal. In any English sentence, the verb of a sentence typically indicates a particular "action" performed by one event participant to the other participant. For example, in the sentence, "The girl saw a viper snake", the action is a "sighting action", with two entities, "the girl" as the subject and "a viper snake" as the object. While we often intuitively assume that the girl would often be terrified while meeting a viper snake, but how can a computer understand this?

Let us investigate why a girl would usually be terrified when she sees a viper snake. One way to reason is that the snake "usually" performs certain undesirable actions (e.g., bite, paralyze, attack, or kill) but "never" performs desirable actions (e.g., love, feed, or supply) on a girl. Contrarily, a kid would usually be happy when she sees his/her mother since she "usually" performs desirable actions but "rarely" performs undesirable actions on a baby. The real-life probability distribution of actions perform by a subject to an object is termed as the "Common Action Distribution (CAD)" in this paper. In practice, knowing the CAD between the subject and the object in a specific event would allow a computer to reasonably guess the emotion invoked. For example, if the CAD between a vampire and a girl is close to that between a cat and a mouse, the

emotion of a girl when she saw a vampire should most likely similar to that of a mouse when it saw a cat. As compared to the relevant work by Wu et al. [1], our approach adopts the common actions between two event participants instead of using their attributes as the underlying features to determine the emotions.

Based on the above analysis, it appears the success of our approach for emotion detection would hinge on the ability of a computer to automatically retrieve the CADs between two entities that participate in an event. While people may intuitively believe that CADs between entities are difficult or impossible to obtain without extensive human annotations, we resort to web-based text mining and semantic role labeling techniques [2,3,8] that fortunately provide a convenient solution to achieve our goal. First of all we introduce the overall flow diagram for our proposed emotion detection engine, as shown in Figure 1. First, we select a number of entity pairs (e.g., cat and dog, girl and spider, etc.) as the Reference Entity Pairs (RE-Pairs). For each RE-Pair, the CAD is obtained using web-based text mining techniques. For each RE-Pairs, possible emotion-invoking events that often occur between the two entities are enumerated and manually assigned with emotions to form a set of "Emotion Generation Rules". In the emotion sensing stage, the subject and object of a target event, termed as the Target Entity Pair (T-Pair), in a sentence is first recognized automatically using semantic role labeling techniques. The CAD of the T-Pair is obtained using web-based text mining techniques. The emotion of the target event is assigned based on the "Emotion Generation Rules" (EGRs) of the RE-Pairs with best match on the CAD to that of the T-Pair. In the following, the underlying principles and detail operations of the proposed framework will be elaborated.

2.1 Web-Based Text Mining for Common Action Distributions (CADs) between Entities

We apply web-based text mining techniques to retrieve common actions between two given entities. In the past, studies based on variations of text mining approaches have been investigated to retrieve different types of knowledge (e.g., [9]). Given the vast amount of textual data available on Web, we believe that the actions of two given entities described in web pages shall give a roughly normalized distribution of their common interactions. In the text mining approach, given a target event with two particular entities (the target subject and target object), the first step of finding the common actions between them is to accumulate raw sentences possibly describing activities participated by them. This is done by formulating a set of effective "lexico-syntactic" patterns which unambiguously describe the activities. These descriptions can be written either in "passive" or "active" style, depending on whether the target subject is presented in an active or passive role in the sentence. In the passive-style, a sentence describes an action which the target object acts upon the target subject. In the active-style, the action is reversed. For example, considering a case where we intend to detect the emotion of a target event, "a girl saw a diamond". In this case, the target subject is "a girl" and the target object is "a diamond". Sentences with an active-style includes "a girl bought a diamond", "a girl lost a diamond".

On the contrary, examples of sentences with a passive-style includes “a diamond cut a girl”, “a girl was lured by a diamond”. Common actions accumulated in the active style and passive style are processed separately.

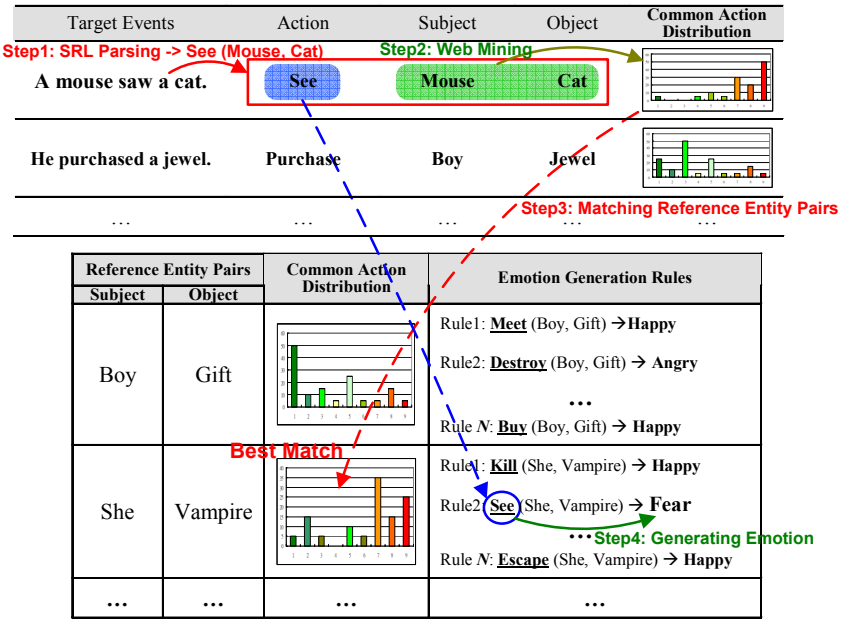


Fig. 1. Illustrative diagram for the proposed approach for sentence-level emotion sensing

We demonstrate more examples to illustrate the process for obtaining the common actions between two particular entities. Considering an attempt to determine the emotion of a mouse in a target event, “a mouse saw a cat” where “a mouse” and “a cat” are the target subject and target object respectively. To retrieve passive-style sentences which describe the actions that a cat frequently does to a mouse, effective lexico-syntactic query strings would include “a mouse was * by a cat”, “the mouse is * by the cat”, “the mouse is * by a cat”, “the cat * a mouse”, “a cat * the mouse” and so on. With these query strings, we would be able to accumulate a huge number of raw sentences from Web search engines. Sample sentences shown in the snippets of Web search results include:

A mouse was chased by a cat.
The mouse was killed by the cat.

From these results, we know that the common actions that a cat would frequently act upon a mouse include chasing, eating, killing, harassing, and so on. Since it is commonly known that a mouse would be fear in a target event “a mouse

saw a cat", we can generalize this situation to obtain an emotion model like the following: Given "chase", "eat", "kill" and "harass" as the passive-style common actions between S (target subject) and O (target object), in an target event that S is meeting O, the emotion of S is "fear".

In many situations, particularly when the target object rarely serves as an agent that can actively perform an action, the number of the passive style sentences are not sufficient enough to provide a convincing CAD. In such a case, we would use active-style syntactic patterns to accumulate sufficient common actions that the subject frequently acts upon the object. For example, consider a target event, "a girl receives a diamond", where "a girl" and "a diamond" are the target subject and target object respectively. Since the web search results provide very few sentences using passive-style lexico-syntactic patterns as described above, we instead resort to queries based on active-style lexico-syntactic patterns including "girl * a diamond", "girl * diamond", "diamond was * by the girl". Sentences retrieved would include:
girl desires a diamond.
girl loves the diamond.

Based on these results, we know that the common actions that a girl would frequently act upon a diamond include love, desire, want, and so on. Since it is commonly believed that a girl would be happy in a target event "a girl receives a diamond", we can generalize this situation to obtain another event-level emotion model like the following: Given "love", "desire", and "want" as the active-style common actions between S (target subject) and O (target object), in a target event that S is receiving O, the emotion of S is "happy".

2.2 Accumulating CADs for Selected Reference Entity Pairs (RE-Pairs)

We select a number of reference entity pairs (RE-Pairs) with which possible emotion-invoking events that often occur between them are enumerated and manually assigned with emotions to form a set of "Emotion Generation Rules". The CAD between a RE-Pair is obtained by using the web-based text mining approach described above. It is required that the interactions between the two entities in a RE-Pair are widely described in web pages such that CAD would reliably represent the un-biased distribution common actions between them. There are two set of RE-Pairs given in our emotion recognition system, namely the "passive-style RE-Pair set" and the "active-style RE-Pair set". The former contains common actions of a Re-Pair retrieved from the passive-style sentences, and the latter contains those from the active-style sentences. The scope of the RE-Pair set needs to widely cover different emotion-invoking scenarios such that all the desired emotion categories can be sensed based on the matching of CADs of a target pair (T-Pair) and RE-Pair. For each RE-Pair, a number of "Emotion Generation Rules" (EGRs) are manually constructed. For example, Table 1 gives a list of example EGRs for a RE-Pair, "cat and dog". Rule 1 says that the emotion of a cat when it meets a dog is "Fear"; Rule 2 says that the emotion of a cat when it bites a dog is "Angry".

2.3 Recognition of Subject, Object and Verb in a Sentence Using Semantic Role Labeling Techniques

In a real-life application when the emotions for free-text sentences are to be detected automatically, the subject, verb, and object of an event can be obtained using semantic role labeling technique. A general overview of the state-of-the-art semantic role labeling techniques has been discussed fully in [2,3,8]. Simply speaking, in any sentence, a verb (predicate) dominates an event. The verb's syntactic arguments are usually associated with the participants of the event. A semantic role is the relationship of a syntactic argument with the verb. One commonly utilized scheme for specifying the semantic roles is PropBank annotation [10]. In PropBank annotations, the arguments of a verb are labeled sequentially from ARG0 to ARG5, where ARG0 is usually the subject of a transitive verb; ARG1, its direct object, etc. A variety of adjunctive arguments, such as ARGM-LOC for locatives, and ARGM-TMP for temporal information, are also tagged. As an illustrative example, the set of semantic roles for the sentence, "I saw a girl in the park this morning" based on the PropBank style markup, can be presented as:

[ARG0 I] [Target saw] [ARG1 a girl] [ARGM-LOC in the park] [ARGM-TMP this morning]

Table 1. Emotion Generation Rules (EGRs) based on a Reference Entity Pair, "cat" and "dog"

Rule	Target Action (Verb synonym group)	Emotion of subject (cat)
1	Meet (encounter, join, see, connect)	Fear
2	Bite (nip, pierce, pinch, squeeze)	Angry
3	Kill (slay, murder, slaughter, destroy)	Angry
4	Escape (evade, flee, abscond)	Happy
5	Begin (start, commence)	N/A

2.4 Matching of CADs among Entity Pairs

During the emotion recognition stage, the emotion of the subject in a target event is assigned based on the similarity comparison between the Common Action Distribution of the Target Pair (T-Pair, i.e. the event participants) with those of the RE-Pairs. The emotion of the subject in a target event is based on the RE-Pairs with most similar CAD to that of the T-Pair. For instance, for a target event, "a girl met a wolf", suppose that the CAD that a wolf frequently acts on a girl is similar to that of a RE-Pair, "a cat" and "a dog", the emotion of the girl, which is "fear", is assigned according to the Emotion Generation Rules (EGRs) of the RE-Pair, "a cat" and "a dog". The comparison of CADs can be achieved by using "Mean Square Error (MSE)" or "KL divergence" algorithms which are commonly adopted to compare histograms [12]. For a T-Pair, if the passive-style

sentences outnumber the active-style sentences, the matching of the CADs to the RE-Pairs is based on the "passive-style reference pair set", and vice versa.

In practice, the matching for the CADs between the T-Pair and RE-Pair are based on a coarse-grained verb categorization. Targeting at emotion sensing applications with the big-six emotions [10], including Happy, Sad, Surprise, Angry, Fear, and Disgust, we manually grouped widely-used verbs into 9 different categories roughly according to their "degree of pleasure" from positive to negative. The core of this categorization process is to group verbs that have an "affection" overlap according to the emotion categories to be automatically sensed in an application. For example, consider two different events, "A mouse is chased by a cat" and "A mouse lost his baby". If the emotion categories desired to be sensed are merely "Happy" and "Unhappy", both verbs, "chase" and "loss" can be classified into the same group, say "negative actions". However, for a system to classify four emotion categories, "Happy", "Fear", "Disgust" and "Sad", the verb "chase" and "loss" may better be classified to different categories so as to distinguish sadness-invoking events to fear-invoking ones. Thereby, in general, when more emotions are considered, more fine-grained verb categorization becomes necessary. At this stage, the criteria for the refinement of verb categorization have been informal and intuitive. Table 2 shows a verb categorization scheme and example verbs which we propose for a system that aims to detect the emotions of "Happy", "Sad", "Angry", "Fear", "Surprise", "Disgust", and "Neutral".

3 Evaluations

To evaluate the performance of the proposed methodology we have conducted the following evaluation experiments. We recruited about 100 university undergraduate students (mostly freshman) majoring in different fields to participate in the online evaluations to determine the accuracy of the emotion of an event obtained using the proposed approach. The evaluations were carried out with a collection of a variety of common events that a person encounters in daily lives. Such a design is particularly tailored for the emotion detection in chatting room applications. The objects of the events are accumulated using web mining techniques based on a number of heuristic lexico-syntactic patterns. Each sentence is parsed by a semantic role labeling tool to automatically identify the object of the event described in the collected sentences. We use a publicly available semantic role labeling tool, ASSERT [2] to perform the task. For example, with a query string "she bought a * yesterday", we receive a set of sentences that contain the desired objects such as "chicken nuggets", "i-pod", etc.

The advantages of this web mining approach for accumulating the objects as compared to a dictionary lookup are multifold. First of all, the objects described in Web pages cover a wide spectrum of terms and concepts which is way beyond that of lexical resources such as WordNet. Many modern terms and domain specific terms, such as Wii, iPhone4S, Mercedes S600, Rolex, and Chanel, which can be found in daily conversations were also obtained easily. Secondly, the

modifiers to the noun phrase of the object are captured together. This would allow us to test the robustness of an emotion recognition system dealing with events such as "She saw a drunk driver on the street".

The kinds of emotions chosen in this evaluation is the big-six emotions [15], including Happy, Sad, Surprise, Angry, Fear, and Disgust which are commonly used in chatting room related applications. Totally, there are 6 passive-style reference entity pairs and 4 active-style reference entity pairs. The evaluation is based on automatically composed sentences that describe different events linking a human subject with the objects collected. For example, when we use I (no gender), boy, and girl as the subjects, sample sentences being "I saw a car accident", "A boy plays a Wii", "A girl buys a hamburger", etc. are composed dynamically. By randomly combining a verb with the different subject and object collected (roughly 1700 or so), we easily generated a huge number of test sentences for the evaluation experiments.

For each sentence, an assessor was required to fill in the online assessment form to assess the degree of satisfactory of the emotion yielded by the system for the sentence. Degree of agreement on evaluation of the automatic emotion assignment is based on a 5-point Liker scale with the statements listed by categories of "strongly agree", "agree", "neutral", "disagree", and "strongly disagree". Whenever a dynamically generated sentence is meaningless (for example, an ant plays a computer) or beyond the knowledge of the assessor, he or she is allowed to skip the question. A score of 5 indicated that they strongly agreed with the emotion outcome generated by our approach, and 1 meant that they strongly disagreed.

Figure 2 lists the evaluation results for the two different experiments respectively. For the first session of the assessment where the emotions were given

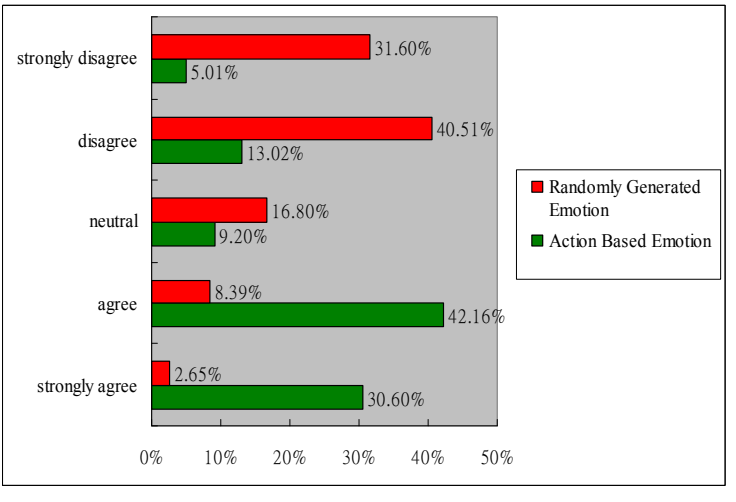


Fig. 2. Evaluation results

Table 2. Categories for the verbs in CADs, from Categories 1 (most positive) to Categories 9 (most negative), for emotion sensing applications for the big-six emotions

1	2	3	4	5	6	7	8	9
Love	Sing	Save	Begin	Find	Put	Hide	Challenge	Shout
Marry	Dance	Create	Conduct	Group	Run	Astonish	Warn	Leave
Wed	Solve	Aim	Deliver	Increase	Evolve	Addict	Worry	Pollute
Engage	Ease	Collect	Emphasize	Live	Change	Decrease	Compete	Remove
Congratulate	Cooperate	Buy	Expect	Meet	Lean	Examine	Forget	Lie
Entertain	Ripen	Accept	Express	Assign	Sit	Evaluate	Disturb	Abandon
Greet	Ride	Espouse	Prepare	Classify		Finish	Bear	Bite
Encourage	Guarantee	Access	Plan				Tolerate	Kick
		Ask Acquire	Grow				Fall	Hate
		Open	Abound					Kill
		Want						
		Most Positive		⟶		Most Negative		

randomly, the average score of the evaluation is approximately 1.8 on the five point scale (72.11% of assessors "strongly disagree" and "disagree"). For the second session of the assessment where the emotions are given by our emotion detection system, the average rating of the evaluation is approximately 3.8 (72.76% of assessors "strongly agree" and "agree"). The evaluation findings indicate clearly that the precision of the emotion detection system based on the proposed approach using the common actions was rather satisfactory.

4 Conclusions

This paper proposes an event-level textual emotion detecting approach based on the common action distribution between event participants. We accumulate these common actions by adopting a web-based text mining approach. No need of any large-scale lexical sources or knowledgebase is required in the proposed framework. Our results indicate that the approach for event-level emotion detection is very robust and has achieved a high precision ratio.

Acknowledgement. This work was supported in part by the National Science Council of Taiwan under grants NSC 98-2221-E-001-010-MY3. The authors would like to thank PIXNET (http://www.pixnet.net) for providing dataset and platform for further research work.

References

1. Wu, C., Chuang, Z., Lin, Y.: Emotion recognition from text using semantic labels and separable mixture models. ACM Transactions on Asian Language Information Processing (TALIP) 5(2), 165–183 (2006)
2. Pradhan, S., Hacioglu, K., Ward, W., Martin, J., Jurafsky, D.: Semantic role parsing: Adding semantic structure to unstructured text. In: Third IEEE International Conference on Data Mining, ICDM 2003, pp. 629–632 (2003)

3. Carreras, X., Màrquez, L.: Introduction to the CoNLL-2005 shared task: Semantic role labeling. In: Proceedings of the Ninth Conference on Computational Natural Language Learning, pp. 152–164 (2005)
4. Liu, H., Lieberman, H., Selker, T.: A model of textual affect sensing using real-world knowledge. In: Proceedings of the 8th International Conference on Intelligent User Interfaces, pp. 125–132 (2003)
5. Ortony, A., Clore, G., Collins, A.: The cognitive structure of emotions. Cambridge Univ. Pr. (1990)
6. Alm, C., Roth, D., Sproat, R.: Emotions from text: machine learning for text-based emotion prediction. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 579–586 (2005)
7. Lu, C., Lin, S., Liu, J., Cruz-Lara, S., Hong, J.: Automatic event-level textual emotion sensing using mutual action histogram between entities. Expert Systems with Applications 37(2), 1643–1653 (2010)
8. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics 28(3), 245–288 (2002)
9. Etzioni, O., Cafarella, M., Downey, D., Popescu, A., Shaked, T., Soderland, S., Weld, D., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. Artificial Intelligence 165(1), 91–134 (2005)
10. Dang, H., Palmer, M.: The role of semantic roles in disambiguating verb senses. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 42–49 (2005)
11. Pradhan, S., Ward, W., Hacioglu, K., Martin, J., Jurafsky, D.: Shallow semantic parsing using support vector machines. In: Proceedings of HLT/NAACL 2004, p. 233 (2004)
12. Kullback, S., Leibler, R.: On information and sufficiency. The Annals of Mathematical Statistics 22(1), 79–86 (1951)
13. Strapparava, C., Valitutti, A.: WordNet-Affect: an affective extension of WordNet. In: Proceedings of LREC, vol. 4, pp. 1083–1086 (2004)
14. Calvo, R., D'Mello, S.: Affect detection: An interdisciplinary review of models, methods, and their applications. IEEE Transactions on Affective Computing 1(1), 18–37 (2010)
15. Ekman, P.: Facial expression and emotion. American Psychologist 48(4), 384 (1993)
16. Soon, W., Ng, H., Lim, D.: A machine learning approach to coreference resolution of noun phrases. Computational linguistics 27(4), 521–544 (2001)

Towards a Chinese Common and Common Sense Knowledge Base for Sentiment Analysis

Erik Cambria[1], Amir Hussain[2], Tariq Durrani[3], and Jiajun Zhang[4]

[1] Temasek Laboratories, National University of Singapore, Singapore
cambria@nus.edu.sg
[2] Dept. of Computing Science and Mathematics, University of Stirling, UK
ahu@cs.stir.ac.uk
[3] Dept. of Electronic and Electrical Engineering, University of Strathclyde, UK
durrani@strath.ac.uk
[4] National Laboratory of Pattern Recognition, Chinese Academy of Sciences, China
jjzhang@nlpr.ia.ac.cn
http://sentic.net

Abstract. To date, the majority of sentiment analysis research has focused on English language. Recent studies, however, show that non-native English speakers heavily support the growing use of Internet. Chinese, specifically, is poised to outpace English as the dominant language online in a few years' time. So far, just a few isolated research endeavors have been undertaken to meet the demands of real-life Chinese web environments. Natural language processing research endeavor, in fact, primarily depends on the availability of resources like lexicons and corpora, which are still very limited for sentiment analysis research in Chinese language. To this end, we are developing a Chinese common and common sense knowledge base for sentiment analysis by blending the largest existing taxonomy of English common knowledge with a semantic network of English common sense knowledge, and by using machine translation techniques to effectively translate its content into Chinese.

Keywords: AI, NLP, KR, Sentiment Analysis, Sentic Computing.

1 Introduction

Sentiment analysis deals with information retrieval and knowledge discovery from text using data mining and natural language processing (NLP) techniques to distill knowledge and opinions from the huge amount of information on the World Wide Web [1]. Last year, the number of Chinese web users soared to 450 million (more than a third of China's population), according to Wang Chen, head of China's State Council Information Office. The United States of America boast just under half that many, but since English is more widely spoken globally, a majority of sites are published in that language. Statistics culled by the marketing firm Internet World Stats[1], in fact, show that, with Web use in China growing at such a rapid rate, it could take less than five years for Chinese to become the most popular language on the Net (Fig. 1).

[1] http://internetworldstats.com

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 437–446, 2012.

Hence, the opportunity to capture the opinions of Chinese web users about social events, political movements, company strategies, marketing campaigns, and product preferences is raising more and more interest both in the scientific community, for the exciting open challenges, and in the business world, for the remarkable fallouts in marketing and financial market prediction in the Middle Kingdom. The limited availability of resources like lexicons and corpora in Chinese language, however, makes such research endeavor even more challenging than English sentiment analysis. Mining Chinese opinions and sentiments from the Web, in fact, is an extremely difficult task that involves a deep understanding of most of the explicit and implicit, regular and irregular, syntactical and semantic rules proper of the Chinese language.

Besides syntactical rules, one of the main hurdles to overcome in Chinese sentiment analysis research is the lack of both Chinese common knowledge bases and Chinese common sense knowledge bases. Common knowledge represents human general knowledge acquired from the world, e.g., 犬瘟热是国内动物疫病 (canine distemper is a domestic animal disease). Common sense knowledge is some obvious thing that people normally know but usually leave unstated, e.g., 猫可以捕猎老鼠 (cat can hunt mice) and 猫是可爱的 (cat is cute). It is through the combined use of common and common sense knowledge that we can have a grip on both low and high level concepts in natural language sentences and, hence, effectively communicate with other people without having to continuously ask for definitions and explanations. Common sense knowledge, moreover, enables the propagation of sentiment from affect words, e.g., 高兴 (happy) and 伤心 (sad), to general concepts, e.g., 生日礼物 (birthday gift), 高中毕业 (school graduation), 癌症 (cancer) and 犬瘟热 (canine distemper), which is useful for tasks such as sentiment elicitation and polarity detection.

In this work, we blend ProBase [2], the largest existing taxonomy of English common knowledge, with ConceptNet [3], a natural-language-based semantic network of English common sense knowledge, and use an augmented string-to-tree model to effectively translate the newly built knowledge base into Chinese. The structure of the paper is as follows: Section 2 presents related works in the field of sentiment analysis and knowledge base design in Chinese language, Section 3 explains motivations and strategies behind the development of a common and common sense knowledge base and illustrates the machine translation techniques adopted, Section 4, eventually, comprises concluding remarks and future directions.

2 Related Work

The growth of online Chinese social media such as Weibo[2], QZone[3], and Renren[4] is fostering the proliferation of many business and research activities around the field of Chinese sentiment analysis.

[2] http://weibo.com
[3] http://qzone.qq.com
[4] http://renren.com

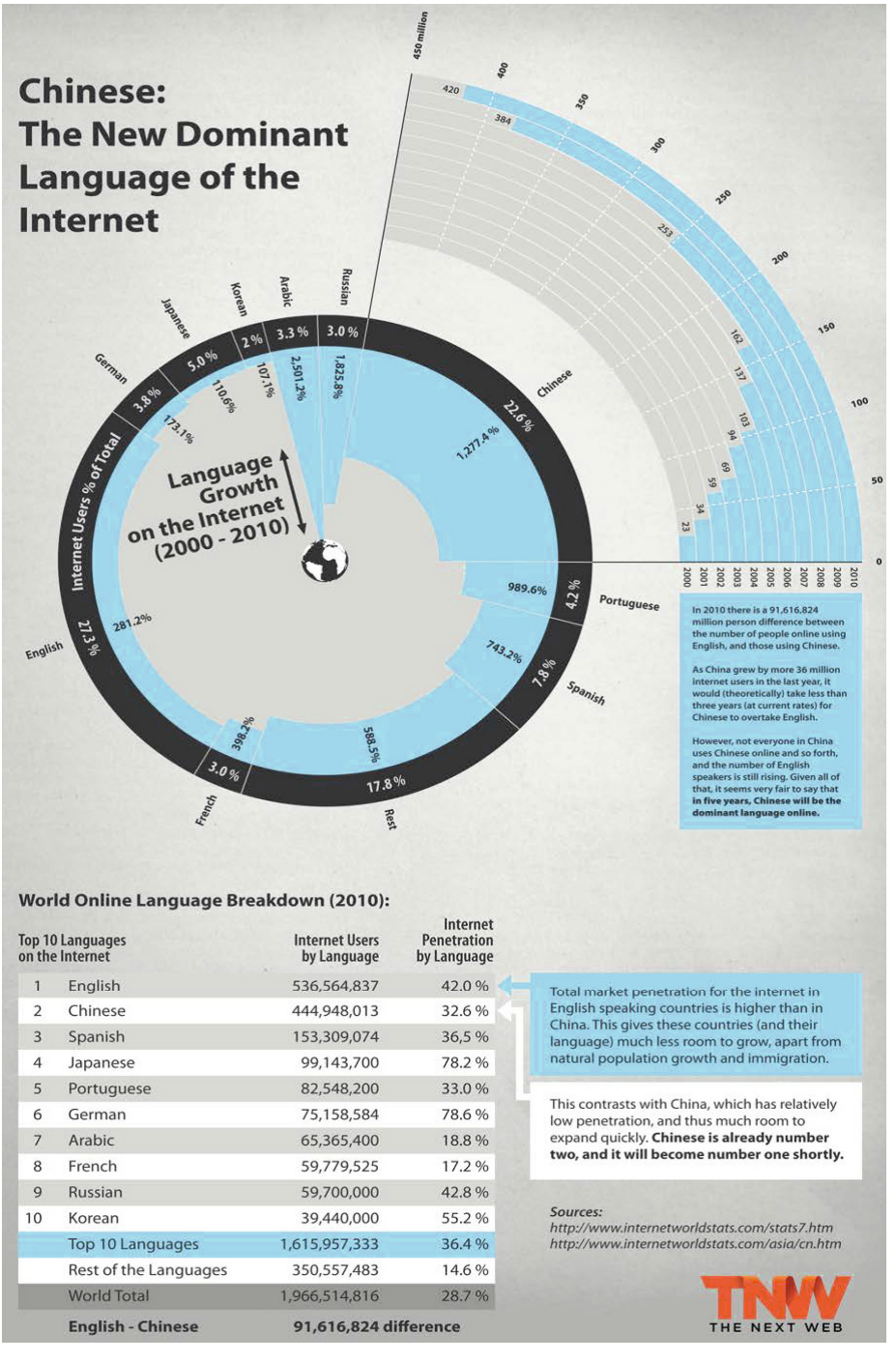


Fig. 1. An infographic by Nextweb showing that, with Chinese Internet growth rising at the rate it is, it could be less than five years before Chinese becomes the dominant language on the Internet

The automatic analysis of user generated contents such as online news, reviews, blogs and tweets in Chinese language, in fact, can be extremely valuable for tasks such as mass opinion estimation, corporate reputation measurement, political orientation categorization, stock market prediction, and customer preference study in the Middle Kingdom. Distilling useful information from such unstructured data, however, is a multi-faceted and multi-disciplinary problem as opinions and sentiments can be expressed in a multitude of forms and combinations in which it is extremely difficult to find any kind of regular behavior [4]. A lot of conceptual rules, in fact, govern the expression of opinions and sentiments in Chinese language and there exist even more clues that can convey these concepts from realization to verbalization.

Most of current approaches to Chinese sentiment analysis rely on rather unambiguous affective keywords extracted from an existing knowledge base or set of documents [5,6,7,8,9,10,11] or from a purpose-built lexicon based on a domain-dependent corpus [12,13,14,15,16]. Such approaches are still far from being able to perfectly extract the cognitive and affective information associated with natural language and, hence, often fail to meet the golden standard of human annotators.

The main aim of this work is to build possibly the most comprehensive resource of common and common sense knowledge in order to perform a domain-independent clause-level analysis of Chinese opinions and sentiments on the Web. Previous attempts to build a similar resource have been made by HowNet [17], an online common sense knowledge base unveiling inter-conceptual relations and inter-attribute relations of concepts as connoting in lexicons of the Chinese and their English equivalents, and ConceptNet5[5]. Such knowledge bases, however, simply contain common sense knowledge and lack common knowledge, which is particularly useful for the sentiment analysis subtasks of topic and feature spotting.

3 Building the Knowledge Base

In standard human-to-human communication, people usually refer to existing facts and circumstances and build new useful, funny or interesting information on the top of those. This common knowledge comprehends information usually found in news, articles, debates, lectures, etc. (factual knowledge) but also principles and definitions that can be found in collective intelligence projects such as Wikipedia[6] (vocabulary knowledge).

However, when people communicate with each other, in fact, they also rely on similar background knowledge, e.g., the way objects relate to each other in the world, people's goals in their daily lives and the emotional content of events or situations. This taken for granted information is what we call common sense, obvious things people normally know and usually leave unstated.

[5] `http://conceptnet5.media.mit.edu`
[6] `http://en.wikipedia.org`

3.1 English Common Knowledge

Attempts to build an English common knowledge base comprehend both resources crafted by human experts or community efforts, such as WordNet [18], a lexical knowledge base of about 25,000 words grouped into an ontology of synsets, or Freebase [19], a social database of 1,450 concepts, and automatically-built knowledge bases, such as WikiTaxonomy [20], a taxonomy of about 127,000 concepts extracted from Wikipedia's category links, or ProBase[7].

ProBase contains about 12 million English concepts learned iteratively from 1.68 billion web pages in Bing[8] web repository. The taxonomy is probabilistic, which means every claim in ProBase is associated with some probabilities that model the claim's correctness, ambiguity, and other characteristics. The probabilities are derived from evidences found in web data, search log data, and other available data. The core taxonomy consists of the "IsA" relationships extracted by using syntactic patterns such as the Hearst patterns [21]. For example, a segment like "artists such as Pablo Picasso" can be considered as a piece of evidence for the claim that 'pablo picasso' is an instance of the concept 'artist'.

3.2 English Common Sense Knowledge

One of the biggest projects aiming to build a comprehensive common sense knowledge base is Cyc [22]. Cyc requires knowledge engineers working on some specific languages, and contains just 120,000 concepts as this is labor intensive and time consuming. A more recent project is Open Mind Common Sense (OMCS), which has been collecting pieces of knowledge from volunteers on the Internet since 2000 by enabling the general public to enter common sense into the system with no special training or knowledge of computer science. OMCS exploits these pieces of common sense knowledge to automatically build ConceptNet, a semantic network of 173,398 nodes (Fig. 2). WordNet contains very detailed descriptions of every word's various senses but it does not include enough general Web information. ProBase, which provides more concepts, includes pieces of knowledge that match general distribution of human knowledge. ConceptNet, in turn, contains implicit knowledge that people rarely mention on the Web, which is a good complementary material to ProBase.

To this end, in this work we blend ProBase and ConceptNet and, hence, build a comprehensive knowledge base that can be seen as one of the first attempts to emulate how tacit and explicit knowledge are organized in human mind and how this can be exploited to perform reasoning within natural language tasks. Providing a machine with a database of millions of common and common sense concepts, in fact, would still be not enough for it to be intelligent: it needs to be taught how to handle this knowledge, retrieve it when necessary, make analogies, and learn from experience [23].

[7] `http://research.microsoft.com/probase`
[8] `http://bing.com`

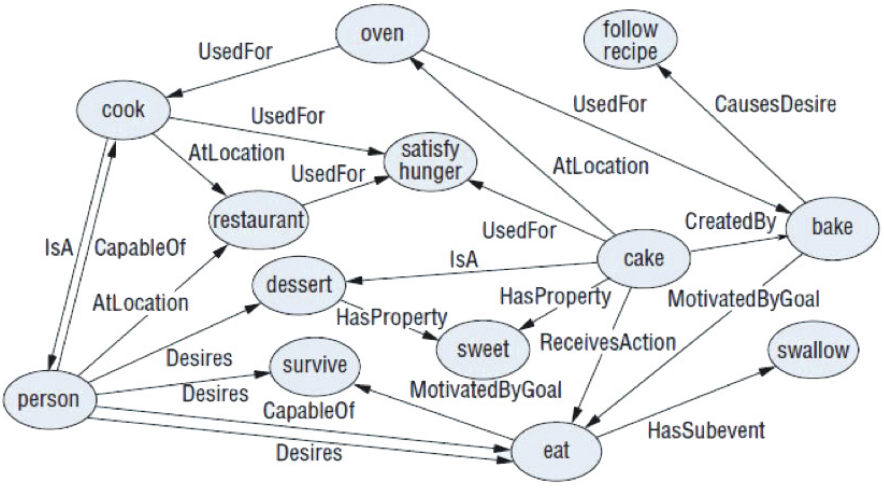


Fig. 2. ConceptNet represents the information in the Open Mind corpus as a directed graph where nodes are concepts and labeled edges are assertions of common sense that interconnect them

3.3 English Common and Common Sense Knowledge

In this work, we focus on IsA relationships to build a semantic network of common knowledge, which we call Isanette (IsA net) [24]. It represents hyponym-hypernym relationships as a 2,715,218 × 1,331,231 matrix having instances (e.g., 'pablo picasso') as rows and concepts (e.g., 'artist') as columns. Performing reasoning on Isanette as it is, however, is not very convenient as it is a very large and fat matrix that contains noise and multiple forms, since all of the evidences are automatically extracted from the Web. To this end, we firstly clean it by applying different NLP techniques and, secondly, enhance its consistency and further reduce its sparseness by adding complementary common sense knowledge.

We build Isanette out of 23,066,575 IsA triples extracted with the form <instance, concept, confidence score>. Before generating the matrix from these statements, however, we need to solve two main issues, namely multiple word forms and low connectivity. We address the former issue by processing both subjects and objects of triples with OMCS lemmatizer, which groups together the different inflected forms of words (different cases, plurals, verb tenses, etc.) so that they can be stored in Isanette as a single item. In case of duplicates, we simply consider the triple with higher confidence score. As for Isanette's connectivity, if we want to apply dimensionality reduction techniques on it in order to find similar patterns, we would like the matrix to be as less sparse as possible. To this end, we firstly want to get rid of hapax legomena, that is instances/concepts with singular out-/in-degree.

These nodes can be useful for specific tasks such as finding the meaning of uncommon instances or give an example of a rare concept. For more general reasoning tasks, however, hapax legomena are very bad as they enlarge dimensionality without providing overlapping information that can be useful for finding similar patterns and perform analogies. In this work, we choose to discard not only hapax legomena but also the other nodes with low connectivity, in order to heavily reduce Isanette's sparseness.

In particular, we used a trial and error approach and found that the best trade-off between size and sparseness is achieved by setting the minimum node connectivity equal to 10. This cut-off operation leaves out almost 40% of nodes and makes Isanette a strongly connected core. Moreover, we exploit dimensionality reduction techniques to infer negative evidence such as 'carbonara' is not a kind of 'fuel' or 'alitalia' is not a 'country', which is very useful to further reduce Isanette's sparseness and improve reasoning algorithms.

3.4 Chinese Common and Common Sense Knowledge

As a subsumption common knowledge base, Isanette lacks information like a 'dog' is a 'best friend' (rather than simply an 'animal') or a 'rose' is a kind of 'meaningful gift' (rather than simply a kind of 'flower'), that is common sense that is not usually stated in web pages (or at least not that often to be extracted by Hearst patterns with a high enough confidence score). To overcome this problem, we enrich Isanette with complementary hyponym-hypernym common sense knowledge from ConceptNet. In particular, we extract from the Open Mind corpus all the assertions involving IsA relationships with a non-null confidence score, such as "dog is man's best friend" or "a birthday party is a special occasion". We exploit these assertions to generate a directed graph of about 15,000 nodes, interconnected by IsA edges [25].

To merge this subsumption common sense knowledge base with Isanette, we use blending [26], a technique that performs inference over multiple sources of data simultaneously, taking advantage of the overlap between them. Blending combines two sparse matrices linearly into a single matrix in which the information between the two initial sources is shared. This alignment operation yields a new strongly connected core, $C \in \mathcal{R}^{m \times n}$, in which common and common sense knowledge coexist, i.e., a matrix 340,000 $\times$ 200,000 whose rows are instances such as 'birthday party' and 'china', whose columns are concepts like 'special occasion' and 'country', and whose values indicate truth values of assertions.

The last step to build our knowledge base for Chinese sentiment analysis is to effectively translate the pieces of English common and common sense knowledge so far obtained into Chinese. Because Isanette contains a lot of multiple-word concepts, a word-based [27] or a phrase-based [28] machine translation technique would not be suitable. Hence, we go for a fuzzy syntax-based model [29] (Fig. 3). Such technique exploits the GHKM algorithm [30] for extracting (minimal) string-to-tree translation rules from a triple (f, e_t, a), where f is the source-language sentence, e_t is a target-language parse tree whose yield e is the translation of f, and a is the set of word alignments between e and f.

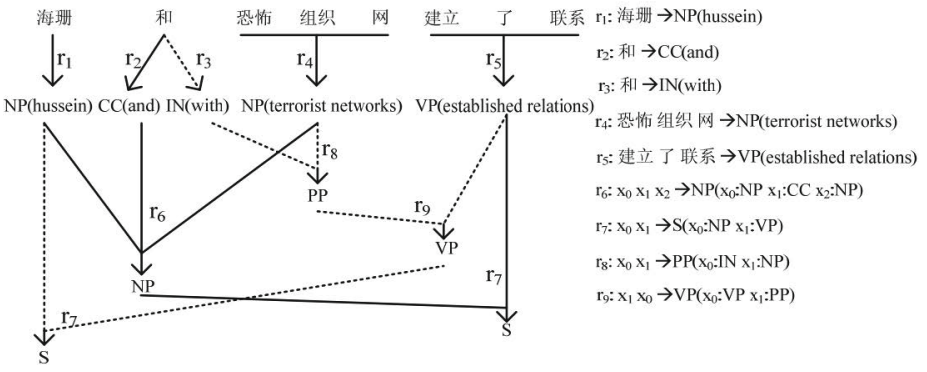


Fig. 3. A sample string-to-tree translation operated by the adopted fuzzy syntax-based model. Rules used are listed on the right.

The basic idea of GHKM is to obtain the set of minimally-sized translation rules which can explain the mappings between source string and target parse tree. The machine translation technique adopted, in particular, uses fuzzy matching to calculate similarity. While 0-1 matching assigns similarity 1 for exact matches and 0 for mismatch, and likelihood matching directly utilizes the likelihood to measure the similarity, we go one step further by adopting a measure of deep similarity, computed using latent distributions of syntactic categories.

Huang et al. [31] proposed this method to compute the similarity between two syntactic tag sequences, used to impose soft syntactic constraints in hierarchical phrase-based models. Analogously, we borrow this idea to calculate the similarity between two syntax-augmented machine translation (SAMT) syntactic categories, and then apply it to calculate the degree of matching between a translation rule and the syntactic category of a test source string for purposes of fuzzy matching.

Such procedure, termed deep similarity matching, allows to represent each category by a real-valued feature vector, instead of directly using SAMT-style syntactic categories. With the real-valued vector representation for each SAMT-style syntactic category, the degree of similarity between two syntactic categories can be simply computed as a dot-product of their feature vectors. This computation yields a similarity score ranging from 0 (totally different syntactically) to 1 (totally identical syntactically), which ultimately allows a better translation of multi-word concepts from English into Chinese.

4 Conclusions and Future Work

In this work, we blended together English common and common sense knowledge and used an augmented string-to-tree model to effectively translate such knowledge into Chinese, in order to build probably the most comprehensive resource for Chinese sentiment analysis.

In the future, we plan to carry out a thorough evaluation of the resource, which is currently underway. Further research studies are now planned to investigate if a better trade-off between size and sparseness of the resource can be found. At the same time, we plan to explore multi-dimensionality reduction techniques to perform reasoning on the knowledge base.

Even if we manage to teach a machine 15 million and such things, in fact, it will still be not enough for it to be intelligent: it needs to be taught how to handle this knowledge, retrieve it when necessary, make analogies, and learn from experience.

Acknowledgments. This work has been part-funded by the Royal Society of Edinburgh (UK) and the Chinese Academy of Sciences in Beijing (P.R. China), within the China-Scotland SIPRA (Signal Image Processing Research Academy) Programme.

References

1. Cambria, E., Grassi, M., Hussain, A., Havasi, C.: Sentic computing for social media marketing. Multimedia Tools and Applications (2011), http://dx.doi.org/10.1007/s11042-011-0815-0
2. Wu, W., Li, H., Wang, H., Zhu, K.: Towards a probabilistic taxonomy of many concepts. Technical Report MSR-TR-2011-25, MSRA (2011)
3. Havasi, C., Speer, R., Alonso, J.: ConceptNet 3: A flexible, multilingual semantic network for common sense knowledge. In: Proceedings of RANLP, Borovets (2007)
4. Cambria, E., Benson, T., Eckl, C., Hussain, A.: Sentic PROMs: Application of sentic computing to the development of a novel unified framework for measuring health-care quality. Expert Systems with Applications (2012), http://dx.doi.org/10.1016/j.eswa.2012.02.120
5. Ye, Q., Lin, B., Li, Y.: Sentiment classification for Chinese reviews: A comparison between SVM and semantic approaches. In: Proceedings of the International Conference on Machine Learning and Cybernetics, vol. 4, pp. 2341–2346 (2005)
6. Ye, Q., Shi, W., Li, Y.: Sentiment classification for movie reviews in Chinese by improved semantic oriented approach. In: Proceedings of HICSS (2006)
7. Zagibalov, T., Carroll, J.: Unsupervised classification of sentiment and objectivity in Chinese text. In: Proceedings of the International Joint Conference on Natural Language Processing, pp. 304–311 (2008)
8. Zagibalov, T., Carroll, J.: Automatic seed word selection for unsupervised sentiment classification of Chinese text. In: Proceedings of COLING, vol. 1, pp. 1073–1080 (2008)
9. Tan, S., Zhang, J.: An empirical study of sentiment analysis for Chinese documents. Expert Systems with Applications 34, 2622–2629 (2008)
10. Zhang, C., Zeng, D., Li, J., Wang, F., Zuo, W.: Sentiment analysis of Chinese documents: From sentence to document level. Journal of the American Society for Information Science and Technology 60, 2474–2487 (2009)
11. Zhai, Z., Xu, H., Li, J., Jia, P.: Sentiment classification for Chinese reviews based on key substring features. In: Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering, pp. 1–8 (2009)

12. Zhu, Y., Min, J., Zhou, Y., Huang, X., Wu, L.: Semantic orientation computing based on HowNet. Journal of Chinese Information Processing 20, 14–20 (2006)
13. Xu, L., Lin, H., Yang, Z.: Text orientation identification based on semantic comprehension. Journal of Chinese Information Processing 21, 96–100 (2007)
14. Xu, J., Ding, Y., Wang, X.: Sentiment classification for Chinese news using machine learning methods. Journal of Chinese Information Processing 21, 95–100 (2007)
15. Yao, T., Lou, D.: Research on semantic orientation analysis for topics in Chinese sentences. Journal of Chinese Information Processing 21, 73–79 (2007)
16. Yang, J., Hou, M., Wang, N.: Recognizing sentiment polarity in Chinese reviews based on topic sentiment sentences. In: Proceedings of NLP-KE (2010)
17. Dong, Z., Dong, Q.: HowNet and the Computation of Meaning. World Scientific (2000)
18. Fellbaum, C.: WordNet: An Electronic Lexical Database (Language, Speech, and Communication). The MIT Press (1998)
19. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: A collaboratively created graph database for structuring human knowledge. In: Proceedings of SIGMOD, Vancouver, 1247–1250 (2008)
20. Ponzetto, S., Strube, M.: Deriving a large-scale taxonomy from Wikipedia. In: AAAI, Vancouver, 1440–1445 (2007)
21. Hearst, M.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of COLING, Nantes, pp. 539–545 (1992)
22. Lenat, D., Guha, R.: Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project. Addison-Wesley (1989)
23. Cambria, E., Hussain, A.: Sentic Computing: Techniques, Tools, and Applications. Springer, Heidelberg (2012)
24. Cambria, E., Song, Y., Wang, H., Hussain, A.: Isanette: A common and common sense knowledge base for opinion mining. In: Proceedings of ICDM, Vancouver (2011)
25. Cambria, E., Havasi, C., Hussain, A.: SenticNet 2: A semantic and affective resource for opinion mining and sentiment analysis. In: Proceedings of FLAIRS, Marco Island (2012)
26. Havasi, C., Speer, R., Pustejovsky, J., Lieberman, H.: Digital intuition: Applying common sense using dimensionality reduction. IEEE Intelligent Systems 24(4), 24–35 (2009)
27. Brown, P., Della Pietra, S., Della Pietra, V., Mercer, R.: The mathematics of machine translation: Parameter estimation. Computational Linguistics 2(19), 263–311 (1993)
28. Koehn, P., Och, F., Marcu, D.: Statistical phrase-based translation. In: Proceedings of HLT/NAACL (2003)
29. Zhang, J., Zhai, F., Zong, C.: Augmenting string-to-tree translation models with fuzzy use of source-side syntax. In: Proceedings of EMNLP, pp. 204–215 (2011)
30. Galley, M., Hopkins, M., Knight, K., Marcu, D.: What's in a translation rule. In: Proceedings of HLT/NAACL, pp. 273–280 (2004)
31. Huang, Z., Cmejrek, M., Zhou, B.: Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions. In: Proceedings of EMNLP, pp. 138–147 (2010)

Emotion Tracking on Blogs - A Case Study for Bengali

Dipankar Das, Sagnik Roy, and Sivaji Bandyopadhyay

Department of Computer Science and Engineering, Jadavpur University, India
{dipankar.dipnil2005,sagnikroy59}@gmail.com,
sivaji_cse_ju@yahoo.com

Abstract. Rapid growth of blogs in the Web 2.0 and the handshaking between multilingual search and sentiment analysis motivate us to develop a blog based emotion analysis system for Bengali. The present paper describes the identification, visualization and tracking of bloggers' emotions with respect to time from Bengali blog documents. A simple pre-processing technique has been employed to retrieve and store the bloggers' comments on specific topics. The assignment of Ekman's six basic emotions to the bloggers' comments is carried out at word, sentence and paragraph level granularities using the Bengali *WordNet Affect Lists*. The evaluation produces the *precision*, *recall* and *F-Score* of 59.36%, 64.98% and 62.17% respectively for 1100 emotional comments retrieved from 20 blog documents. Each of the bloggers' emotions with respect to different timestamps is visualized by an emotion graph. The emotion graphs of 20 bloggers demonstrate that the system performs satisfactorily in case of emotion tracking.

Keywords: Emotion, Tracking, Blog, Bengali WordNet Affect.

1 Introduction

Human-machine interface technology has been investigated for several decades. Scientists have found that emotion technology can be an important component in artificial intelligence [20]. Recent research has placed more emphasis on the recognition of nonverbal information, and has especially focused on emotion reaction. Though emotions are not linguistic things, the most convenient access that we have to them is through the language [22].

Nowadays, in the Natural Language Processing (NLP) communities, several research activities on sentiment and/or emotion analysis are in full swing. Sentiment of people is important as it has great influence on our society. Recently, the identification of the temporal trends of sentiments on different topics has drawn attention of NLP communities [8] [6]. The perspectives of sociology, psychology and commerce along with the close association among people, topic and sentiment motivate us to investigate the insides of emotional changes of people over topic and time [29].

In psychology and common use, emotion is an aspect of a person's mental state of being, normally based in or tied to the person's internal (physical) and external (social) sensory feeling [30]. The determination of emotions expressed in the text is

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 447–456, 2012.

itself a challenging issue as emotion is not open to any objective observation or verification [19]. Moreover, the same textual content can be presented with different emotional slants [9]. Hence, the actual challenge lies in the identification of the true emotional stances from text [1].

On the other hand, emotion analysis is becoming a recent sub discipline at the crossroads of information retrieval [21] and computational linguistics [25]. Information is concerned not only with the topic of a document but also with the emotion it expresses. It has a rich set of applications such as tracking users' emotion about products or events or about political candidates as expressed in online forums, customer relationship management, stock market prediction, social networking etc.

Emails, weblogs, chat rooms, online forums and even twitter are being considered as the affective communication substrates to analyze the reaction of emotional catalysts. Blog is becoming one of the crucial communicative and informative repositories of text based emotional contents in the Web 2.0 [13]. The reason may be the explosive growth of the social media content on the Web in the past few years. Many blogs act as the online diary of the bloggers reporting daily activities. Sometimes, blog posts are annotated by other bloggers. The blog is therefore considered in our present attempt for analyzing the reactions of users' emotions associated with various timestamps.

The present task involves the identification, visualization and tracking of bloggers' emotions with respect to time. The blog posts are collected from the Bengali web blog archive. The sections of the bloggers' comments for a given topic contain nested tree like structures along with distinguishable and useful information regarding individual blogger such as document identification number (#docid), section (#sectionid) and timestamp (#tid). The topic and comments with respect to each of the bloggers are visually represented based on the timestamp. The emoticons and their corresponding emotion tags are stored manually in a separate knowledgebase and are used for sentential emotion tagging.

It is said that sentiment is typically a localized phenomenon that is more appropriately computed at the paragraph, sentence or entity level granularities [14]. Thus, we have identified Ekman's [7] six basic emotions from the bloggers' comments at sentence and paragraph level granularities using the information of word level constituents. The word level lexical information was acquired from the Bengali *WordNet Affect* Lists [5]. The lexicon based baseline system achieves the precision, recall and F-Score of 59.36%, 64.98% and 62.17% respectively while evaluating 1100 emotional sentences. It was observed that the baseline system suffers due to the presence of inflected Bengali surface words in the text. Therefore, we have employed an open source Bengali shallow parser for identifying the root forms of the words. The morphology based baseline system was improved by achieving the average F-Score of 3.13% with respect to all six emotion classes. We have considered the individual comment section of a blogger as a separate paragraph. Thus, the sentence level emotions that are acquired from word level are assigned to the corresponding comment sections at the paragraph level.

The cumulative emotions acquired from the user comments at a particular time instance are termed as emotional intensity. The emotional intensities with respect to a

specific blogger are ordered based on the temporal ordering of the associated timestamps. Based on such temporal ordering, the emotions of a blogger are represented using a graph. The time based plotting of the emotions in graphical structure shows the change of emotions with respect to a specific blogger as well as the change of emotions for different bloggers on the same topic at the same or different timestamps. The manual evaluation of the tracking through emotion graphs shows that the system satisfactorily demonstrated the tracking of bloggers' emotions.

The rest of the paper is organized as follows. Section 2 describes the related work. The acquisition of bloggers' information from the nested comment sections is discussed in Section 3. Automatic emotion tagging of the bloggers' comments and evaluation are described in Section 4. The generation and analysis of emotion graphs are discussed in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

Major studies on Opinion Mining and Sentiment Analysis have been attempted with more focused perspectives rather than fine-grained emotions [18]. A related study of sentence level emotional affinity was carried out using emotion lexicon and handcrafted rules [15]. Whitelaw et al. [23] constructed a lexicon that provides appraisal attributes for terms. The authors have used them as features for classification along with the bag-of-words model. Leopold et al. [12] has shown that the pre-processing steps like stemming and lemmatization have been found to be detrimental to classification accuracy. In contrast, the present technique based on the Bengali *WordNet Affect* Lists has been improved by incorporating morphological knowledge into account.

Yahoo! Kimo Blog was also used as corpora to build emotion lexicons [26] [27] [28]. In their studies, emoticons were used to identify emotions associated with textual keywords. We have also prepared a similar type of knowledgebase for storing the emoticons and their associated emotion tags. Opinion mining at word, sentence and document levels from news and web blog articles classifies the blog posts into topic and genre independent manner [11]. We have explored the potentiality of emotional words towards sentence and paragraph level granularities.

In case of tracking and visualization, Mishne and de Rijke [17] proposed a system, MoodViews to analyze the temporal change of sentiment. MoodViews analyzes multiple sentiments by using 132 sentiments used in LiveJournal. Although our concept of the emotion graph is similar to MoodViews, we focus on temporal relations between bloggers' comments associated with similar or different types of emotions. Havre et al. [10] proposed a system called ThemeRiver that visualizes thematic flows along with time-line. Although our approach is different from ThemeRiver, we focus on visualization of emotion flows of the bloggers on a given topic based on time. The temporal sentiment identification from social events has been carried out in [8]. In contrast, we have explored the visualization and tracking of the bloggers emotions in Bengali instead of considering any coarse grained sentiments (positive or negative).

But, all the above-cited works have been at-tempted for either English or Chinese. Recent study shows that non-native English speakers support the growing use of the Internet and the rapidly growing web users from multilingual communities focus the attention to improve the multilingual search engines on the basis of sentiment or emotion. Under the domain of multilingual sentiment analysis, work has been carried out for several European languages. Mihalcea et al. [16] explore the possibility of rapidly developing resources in the target language by relying on the resources in English whereas Banea et al. [2] creates lexicon for sentiment analysis in Romanian using a bootstrap method.

To the best of our knowledge, Bengali is the sixth popular language in the world, second in India and the national language of Bangladesh. Though the emotion analysis task for Bengali has been started [3], at present, no work on emotion tracking has been carried out for Bengali or for any Indian language. The present task and methodologies would help in the development of emotion analysis systems for other Indian languages as well.

3 Pre-processing of Corpus

The blog documents are retrieved from the Bengali web blog archive [31] and are stored in the format shown in Figure 1. Each of the blog documents is assigned a unique document identifier (*docid*#) followed by a section devoted for topic and several sections devoted for different users' comments. Each comment section consists of several nested and over-lapped sub sections that also contain the bloggers' comments. Each of the comment sections of an individual blogger is uniquely identified by the notion of section identification number (*secid*#). Each section contains the user identification number (*uid*#) and the associated timestamp (*tid*#).

```
-<DOC docid = xyz>
        +<Topic>…. </Topic>
        -<User Comments id=UC1>
            -<U uid=1, tid=t1, secid=UC1>….
                -<U uid=2, tid=t2, secid=UC1.1>…</U>
                -<U uid=3, tid=t3, secid=UC1.2>…</U>
                    -<U uid=1, tid=t4,
secid=UC1.2.1>…</U>
                    ….
                </U>
          </User Comments>
        +<User Comments id=UC2>
        +<User Comments id=UC3>
          …
  </DOC>
```

Fig. 1. General structure of a blog docu-ment

All the comment sections (identified by the tag <*User Comments id*=UC#>) in the individual blog documents refer to a single topic. However, in the present task each comment section has been considered as a separate unit by assuming that no inter emotional impact exists among the various comment sections within a single blog document.

We have considered the individual comment section as separate paragraph that contains several emotional sentences. The sentences that are present in the bloggers' comment sections were already annotated during the development of Bengali emotional blog corpus [4]. Hence, by default, each of the comment sections is assigned with all types of emotions that are present at the annotated sentences of that section. A total of 1100 emotional sentences on eight different potential topics (Comics, Politics, Sports, Movies, Music, Buzz, Short Stories and Miscellaneous) with respect to 20 different bloggers are considered for conducting our present experiments.

A portion of the whole annotated blog corpus [4] containing 20 blog documents with 1100 sentences is considered in the present task. Each sentence of the corpus was annotated with the emotional components such as emotional expression (word/phrase), intensity, associated holder and topic(s). Ekman's six emotion classes (*anger, disgust, fear, happy, sad* and *surprise*) along with three types of intensities (*high, general* and *low*) were considered for the sentence level annotation. As the sentences were collected from the blog posts, the writers of the blog posts are assumed as the default additional emotion holders [25]. The document level topic and corresponding bloggers' comments are retrieved, filtered from the blog documents and stored in XML file format as shown in Figure 2.

```
<DOC docid=blogextract>
        <Topic>
সবাই মতামত দে
লিখেছে
হাবিব
19 ডিসেম্বর (রবিবার), 2010 10:49
নুরুল কবীর মনে করে
বর্তমানে রাষ্ট্রযন্ত্র স্বাধীন ও গণতান্ত্রিক সাংবাদিকতার বিরুদ্ধে খড়গহস্ত হয়ে পড়েছে ??
রাষ্ট্র যদি সাংবাদিকদের বিরুদ্ধে খড়গ হস্তই হত তবে নুরুল কবীর সে কথা বলে কি করে ??
বিস্তারিত এইখানে দেখেন ।
আপনি কি মনে করেন ?
আমার উত্তরটা দিন ।
 ভাল হয় উত্তরটা ডেসক্রিপটিভ হলে মানে উত্তরের সপক্ষে আপনার যুক্তি তুলে ধরলে ।
        </Topic>

            <U uid=ফারমার , tid=December 19, 2010  10:54 , secid=1>
মডিউর রহমান হাসিনার কাছে মাফ চেয়ে রক্ষা পেলো, বাকীদের কথা ছেড়ে দিলাম।

</U>
            <U uid=ফারমার , tid=December 19, 2010  10:57 , secid=1.1>
ফারমার,ক্রিমিনাল মাহমুদুর রহমান জাফের নাম নিতে হয়তো সমানে হয়রত ও গেছনে স: যোগ করে।

</U>
            <U uid=হাবিব , tid=December 20, 2010  12:54 , secid=1.1.1>
ফারমার, তাই নাকি স্বঘোষিত মুক্তিযোদ্ধা সাহেব ?
```

Fig. 2. XML structure of a blog document

The whole system is represented by several user friendly interactive interfaces. The first step is to select the target blog document from the blog archives. The comments

of each of the bloggers in the selected document are separated based on the #uid information. Separately, a knowledge base (as shown in Table 1) for the emoticons was also prepared by experts after minutely analyzing the Bengali blog data. Each image link of the emoticon in the raw corpus was mapped into its corresponding textual entity in the tagged corpus according to their proper emotion types using this knowledge base. The knowledgebase has also been used during the emotion tagging of the sentences by considering each of the emoticons as a separate word.

Table 1. Knowledge base for emoticons

Emoticon		Tags
☺ , :-)	<emo_icon_happy>	*happy*
☹, :-S	<emo_icon_sad>	*sad*
:-@, :-a	<emo_icon_ang>	*anger*
:-$, :-D	<emo_icon_dis>	*disgust*
:'(, :-F	<emo_icon_fear>	*fear*
:-O, :-P	<emo_icon_sur>	*surprise*
😐, :-|	<emo_icon_ntrl>	*neutral*

Our aim is to identify the bloggers emotions at various timestamps from the comment sections provided with respect to a topic. Therefore, we have passed the sentences of the bloggers' comments to a lexicon based emotion tagging system for identifying the sentence and paragraph level emotions.

4 Emotion Tagging Based on Bengali *WordNet Affect*

The emotion tagging system is based on the lexical words found in the Bengali *WordNet Affect* Lists (*Beng_WAL*) [5]. The *Beng_WAL* has been developed from the affect wordlists already available in English. It is organized in six basic emotion classes such as anger, disgust, fear, joy, sadness and surprise. The representation of the emotional words and synsets of the *Beng_WAL* is as follows (in Figure 3).

a#00117872 প্রকুপিত / ক্রুদ্ধ / রুষিত / রুষ্ট / কুপিত/# অগ্নি /অগ্নিকাণ্ড /উত্তাপ #
ক্ষিপ্ত /চণ্ড /প্রচণ্ড /রুদ্র # উদ্ভ্রান্ত /উন্মদ /উন্মত্ত /খেপা/মত্ত /খাপ্পা # অমর্ষ /অমর্ষণ

Fig. 3. Example of a Translated Bengali Synset

If a word in a sentence is present in any of the Bengali *WordNet Affect* lists; the sentence is tagged with the emotion label corresponding to that affect list. The evaluation of the baseline system based on the *Beng_WAL* is shown in Table 2. It has been observed that the baseline system suffers from the identification of the lexical keywords as the Bengali words are morphologically rich in nature. Thus, we have also incorporated an open source morphological analyzer in our baseline system. Thus, the algorithm has been modified a bit. If any word is not found in any of the six lists, each

word of the sentence is passed through the morphological process to identify the root form which is again searched in the Bengali *WordNet Affect* lists. If the root form is found in any of the six Bengali *WordNet Affect* lists, the sentence is tagged accordingly. Otherwise, the sentence is tagged as non-emotional or neutral.

The average F-Score of the baseline system has been improved by 3.13% with respect to the six emotion classes after incorporating the morphology. It has been observed that due to the fewer number of word level instances in some of the affect lists (e.g., *fear, disgust, surprise*), the performance of the system gives poor results even after including the morphological knowledge. The comparative results of the baseline and morphology based systems for 1100 test sentences were shown in Table 2. It has been observed that though the morphological system outperforms the baseline, both the systems sometimes fails to capture the emotional expressions inscribed in sentences. The reason may be the metaphoric use of emotion or the ungrammatical structure of some blog sentences. The system suffers due to the lexical coverage of the *Beng_WAL* in handling the colloquial words and idioms.

Table 2. Precision (Prec.), Recall (Rec.) and *F-scores* (FS) (in %) of the Baseline and Morphological systems for six emotion classes on the test set

Categories	**Baseline**			**Morph**
	Prec.	**Rec.**	**FS**	**FS**
anger	61.32	67.17	63.54	67.32
disgust	53.44	62.07	57.89	60.12
fear	57.78	65.66	62.84	66.08
joy	65.36	69.59	67.42	69.77
sadness	64.21	67.31	65.02	68.21
surprise	54.09	58.10	56.33	60.34
Average	**59.36**	**64.98**	**62.17**	**65.30**

The evaluation results of the baseline and morphology based systems show that both the systems perform satisfactorily for the simple sentences containing single emotional expression. The error analysis suggests that not only the rich morphology but also the free phrase order nature of Bengali restricts the baseline system to capture the metaphoric presence of the emotions. The baseline system suffers in disambiguating the emotions for complex and compound sentences, as no full-fledged dependency parser is available in Bengali.

The emotion tagging system tags the emotional sentences of the bloggers' comments. All the sentence level emotions are assigned at paragraph level. It has been observed that different emotions are associated with different paragraphs or comment sections associated with different timestamps. Thus, we planned to analyze the emotional changes of a blogger with respect to time and emotions of different bloggers on a given topic at the same or different timestamps.

5 Generation of Emotion Graph

Ekman's six different emotions are tagged at the sentence level. By hypothesis, temporal relations also exist among the comments that are associated with ordered timestamps. The emotions between each of the timestamps are represented using a graphical format. The paragraph level emotions acquired from the user comments are represented based on the temporal relations provided by the timestamps. Considering each and individual bloggers' section as a separate paragraph, the sentential emotions are assigned at the paragraph level. As the comment sections with respect to a single blogger are ordered according to the associated timestamps, the emotions are also represented based on the timestamps.

The number of emotions present in the individual comment sections is used for emotional intensity. The X axis and the Y axis in the emotion graph represent the time and emotion intensity respectively. Six separate colors are used for showing six emotions in the generated graph (as shown in Figure 4). The manual evaluation of the generated emotion graphs for 20 bloggers shows that the system performs satisfactorily in tracking emotions with respect to time.

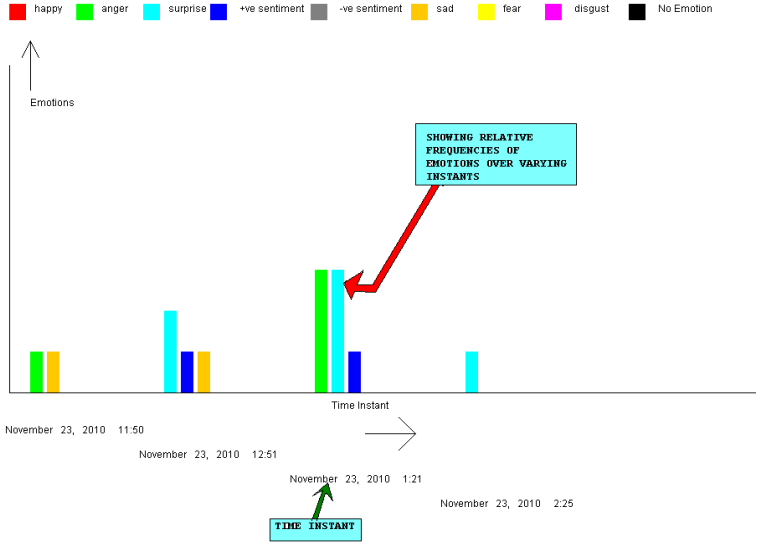


Fig. 4. Emotion Graph of a blogger with respect to the associated timestamps

6 Conclusion

In the present work, we have reported our work on identification and tracking of bloggers' emotions from Bengali blog documents. An emotion tagging system along with the lexical word level keyword spotting technique has been described in this work. From the overall analysis, it is observed that the approach can be applied for

other Indian languages. The metaphorical analysis concerning genre of the corpus to identify sentence level emotions are the future areas to be explored. More emotion annotated data is required to improve the performance of the system as well. Though the system performs satisfactorily in case of tracking emotions of the bloggers, the dependency among the bloggers' emotions is to be analyzed for capturing the reasons of emotional changes. The hypothesis of the present model will be used in future for developing a topic driven emotion tracking model.

References

1. Alm, C.O., Roth, D., Sproat, R.: Emotions from text: machine learning for text-based emotion prediction. In: Proceedings of the Conference on HLT (EMNP) Vancouver, British Columbia, Canada, pp. 579–586 (2005)
2. Carmen, B., Mihalcea, R., Wiebe, J., Hassan, S.: Multilingual subjectivity analysis using machine translation. In: Proceedings of EMNLP 2008 (2008)
3. Das, D., Bandyopadhyay, S.: Word to Sentence Level Emotion Tagging for Bengali Blogs. In: The Proceedings of Short Paper, ACL-IJCNLP 2009, Suntec, Singapore, pp. 149–152 (2009)
4. Das, D., Bandyopadhyay, S.: Labeling Emotion in Bengali Blog Corpus – A Fine Grained Tagging at Sentence Level. In: 8th Worshop on Asian Language Resources (ALR8), COLING 2010, Beijing, China, August 21-22, pp. 47–55 (2010)
5. Das, D., Bandyopadhyay, S.: Developing Bengali *WordNet Affect* for Analyzing Emotion. In: Proceedings of 23rd International Conference on the Computer Processing of Oriental Languages (ICCPOL 2010), California, USA, pp. 35–40 (2010)
6. Das, D., Kolya, A.K., Ekbal, A., Bandyopadhyay, S.: Temporal Analysis of Sentiment Events – A Visual Realization and Tracking. In: Gelbukh, A.F. (ed.) CICLing 2011, Part I. LNCS, vol. 6608, pp. 417–428. Springer, Heidelberg (2011)
7. Ekman, P.: Facial expression and emotion. American Psychologist 48(4), 384–392 (1993)
8. Fukuhara, T., Nakagawa, H., Nishida, T.: Understanding Sentiment of People from News Articles: Temporal Sentiment Analysis of Social Events. In: ICWSM 2007, Boulder, Colorado, USA (2007)
9. Grefenstette, G., Qu, Y., Shanahan, J.G., Evans, D.A.: Coupling niche browsers and affect analysis for an opinion mining application (2004)
10. Havre, S., Hetzler, E., Whitney, P., Nowell, L.: ThemeRiver: Visualizing Thematic Changes in Large Document Collections. IEEE Transactions on Visualization and Computer Graphics 8(1), 9–20 (2002)
11. Ku, L.-W., Liang, Y.-T., Chen, H.H.: Opinion extraction, summarization and tracking in news and blog corpora. In: AAAI 2006 Spring Symposium on Computational Approaches to Analyzing Weblogs, AAAI Technical Report, pp. 100–107 (2006)
12. Leopold, E., Kindermann, J.: Text categorization with support vector machines, How to represent texts in input space? Mach. Learn. 46(1-3), 423–444 (2002)
13. Lin, K.H.-Y., Yang, C., Chen, H.-H.: What Emotions News Articles Trigger in Their Readers? In: SIGIR, pp. 733–734 (2007)
14. Liu, B.: The challenge is still the accuracy of sentiment prediction and solving the associated problems. In: 5th Annual Text Analytics Summit (2009)
15. Liu, H., Lieberman, H., Selker, T.: A Model of Textual Affect Sensing using Real-World Knowledge. In: IUI 2003: Proceedings of the 8th International Conference on Intelligent User Interfaces. ACM (2003)

16. Rada, M., Banea, C., Wiebe, J.: Learning multilingual subjective language via cross-lingual projections. In: Proceedings of ACL 2007 (2007)
17. Mishne, G., de Rijke, M.: Capturing Global Mood Levels using Blog Posts. In: Proceedings of AAAI, Spring Symposium on Computational Approaches to Analysing Weblogs, pp. 145–152 (2006)
18. Quan, C., Ren, F.: Construction of a Blog Emotion Corpus for Chinese Emotional Expression Analysis. In: Empirical Method in Natural Language Processing- Association for Computational Linguistics, Singapore, pp. 1446–1454 (2009)
19. Quirk, R., Greenbaum, S., Leech, G., Svartvik, J.: A comprehensive Grammar of the English Language. Longman, New York (1985)
20. Salovey, P., Mayer, J.: Emotional Intelligence. Imagination, Cognition and Personality 9(3), 185–211 (1990)
21. Sood, S., Vasserman, L.: ESSE: Exploring Mood on the Web. In: Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media (ICWSM) Data Challenge Workshop (2009)
22. Strapparava, C., Valitutti, A.: Wordnet-affect: an affective extension of wordnet. In: 4th Language Resource and Evaluation Campaign, pp. 1083–1086 (2004)
23. Whitelaw, C., Garg, N., Argamon, S.: Using appraisal groups for sentiment analysis. In: CIKM 2005: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pp. 625–631. ACM, New York (2005)
24. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. Language Resources and Evaluation 1(2) (2005)
25. Wiebe, J., Wilson, T., Rebecca, F., Bell, M., Martin, M.: Learning Subjective Language. Computational Linguistics 30, 277–308 (2004)
26. Yang, C., Lin, K.H.-Y., Chen, H.-H.: Building Emotion Lexicon from Weblog Corpora. In: Proceedings of the 45th Annual Meeting of ACL, pp. 133–136 (2007)
27. Yang, C., Lin, K.H.Y., Chen, H.H.: Emotion classification Using Web Blog Corpora. In: IEEE, WIC, ACM International Conference on Web Intelligence, pp. 275–278 (2007)
28. Changhua, Y., Lin, K.H.-Y., Chen, H.-H.: Writer Meets Reader: Emotion Analysis of Social Media from both the Writer's and Reader's Perspectives. In: IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology Workshops, pp. 286–290 (2009)
29. Yi, J., Nasukawa, T., Bunescu, R., Niblack, W.: Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In: Proceedings of ICDM (2003)
30. Zhang, Y., Li, Z., Ren, F., Kuroiwa, S.: A Preliminary Research of Chinese Emotion Classification Model. IJCSNS International Journal of Computer Science and Network Security 8(11), 127–132 (2008)
31. http://www.amarblog.com/category/bloggor-bloggor/entertainment/jokes-entertainment-bloggor-bloggor (last accessed May 10, 2009)

Pseudo-code Programming of Designer Activities in Development of Software Intensive Systems

Petr Sosnin

Ulyanovsk State Technical University, Severny Venetc str. 32,
432027 Ulyanovsk, Russia
sosnin@ulstu.ru

Abstract. The paper presents a question-answer approach to programming of designer activities during collaborative designing of software intensive systems. Efficiency of a conceptual work can be essentially increased if the human part of the work will be fulfilled as an execution of a special kind of programs by "intellectual processors" which use models of question-answer reasoning. Such approach was investigated and evolved till an instrumental system providing the pseudo-code programming of intellectual processors combined with computer processors.

Keywords: Conceptual designing, precedent, pseudo-code programming, question-answering, reasoning, software intensive systems.

1 Introduction

The development of software intensive systems (SIS) is an activity the specificity of which is being reflected in a sufficient measure by the technology «Rational Unified Process» (RUP) [1]. The specificity includes the following features:

- collective activity of designers which fulfill different actions by playing corresponding roles (architect, system analyst, programmer and many others) in frames of definite scenarios;
- normative modeling of such activity in the form of workflows the typical units of which are tasks with guides for their repeated decisions;
- usage by designers several hundred of typical tasks (for example, in RUP about 500 units only for conceptual designing) the enormous quantity of examples of which should be decided collaboratively in the coordination;
- necessity to generate by designers the new ideas and to solve creatively new tasks (not only typical tasks) which evolve their personal and collective experience.

The list of specific features can be continued but the named of them are sufficient for explaining the author suggestions described below. First of all one can agree that activities of the designers connected with their work with enormous quantity of typical and creative tasks, should be automated.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 457–466, 2012.

Many approaches and instrumental means for the automation of such work are existed and used. For example in RUP the automation is being supported by the usage of the networking access to the Experience Factory the units of which (templates, guides and the others) are being applied in decisions of current tasks. Except of that the specialized toolkit is accessible on any workplace of designers group. But all of these means do not contain means of programming for the tasks being solved on a conceptual stage of designing.

The extremely low degree of success in the development of SIS (a little more 30%) indicates that new approaches and means are needed for the positive changing of the existed situation [2]. One of directions of the positive change can be connected with the creation and usage of means for programming the designer activity especially in the conceptual designing. By the other words, the degree of automation will be positively increased if the designer will play the role of the "processor" which executes the programs managing the designer activity.

In this paper such role of the designer is being named as an intellectual processor (I-processor) the essence of which is constructively defined below. Executing of this role is supported by a system WIQA (Working In Questions and Answers) which provides the collaborative execution of workflows by the group of I-processors and corresponding computer processors (K-processors) in the client-server medium [3].

2 Related Works

The idea of the designer model as I-processor is inherited by the author from a set of publications [4], [5] and [6] where described the model human processor (MH-processor) as an engineering model of the human performance in solving the different tasks in real time.

The especially known application of the MH-processor is Executive Process-Interactive Control (EPIC) described in detail in [6]. Means of EPIC support the programming of the human interaction with the computerized system in the specialized command language Keystrok Level Model (KLM). A set of basic KLM actions includes the following operators: K – key press and release (keyboard), P – point the mouse to an object on screen, B – button press or release (mouse), H – hand from keyboard to mouse or vice versa and others commands. MH-processor is defined [4] as a system of specialized processors which solve the common task collaboratively. One of these processors is a cognitive processor providing mental reasoning the type and content of which are not specified.

It is necessary to underline that I-processor is similar to MH-processor and includes the cognitive processor also. But the existence and work of this component of I-processor are revealed through reasoning of the question-answer type. This feature is one of the main differences between MH-processor and I-processor. Another important difference is a set of basic commands which includes typical commands of the pseudo-code algorithmic language. There are two ways for managing the designer activity the one of which is based on question-answer reasoning (QA-reasoning) which is registering by designers in WIQA and coming back to support the work of

I-processors. The second way is aimed at the creation of pseudo-code programs being executed by I-processors.

Moreover, interactions of I-processor with K-processor are being implemented in WIQA with the usage of a model of QA-reasoning as in its free form so in pseudo-code program form. QA-reasoning in all versions of their usage supports the interaction of designers with own experience and with its computer models.

3 Question-Answer Reasoning

The choice of QA-reasoning as the basic form for I-processor is determined by the intention to model the dialog nature of consciousness. For that the implicit QA-reasoning accompanying the cognitive processes inside I-processor should "be translated" and transferred to K-processor as an obvious QA-reasoning. Attempts to support the dialog nature of reasoning should help in the coordination of human-computer interactions in the collaborative work of I-processor and subordinated K-processor.

Ensuring of the named coordination implements by means of WIQA aimed at modeling and using of QA-reasoning in conceptual decisions of tasks in the development of SIS. The other useful interpretation of WIQA is a QA-processor which provides the collaborative work of I-processors and corresponding K-processors.

Combining of processors is schematically presented in Fig. 1 which is inherited and adapted from Fig. 1 of the ACM SIGCHI Curriculum for Human-Computer Interaction [7].

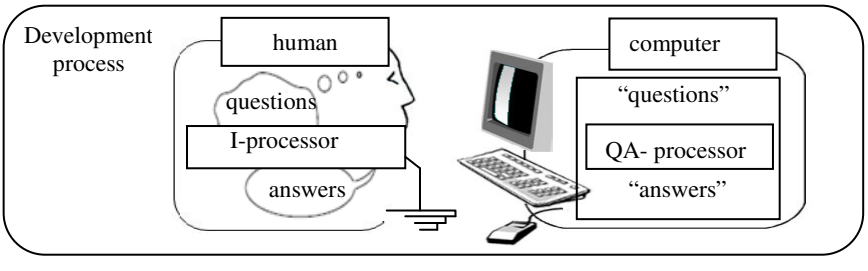


Fig. 1. General question-answer scheme of CHI

In scheme the question is understood by the author as the natural phenomenon which appears (in human brains) at the definite situation when the human interacts with the own experience. In this case the „question" is a symbolic (sign) model of the appropriate question. Used understanding helps to explain the necessity of fitting the „question" in QA-processes. Implicit questions and answers exist in the reality while „questions" and „answers" present them as sign models.

Any unit of QA-reasoning in WIQA presents the decision of the corresponding task and therefore such unit is implemented as the question-answer model (QA-model) of this task. In general case the task can include subtasks each of which

presents by the own QA-model. In current time of the development process all solved and solving tasks with their QA-models are registered in the server of WIQA and any component of the tasks tree or any QA-model is accessible to any designer at the corresponding client place. Such components as interactive objects are visually accessible (for I-processors) through the interface form presented in Fig. 2.

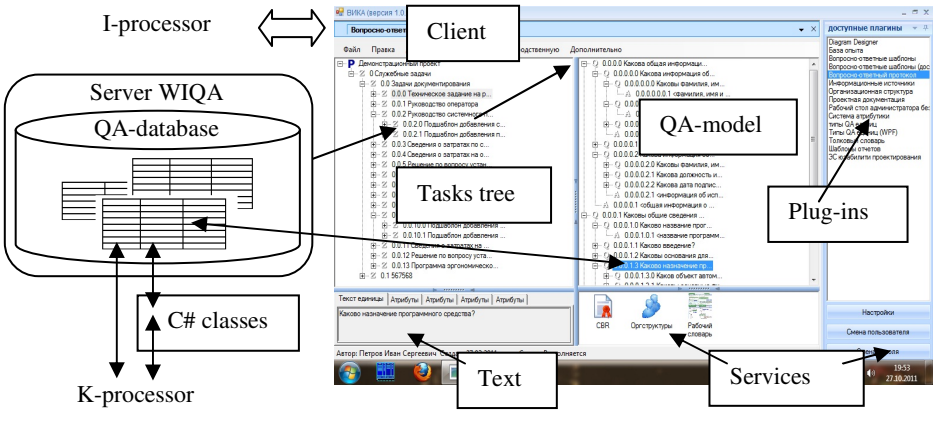


Fig. 2. Presentation of QA-reasoning

This form fulfils the role of an inter-mediator between I-processor and QA-processor. The interface language of WIQA is Russian therefore fields of the screenshot are marked by labels. Any K-processor has the access to the content of the tasks tree and QA-models through direct requests to QA-database or with the use of object-relational mapping (C# classes).

4 Precedents as Basic Form of Designer Activity

The intention to increase the success of the SIS development by the collaborative work of I-processors (used corresponding K-processors) should take into account the suitable presenting and modeling the units of the designer behavior. On a deep persuasion of the author the precedents should execute functions of such units.

Any activity is a naturally-artificial process created on the base of a definite set of precedents the samples of which are extracted from the appropriate experience and its models. Such role of precedents is explained by the following definition: “précédents are actions or decisions that have already happened in the past and which can be referred to and justified as an example that can be followed when the similar situation arises” [8]. There are causes for belief that models of precedents (as units of experience) are based on “natural programs” (N-programs) created on the natural language in its algorithmic usage (N-language).

One can notice that precedents are the natural form of the behavior for their reuse by the human. Such naturalness should be inherited in program models of precedents

which should be being created on an algorithmic language similar to N-language. In the author opinion the class of pseudo-code programming languages (P-languages) is a better class of algorithmic languages for creating of programs (P-programs) for design precedents (DP). In WIQA each precedent is connected with the decision of the definite task in the definite "place" of the tasks tree of the SIS designing.

The use of the precedent as a basic unit of the human interaction with own surrounding demands to choose or build adequate patterns for precedents representations. In describe case the appropriate patterns should provide the intellectual mastering of precedents and their natural using by designers in their collaborative work. The necessary model for the definite precedent (and DP also) can be created on the base of the following logical scheme (pattern):

c h o i c e

while [logical formulae (LF) for motives M= {M_k}]
as [LF for aims C = {C_l}]
if [LF for precondition U'= {U'_n}],
then [plan of reaction (program) r_q],
end so [LF for postconditions U" = {$U"_m$}]
---
there are alternatives {$P_j(r_p)$}.

This logical pattern is a human-oriented scheme the human interaction with which activates the internal logical process on the level of the second signal system in human brains.

But the designer interaction with this precedent model is only one type of actions which are useful for the usage of DP in designing. Therefore DP-models need to create on the base of the framework which integrates a number of useful patterns. Such framework investigated by author is presented in the Fig. 3 where the logical scheme plays the role of the framework kernel.

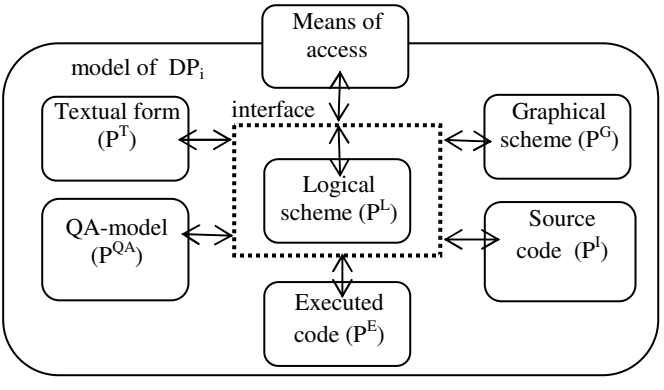


Fig. 3. Structure of precedent model

This integral model includes the following useful components of precedent models: P^T – textual description in the form of the statement of the task for the precedent

creation, P^L – adequate logical formula, P^G – graphical (diagrammatic) scheme(s), P^{QA} – question-answer model, P^I – source code and P^E – being executed code. All of these components are included to the typical materialization of the precedent sample in the knowledge base (precedents base) of WIQA.

The important part of the logical scheme is a reaction plan of the designer(s) behavior which should be coded in the precedent model for the future reuse. Very often the scheme of such behavior is presented as the plan of the reaction. If the plan includes conditions and-or cycles then, its text is better to write in the pseudo-code language similar to the natural language in its algorithmic use. In this case the reaction plan will have the form of P-program.

The difference of P-programs for I-processor from programs for K-processors and from programs for EPIC can be shown (as typical example) by the following instruction fragment provided the task analysis in its stepwise refinement:

```
P: "QA-analysis of task":
O1.Create the initial statement T(Z)of the arisen task Z.
O2. Include copy of the initial statement T(Z) to the
list L^T of texts waiting their analysis.
O3. Take the first text T from the list L^T for its analy-
sis
O4. Extract (and formulate)the next question Q from the
text T.
O5. Include the extracted question Q to the list L^Q of
questions.
O6. If "extraction is not finished" then goto "O3".
O7. Reorder list L^Q for rational formulating the answers
O8. Take the first question Q from the list L^Q
O9. Formulate answer A for taken question Q
. . . . . .
O14. Exclude text T from list L^T
O15. If "analysis is not finished" then goto "O3"
O16. End
```

This example indicates that the work is fulfilled by the human who uses computer assistants. Let's mark that the reaction plan in the form of P-program is being created often as a technique for solving the major task of the corresponding precedent.

The other important task in precedent reuse is connected with the search of the suitable model including its choice in a set of alternatives. Hence, a set of effective and handy means should be accessible for designers for writing and fulfilling P-programs supporting the work of designers with precedents models and their components.

Means of I-processor should support QA-interactions of the human with the precedent in its reuse process. The major part of such interactions consists of the execution of P-programs embedded to the current precedent sample. The main executor of P-programs is the designer who fulfills the role of I-processor.

5 Question-Answer Processor

The system WIQA has been developed previously as QA-processor for the conceptual designing of the SIS by the method of conceptual solving the project tasks [3]. This method is based on the stepwise refining and QA-reasoning which are being evolved in the frame of incremental designing.

Processor WIQA has been implemented in several versions. Elaborations of two last versions were based on architectural views of QA-model and the usage of the repository, MVC, client-server and interpreter architectural styles.

Moreover in created versions have been used object-oriented, component-oriented and service-oriented architectural paradigms.

One of the last versions named as NetWIQA has been programmed on Delphi 6.0 and the second version (named as WIQA.Net) has been created on C# at the platform of Microsoft.Net 3.5.

The structure of WIQA, its functional possibilities and positive effects are described in the publication [3]. The features of WIQA are reflected by its general components structure presented in Fig 4.

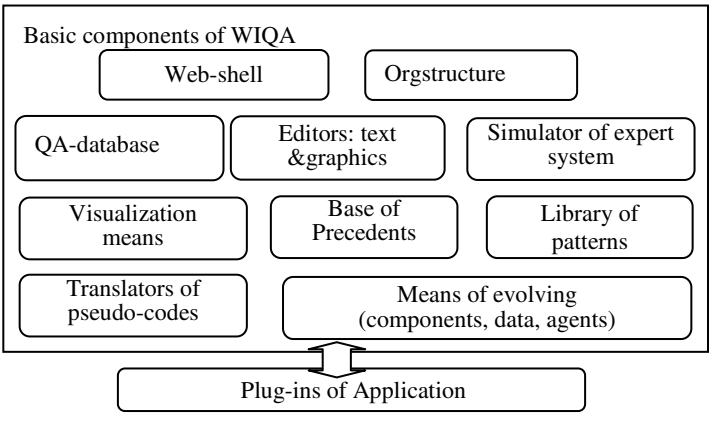


Fig. 4. Components structure of WIQA

One of the latest application helps to designers to create and use models of precedents with of embedded actions. Pseudo-code programming means are implemented as plug-ins which supports the writing of pseudo-codes in QA-model forms and their interpreting or compiling in interactions with DP-models.

6 QA-Means of Pseudo-code Programming

6.1 QA-Model of Data

Any P-program is useful for its understanding as an algorithmic description of interactions of the designer with the corresponding precedent. In WIQA the normative

way for interactions is QA-reasoning. Hence is better to adapt the means of QA-reasoning for their use in pseudo-code programming (P-programming). For such adaptation it is necessary to find the ways for emulations (with the help of QA-reasoning) data and operators of the appropriate language of P-programming. The first step of the realized emulation (in WIQA) is presented in Fig. 5.

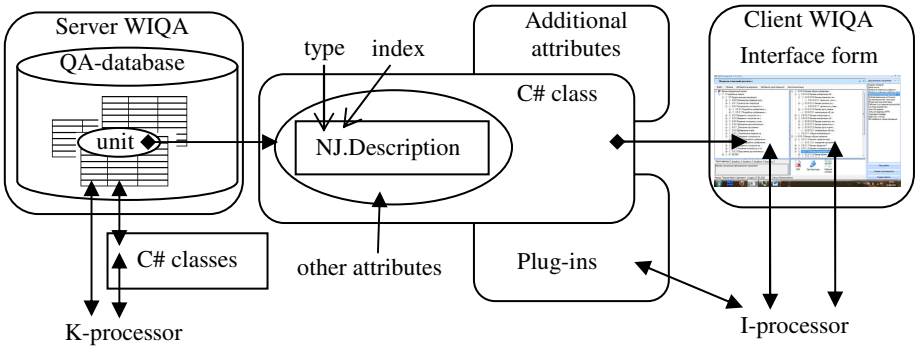


Fig. 5. Object-relation mapping of QA-database unit

This step includes the object relational mapping of necessary units which are chosen in QA-database for presentations of data and operators in P-programs. After mapping any line of source code of P-program s existed as an object (class) formed at C#. Such materialization of the line inherits all attributes of the appropriate unit ("task" Z, "question" Q, "answer" A or the unit of the others subtypes) in QA-database. Moreover, this set of attributes can be expanded by additional attributes {AAi} which can be appointed to the "line" with the help of WIQA plug-ins named "Additional Attributes". So except the field "description" for writing the line of the source code this line has a number of useful normative attributes (type of line "N", unique index name "NJ", owner of unit, time of last changing and the others) and additional attributes (for example, for the data unit its type and other corresponding characteristics). The usefulness will be increased if the presentation of any data or any operator of P-programs includes two connected parts which correspond to "question" and "answer". Objects with named features are defined in WIQA as QA-model of data (or shortly QA-data).

In order to underline that QA-data is used for writing the lines of the source code in P-programs the subtypes "D" , "O" and the corresponding subtype "V" (value) for data and operators were included to the set of types of QA-data. While D-lines and O-lines are used for descriptions the V-lines are used for registering the values for variables and facts of executions of operators.

D-lines include tags "&" which mark variables in textual descriptions explaining the variables meaning. Attributes of types for variables are being appointed as additional attributes. Descriptions in O-lines include special keywords (for example, "if", "then", "do", "while", "end" and the others) indicating the types of operators, names of variables, constants, symbols of operations and commentaries. There is a possibility to define and use "objects", "functions", "procedures" and to appoint synonyms for any keyword.

6.2 Medium of P-programming

Means for P-programming have been developed and embedded to WIQA as its evolution. The creation and use of P-programs are being fulfilled in the operational medium presented in Fig. 6.

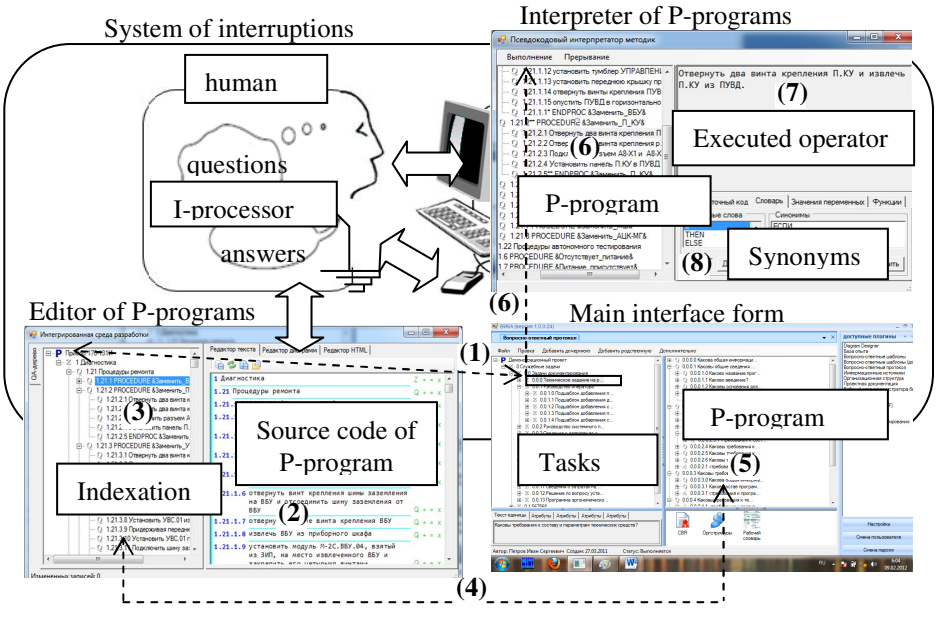


Fig. 6. Operational medium for P-programming

Creating of the definite P-program is beginning from the choice of the point in the tasks tree and the declaration of the new task for this program. The index name of this task (1) will be used as the initial address for computing the index names for any line of source code of P-program which is written in the area (2) of the text editor. The indexed copy of source code is registered in the editor memory and is visualized in the area (3). After saving the current state of the source code its indexed copy transfers (4) to the QA-database (5). In any time any P-program from the database can be loaded to the editor.

Any P-program in any its state can be load (6) to the interpreter for the execution. Any executed operator of P-program is visualized in the special area (7) of the interpreter and in any time the designer can declare the new synonym for the chosen keyword (8). There are other useful possibilities accessible for the designer in the main interface forms presented above in Fig. 3.

The reality of the designer activity is a parallel work with many tasks at the same time. Therefore the special system of interruptions is included into WIQA. It gives the possibility to interrupt any executed task or P-program (if it is necessary) for working with other tasks or P-programs. The interruption system supports the return to any interrupted task or P-program to its point of the interruption.

7 Conclusion

This paper represents new means of programming which are aimed at increasing the effectiveness of the designer activity. Such means based on QA-reasoning widely used in conceptual solutions of project tasks. Programming of the conceptual activity helps to increase the level of its automation and by that to reduce quantity of semantic faults and defects in designer solutions. The suggested means manage the designer who should fulfill the role of I-processor working under control of P-programs in decisions of typical project tasks.

The offered means are adapted at understanding of the designer activity as the work based on precedents. Therefore P-language is defined and built as the object-oriented language the potential of which is sufficient for the expression of the necessary semantics. P-programming is being applied as to the precedents already mastered by designer and for the precedents being created during designing.

Possibilities of P-programming are implemented in the instrumental system WIQA supported the work of designers with the precedent base in the corporate network. Means of P-programming include the interpreter of P-programs for their execution by designers and the compiler of P-programs for the computer execution. The offered means have confirmed the practical usefulness in the development of a number of SIS, including "Multi-agent system for simulation of surrounding the sea vessel" and "Question-answer expert system for ship collision avoidance". Means of P-programming was used for developments plug-ins "System of human-computer interruptions" and about 400 guides embedded to WIQA and also for the creation a set of P-programs for a number of specialized systems.

References

1. Kroll, P., Kruchten, P.: The Rational Unified Process Made Easy: A Practitioners Guide to the RUP. Addison-Wesley (2003)
2. Reports of the Standish Group, http://www.standishgroup.com
3. Sosnin, P.: Means of question-answer interaction for collaborative development activity. Advance in Human-Computer Interaction 2009, 18 (2009)
4. Card, S.K., Thomas, T.P., Newell, A.: The Psychology of Human-Computer Interaction. Lawrence Erbaum Associates, London (1983)
5. Karray, F., Alemzadeh, M., Saleh, J.A., Arab, M.N.: Human-Computer Interaction: Overview on State of the Art Smart sensing and intelligent systems, vol. 1(1), pp. 138–159 (2008)
6. Kieras, D., Meyer, D.E.: An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. Human-Computer Interaction 12, 391–438 (1997)
7. Hewett, T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., Verplank, W.: ACM SIGCHI Curricula for Human-Computer Interaction. ACM Technical Report, p. 162 (2002)
8. Precedent, http://dictionary.reference.com/browse/precedent

Intelligent Aspects of AIDA Programming

Yutaka Watanobe, Lin Gu, and Nikolay Mirenkov

University of Aizu, Aizu-wakamatsu, Fukushima 965-8580, Japan
yutaka@u-aizu.ac.jp, nikmir@u-aizu.ac.jp

Abstract. AIDA stands for animation and images to develop algorithms. It is a languge of algorithmic CyberFrames, Animation Scenes, and CyberFilms within the Filmification modeling (F-modeling) environment where pictures and moving pictures are used for the representation of features of computational algorithms and data structures. Generic pictures of the algorithmic super-characters are used to compose compound pictures defining algorithmic steps. The generic and compound pictures, as well as their series, are developed and acquired in special galleries of an open type where supportive pictures of embedded clarity annotations are also included. The acquisition is oriented to enhancing users perception, comprehension and cognition within works on developing application models, corresponding algorithms and programs. In this paper, a general scheme of the F-modeling environment on data/knowledge acquisition and examples of pictures from the galleries are presented. A new case of such acquisition related to program in pictures for the sparse matrix-vector multiplications is also presented and some technigues for creating embedded clarity annotations are explained.

Keywords: programming in pictures, Algorithmic CyberFilm, AIDA.

1 Introduction

Intelligence of software for computational engineering, as for other computational fields, is usually based on a collection of decision support technologies for application researchers and practitioners in creating reliable models and corresponding programs, and in performing relevant experiments and simulations. The aims of the technologies are to enable people for making faster and better decisions, as well as for producing robust solutions. Such technologies are embedded in languages of specification and modeling, in special supportive tools and unified environments, as well as in some systems of data/knowledge acquisition. Roughly, these languages, tools, and systems can be divided to two subsets: one with orientation to make decisions instead of users and another with orientation to help users in making their own decisions (see, for example, [1-5]). In fact, a majority of the technologies include features of both types with some prevalence of one of them.

One of ways to support the application programmers in their decisions is solutions enhancing their perception, cognition, and comprehension abilities. This

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 467–476, 2012.

is abilities to integrate, analyze, and act on large amounts of data from various sources at once [6]. Existing programming environments based on UML and other diagrams are still not easy with abstract, necessity to possess specific skills for putting together different views, and with understanding dynamical processes through static structures [7-8]. In addition, they are not so good in displaying large amounts of data to be useful for enhancing the above mentions abilities. Among examples of some successful enhancing, analogical representations of programs, a semiotic analysis of comics and self-assembling tiling can be mentioned [9-10]. A variety of papers where the concept of putting human needs first is applied and where a necessity for uniting scientific and artistic sides to obtain a balance between usefulness and attractiveness of technologies can also be pointed [11]. Another important thing of the software intelligence is related to forms of data/knowledge acquisitions within the corresponding environments and tools. In conventional programming environment such acquisitions are usually reduced to adding new items in libraries of components, classes and/or procedures. These items of black-box types are useful in general, but not so supportive in enhancing perception, cognition, and comprehension abilities of the users.

AIDA language and its F-modeling environment support pictures and moving pictures as super-characters for representing features of computational algorithms. Within this approach some "multidimensional data spaces" are traversed by "fronts of computation" and necessary operations are performed during these traversal processes. Generic pictures of the algorithmic super-characters are used to compose compound pictures (called Algorithmic CyberFrames) defining algorithmic steps. Compound pictures are assembled into special series of Algorithmic CyberScenes and an Algorithmic CyberFilm. AIDA stands for animation and images to develop algorithms and, as a Japanese word with meaning related to space/time, points to some closeness of forms representing models and algorithms. Research results and experiments with representing various types of algorithms in pictures, etc., which are a background of AIDA concepts, can be found in [12-16] and other papers. AIDA is a language of modeling, coding, and, in some part, documentation.

In this paper we consider some intelligent aspects of AIDA and its environment with a focus on data/knowledge acquisitions which have influence on users perception, comprehension, and cognition abilities. To present our methods of enhancing the abilities, we select a type of most difficult pictures which can be involved within such acquisitions. These pictures are related to a sparse matrix format and an algorithm of sparse matrix vector multiplication.

2 AIDA Programming: A Brief Overview

As we have already mentioned in the Introduction, AIDA language is a basis of the F-modeling environment where pictures and moving pictures are used as super-characters for more direct representation of features of computational algorithms and data structures. There are compound pictures to define algorithmic CyberFrames and generic pictures to define the contents of compound

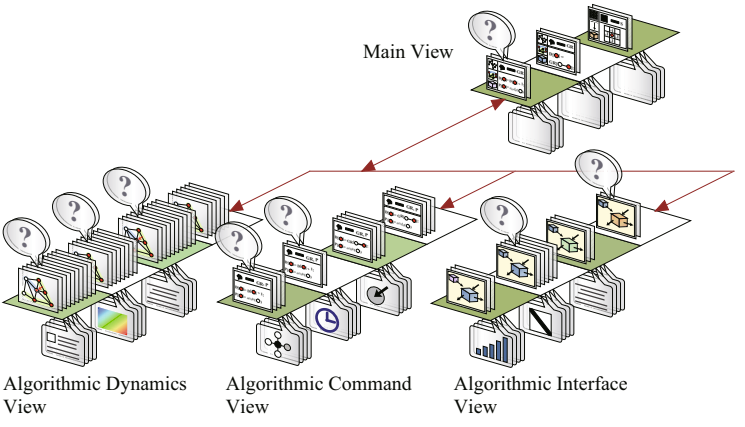


Fig. 1. A structure scheme of AIDA program format

pictures. Compound pictures are assembled into special series of Algorithmic CyberScenes and an Algorithmic CyberFilm. Fig.1 depicts a structural scheme of AIDA program format. The top-right series of frames represents Main (Integrated) view of an algorithm. Many super-characters involved in the view are intuitively understandable and do not require serious efforts for memorization. However, some of them are not so intuitively understandable. In addition, different people can have different levels of cognition and perception. To enhance the comprehension of the super-characters and their compositions, each picture of the Main view frames is supported by corresponding meaning annotation (in Fig. 1 pointed by a question mark) and/or by a series of frames of the bottom layer views. On the bottom layer there are Algorithmic Dynamic, Command and Interface views. They are also to clarify (if necessary) meaning of corresponding super-characters. Series of frames of Dynamics view clarify meaning of CyberScene super-characters by presenting animation of CyberScene skeletons (space data structures and traversal schemes of computational flows on these structures). Series of frames of Command view disclose some details related to application values of variables and formulas (actions) that are attached to the space-time points of the skeletons. Series of frames of Interface view represent extended vision of input/output operations that define the algorithmic interface with external world, as well as how a software component based on the algorithm should look from outside. It is important to note that each frame and/or a series of frames of the bottom layer views can have their own embedded clarity annotations. For example, a template program in C++, implementing computations presented by animation of Dynamic view, can be displayed as such annotation. Another example is special cover compound pictures which can be put before and/or after a series of CyberFrames to clarify their semantics as a whole.

In fact, the Main view is a compact (integrated) combination of algorithmic features presented by the bottom layer views. The language of this view, that is a set of super-characters and rules of their composition within CyberFrames, is

a very-high level language which, in a sense, can be used independently for programming and generating the executable code. All other views and annotations are involved only on demand. This means that after some practice with this language the users can define algorithms and perform numerical experiments without watching algorithmic skeleton animations and other view CyberFrames, and without reading various types of the embedded clarity annotations.

Within the F-modeling environment, the application users usually do not create frames, algorithmic skeletons and necessary template programs, but take and assemble them from galleries (libraries). For these users, the programming is reduced to defining sizes of the space structures, declaring variables on these structures, specifying operations on traversal nodes, including input/output operations, and attaching new annotations. An essential aspect of the environment is various opportunities for users to present important features of their application models, including features of possible input parameters and corresponding simulation algorithms. Very often, such features do not have a direct influence on the code generation, but very useful for readability and understandability of AIDA programs. Immediate access to these features simplifies visual debugging of the programs, modifications and maintenance of them, as well as inspections of external examiners.

3 Current Forms of Data/Knowledge Acquisition in F-Modeling Environment

Intelligent aspects of the F-modeling environment are based on a collection of the following decision support technologies:

- An open set of algorithmic super-characters and editors of CyberFrames
- CyberScenes and editors of CyberFrame series
- Template programs of CyberScenes
- Algorithms of automatic checking the compatibility of different views
- Code generation from CyberFilm specifications
- Galleries and clarity annotations
- Libraries of CyberScenes, CyberFilms, and template programs.

All these technologies are based on some types of data/knowledge acquisition being implemented through extensions of the galleries and ontology type libraries. Editors, checkers, and the code generator (all together) can be considered as Unified Knowledge Management system designed to take into account such extensions. The acquisition is not performed through independent adding a piece of information resources (a super-character, a CyberFrame or a CyberFilm) into a gallery or a library with consequent attaching links (associations) to other pieces, but through simultaneous adding a cluster of the pieces oriented to a few different galleries and libraries. This means that from very beginning intra-cluster links are designed to efficiently perform some predefined operations and tasks. In general, F-modeling environment is developed as Active Knowledge (Electronic) Book where users can quickly find necessary information resource, understand it

and immediately use (re-use) for new goals. Now the data/knowledge acquisition for this book is performed by the environment developers. End users can also make some contributions but only under special refereeing control.

AIDA type representations have been applied to a large variety of algorithms including algorithms on general graphs, trees, pyramids, particles-in-cells, as well as to sequential and parallel matrix multiplications, solving algebraic and partial differential equations, cellular automation-like algorithms, etc. This experience was used to create basic versions of the galleries and libraries. In this paper we consider a new algorithm to show applicability of our approach once more and to disclose how pictures are created for acquiring by our galleries and how AIDA can become more specialized for operations with sparse matrices.

4 Sparse Matrix-Vector Multiplication in the AIDA Integrated View

There are a number of formats for the sparse matrix representation [17]. In this case study we use the Yale format where three one-dimensional arrays are involved to define the sparse matrix content. Fig. 2 presents some explanation of this format and Fig. 3 shows an AIDA integrated view of sparse matrix-vector multiplication. In this program there is a declaration section and an algorithmic section. The declaration section consists of two top rows declaring five 1-D space structures and one 0-D space structure of the algorithmic activities. In addition, on each 1-D structure an integer variable is declared (A, C, R, V, M on structures st1, st2, st3, st4, and st5, respectively) and a size of the corresponding structure is pointed. The 0-D structure of name st0 is defined for some activity, but no variables are associated with it.

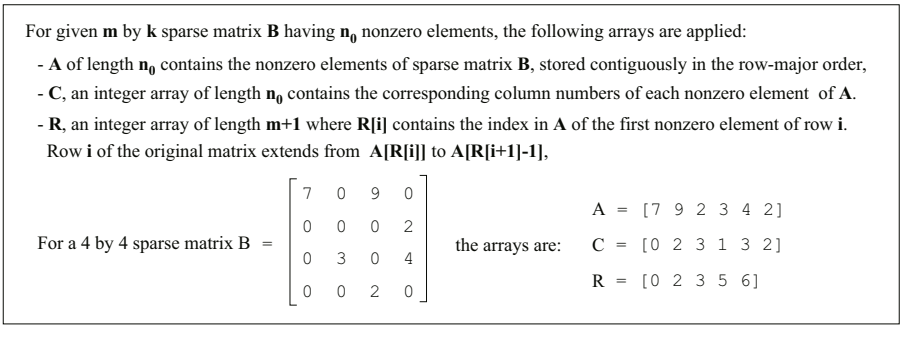


Fig. 2. A sparse matrix format

The algorithmic section consists of seven rows representing three subsections for the input of initial data, the sparse matrix-vector multiplication, and the output of the result. The input subsection includes two scenes represented by two icons (and three rows). The first icon represents three types of activity on space structures st1, st2, and st3. Each activity is defined by operations presented

in corresponding rows to right of the icon. They are the inputs of initial values for variables A, C, and R. Before each input there is a micro-icon of a triple-circle which points places of algorithmic activity in the space structures and informs that operations of each activity are defined as collective ones where all nodes of the same color are involved as a whole for each operation. For example, the first operation defines the collective input of A for all nodes of the st1 structure from a txt-file. The input operation has two attributes to show a source of the getting data (in this case, it is a file micro-icon) and a specific name (ID) of the data item (in this case, it is A.txt). In a case of applying one-circle micro-icons, operations of the activity are defined as individual ones where each node of the same color is involved independently (performs the same operation in parallel). In a similar way, the second icon represents one type of activity on space structure st4. This is to input values of V from a V.txt. To see dynamical features (of the algorithm) which are behind the scene icons, that is space structures and flashing nodes of activity within the structures, the corresponding scene animation can be called.

The subsection of the sparse matrix-vector multiplication is presented by one scene icon requiring the specification of three operations. Two operations (on half-flashing nodes) are related to decision making operations about which nodes have to be involved at the next step of computation and which type of involvement is expected. The first half-flashing operation assigns a node for data reading (for contour-flashing activity) in the st2 structure if the number (column) of the node is greater than the value of R on the left contour-flashing node in the st3 structure and if this number is also less or equal to the value of R on the right contour-flashing node in the st2 structure. The second half-flashing operation assigns a node for data reading in the st4 structure if the number (row) of the node is equal to the value of C on a contour-flashing node of the st2 structure. The third operation to be specified for the scene icon is the dot product, the sum of multiplications of appropriate pairs, of A and V on the contour-flashing nodes in structures st1 and st4. The micro-arrows under the SUM sign show that the nodes for the pairs from the structure st1 are selected in the left-to-right style and the nodes from the st4 structure are selected in the top-to-down style. The result of the dot product is assigned to variable M on full-flashing nodes of structure st5. To see how the space structures are involved and where the operations have to be specified, the scene CyberFrames can be displayed (on the same or on the second monitor) as a set of tiled pictures of Fig. 4 or as corresponding animation.

The output subsection includes one scene represented by a corresponding icon. This icon requires one type of activity to be specified on nodes of space structure st5. This specification is done in the corresponding row after a micro-icon of the triple-circle to right of the icon. This is to output the results of computation (variable M) to a monitor by a collective operation on nodes of structure st5. In general, the format of input/output operations has left and right sides. The left side is for defining a receiver (target) of data and the right side is to define sender (source) of data. An arrow frame pointing to the left and embracing this type of operations is to support the operation perception. Generic pictures (icons)

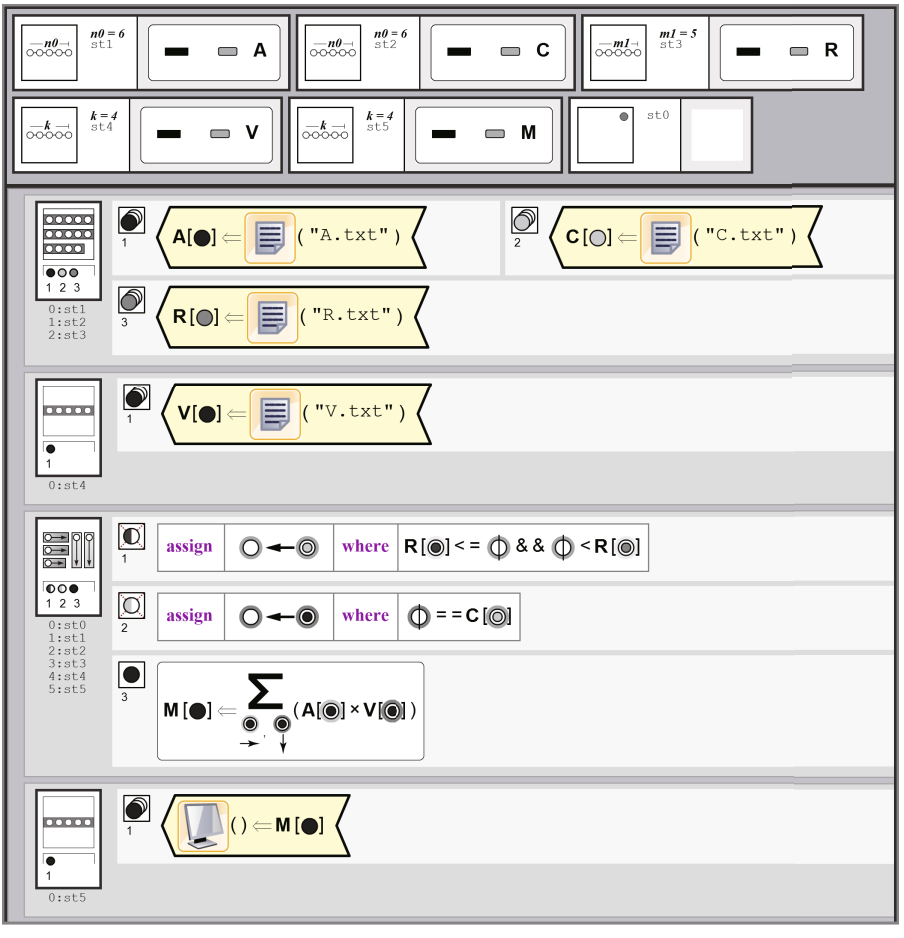


Fig. 3. AIDA Integrated view of sparse matrix-vector multiplication

of super-characters in the Integrated view are supported by special annotations explaining their meaning (as it is provided by this section text) and, for scene icons, by animation (as it is shown in the next section).

5 Sparse Matrix-Vector Multiplication in the AIDA Dynamic View

The Algorithmic Dynamic view is a series of CyberFrames representing dynamical features of computation behind the scene icons of the Integrated view. For each icon of the input/output subsections, such features are presented by one CyberFrame where all nodes of a 1-D space structure are flashed. So, the most important part of the skeleton view is the dynamics related to the third icon representing the matrix-vector multiplication without input/output and depicted by

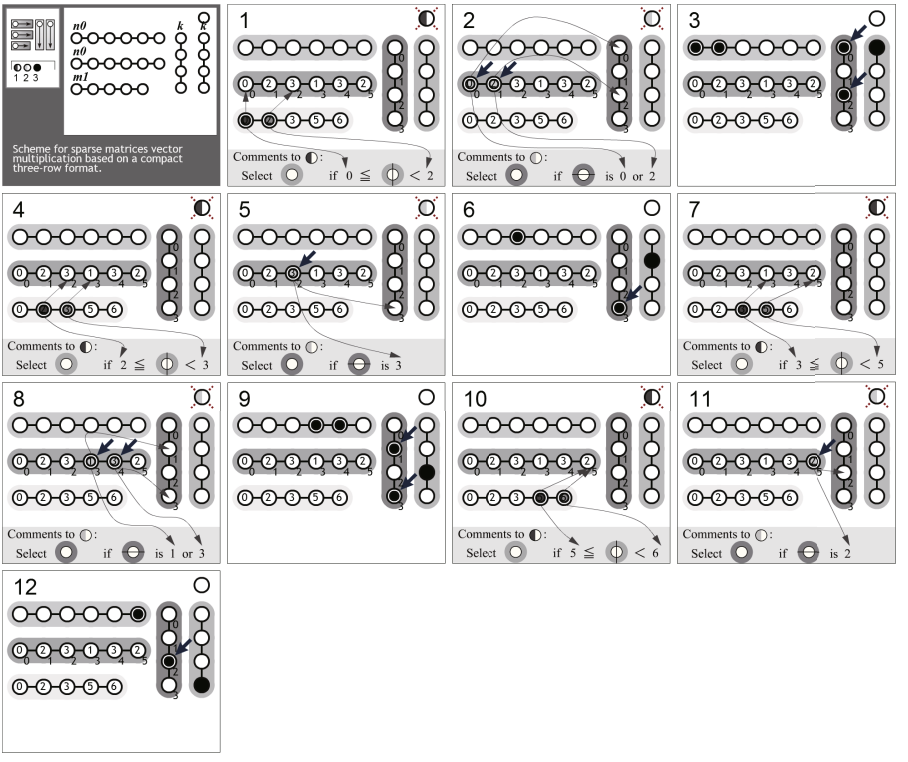


Fig. 4. AIDA Dynamic view of sparse matrix-vector multiplication

Fig. 4. In Fig. 4 the series of CyberFrames is started with a cover CyberFrame representing the scene icon, space structures involved and parameters related to their sizes, as well as some text explanation of why they are involved. The cover CyberFrame does not have any influence on the code generation, but is used to simplify understanding the other CyberFrames and to check compatibility of the space structure mentioning here and in the integrated view. It can also be supported by auxiliary frames of embedded clarity annotation. For example, it can be a text-based explanation of sparse matrix format presented by Fig.2. After this cover frame there are twelve steps of computation represented by twelve CyberFrames. The CyberFrames show a skeleton of the algorithm by displaying three types of activity which should be defined on the space structures (appropriate flashings are applied if the tiles are observed as animation).

The first type is represented by the half-flashing nodes of the st0 structure; a corresponding operation has to define places and types of activity at the next frame. Some arrows at the next frame show them and are interpreted as assume that these are decisions of the half-flashing operation from the previous frame. CyberFrames 1,2,4,5,7,8,10, and 11 are related to the first type of activity. The second type is represented by the counter flashing nodes pointing space positions where data are available for reading. For example, in the first

CyberFrame the half-flashing is related to the decision operation based on data from two left nodes of structure st3. All CyberFrames have nodes related to data for reading, however, in frames 2,5,8, and 11 they are not attractively depicted because of the gray-white format of the paper. The third type is represented by the full flashing nodes where data are available on reading and writing. CyberFrames 3, 6, 9, and 12 are related to this type of activity performed on nodes of structure st5. For example, in CyberFrame 12 this type of activity is defined for the bottom node of structure st5 where corresponding operation can read data from contour-flashing nodes of structure st1 and st4. The Dynamic view is some explanation of the computational scheme (the traversal of the space structures) based on a limited number of nodes in the structures. To enhance such explanation, the CyberFrames have some background/foreground images to clarify computational steps based on activity of the half-flashing type. These images include some comments to operations on decision making, including illustrative data at nodes and in nodes, as well as special arrows showing relations between the illustrative data and conditions for selecting the next frame activity. The illustrative numbers inside nodes are to make reference to possible values of variables on these nodes and the illustrative number at nodes are to make references to possible positions of the node involved. Usually, such images do not have influence on the code generation, but are important for understanding algorithms. This technique is also used for CyberFrames of the Integrated and other views. In addition to the background/foreground images which are explicitly integrated into the CyberFrames, there are additional images of embedded clarity annotations which are called on demand. The editors of the CyberFilm programming environment provide necessary support to users for attaching the explicit and implicit annotations.

6 Conclusion

AIDA programming is a process of assembling a set of visual super-characters into series of compound pictures to represent different algorithmic features. In this paper we have shown how generic and compound pictures can be created to be included into F-modeling environment galleries and how they should be supported by embedded clarity annotations and the algorithmic dynamic view. A scene icon of sparse matrix-vector multiplication has been introduced and the integrated and dynamic views as series of compound pictures have been created. A template program implementing a traversal scheme on nodes of sparse matrix structures has been added into the corresponding library. To enhance users perception of the pictures and views, 1) a cover frame (supported itself by an embedded clarity annotation) for the dynamic view as a whole and 2) special comments images and illustrative data at and in structure nodes on independent frames have also been created. The annotations and comments introduced don't have influence on the code generation, but can be really supportive in understanding meaning of picture based algorithms including operations with sparse matrices.

Acknowledgements. We are grateful to JNES (Japan Nuclear Energy Safety organization) for financial support of the AIDA project and for participation in analysis of its results.

References

1. Engelbrech, A.P.: Computational Intelligence: An Introduction. Wiley (2002)
2. Chaudhuri, S., Dayal, U., Narasayya, V.: An Overview of Business Intelligence Technology. Communications of the ACM 54(8), 88–98 (2011)
3. Bernstein, P.A., Haas, L.M.: Information integration in the enterprise. Communications of ACM 51(9), 72–79 (2008)
4. Chaudhuri, S., Dayal, U., Ganti, V.: Database technology for decision support systems. IEEE Computer 34(12), 48–55 (2001)
5. Sugawara, K., Fujita, H.: On Knowledge Management System for Assisting User's Decision in Office Work. In: New Trends in Software Methodologies, Tools and Techniques, pp. 159–165. IOS Press (2011)
6. Modha, D.S., Ananthanarayanan, R., Esser, S.K., Ndirango, A., Sherbondy, A.J., Singh, R.: Cognitive Computing. Communications of the ACM 54(8), 62–71 (2011)
7. Parnas, D.: The Risks of Stopping Too Soon. Communications of the ACM 54(6), 31–33 (2011)
8. Booch, G.: Draw me a picture. IEEE Software 28(1), 6–7 (2011)
9. Kindborg, M., McGee, K.: Visual programming with analogical representations: Inspirations from a semiotic analysis of comics. Journal of Visual Languages and Computing 18(2), 99–125 (2007)
10. Beckers, F.: Pictures worth a thousand tiles, a geometrical programming language for self-assembly. Theoretical Computer Science 410(16), 1495–1515 (2009)
11. Shneiderman, B.: Leonardo's laptop: human needs and the new computing technologies. MIT Press (2002)
12. Yoshioka, R., Mirenkov, N.: Visual Computing within Environment of Self-explanatory Components. Soft Computing Journal 7(1), 20–32 (2002)
13. Mirenkov, N., Vazhenin, A., Yoshioka, R., Ebihara, T., Hirotomi, T., Mirenkova, T.: Self-explanatory components: a new programming paradigm. International Journal of Software Engineering and Knowledge Engineering 11(1), 5–36 (2001)
14. Watanobe, Y., Mirenkov, N., Yoshioka, R.: Algorithm Library based on Algorithmic CyberFilms. Journal of Knowledge-Based Systems 22(3), 195–208 (2009)
15. Watanobe, Y., Mirenkov, N., Yoshioka, R., Monakhov, O.: Filmification of methods: A visual language for graph algorithms. Journal of Visual Languages and Computing 19(1), 123–150 (2008)
16. Watanobe, Y., Yoshioka, R., Mirenkov, N.: Cognitive Aspects of Programming in Pictures. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 11–20. Springer, Heidelberg (2011)
17. Sparse matrix, `http://en.wikipedia.org/wiki/Sparse_matrix`

Improving Healthcare Using Cognitive Computing Based Software: An Application in Emergency Situation

Roberto Revetria[1], Alessandro Catania[1], Lucia Cassettari[1], Guido Guizzi[2], Elpidio Romano[2], Teresa Murino[2], Giovanni Improta[3], and Hamido Fujita[4]

[1] Dipartimento di Ingegneria Meccanica, University og Genoa, Genoa, Italy
{roberto.revetria,alessandro.catania,cassettari}@unige.it
[2] Dipartimento di Ingegneria dei Materiali e della Produzione,
Univertisty of Naples "Federico II", Naples, Italy
{g.guizzi,elromano,murino}@unina.it
[3] Dipartimento di Scienze Mediche Preventive, University of Naples "Federico II",
Naples, Italy
giovanni.improta@unina.it
[4] Iwate Prefectuarl University, Intelligent Software Systems Laboratory, Takizawa,
Iwate-gun, Iwate, Japan
HFujita-799@acm.org

Abstract. The goal of this paper is to define the platform specifications dealing with medical information sharing both from research viewpoint and in terms of local health care. The purpose of this research work is based on doctor - patient relationships: VDS - Virtual Medical Doctor System. At this stage the platform is only used for scientific purposes.

In particular, we assume the integrated design of the platform based on different levels (layers), which may be interconnected through the information flows (or links). The first level (the lower) involves the construction of a VDS platform. Based on this system it is possible to foresee a number of extensions such as social network for scientific research design, and risk analysis tool.

Keywords: Doctor-patient relationship, VDS, Virtual Medical Doctor System, Human user interaction, Action decision model.

1 Introduction

The present work is focused on designing and developing a decision support system tool able to analyse data collected from different sources. One of this application is developed in medical field with VDS (Virtual Doctor System) developed in medical field applications. Starting our study from the existing data source and discover other data sources in order to make a more complete and exhaustive analysis. One main point is the interaction with medical Doctors to extract the knowledge base and criteria to be implemented on the system. Since the heterogeneity of collected data a creation of different separated Data Mart including specific data for each expected analysis typology is built. At the end a part of this system will be implemented in an Open Source Software called Pentaho used as a Proof of Concept of this study.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 477–490, 2012.

In order to achieve this goal the following steps are carried out based on these essential steps:

- Requirements analysis and collection;
- Data sources identification;
- System design;
- Proof-of-concept implementation.

2 Research Conceptual Foundations and Methodology

Italy and Japan have two very different cultures, however, they have many similarities in the way they consider elderly people. Japan and Italy share the same strong relationships among family members, both have great respect for old parents and very often elderly are going to pass many of the last years of their life under the direct care of their beloved ones. This situation is resulting in needs of more domicile care for the elderly and a distributed assistance to be performed by Medical Doctors (MD). These MD are often generalist and very few ones have received special training for geriatric treatments. When one of their elder patients requires special assistance (i.e. urology) they send him to a clinic or to a specialist resulting in an extra cost for care and in a waste of time as well as in a source of stress for the patient.

Vast majority of the patients requires more health monitoring and screening especially in particular situation (i.e. summer time, flu pandemic exploits, etc.) where they are more vulnerable. In exceptional situations such as natural disasters (i.e. earthquakes, floods, etc.) elderly people needs to be located, assisted and supported with a special care since they're often limited in mobility and because of their critical health conditions. More the increase of the average age of the population pushes the cost of the healthcare assistance higher and higher causing politics to seriously consider more budgets cuts as only solution to a rapidly increasing of welfare costs with MD and hospital struggling to guarantee an average level of service to the elderly. In a few words, today, elderly people are only seen, by politics, as an increasing source of problems resulting in a bad feeling for younger generation. Japan share with Italy the same awareness of the problem, both they have seen their population to became older and welfare costs are continuously increasing the National Debt (225% of the GP for Japan and 118% for Italy) and both known that this situation may only became worst: more elderly, more needs of assistance, more money for healthcare, less money for investments, less opportunity to grow the economy, less feeling for a better future, less money to have children to young couples resulting in an increase of the average age of the population: a vicious loop.

This loop could be interrupted by increasing the level of assistance to the elderly reducing at the same time the cost for it providing a new technology able to act simultaneously in three directions:

1. Increase the screening and monitoring level of the health conditions of the elderly people, allowing early warning on possible pathologies (i.e. prostatic cancer, Alzheimer, hypertension, diabetes, etc.) .

2. Provide direct assistance to elderly people (diagnosis, psychological support, treatment monitoring.
3. Provide support in case of natural disaster by providing for most vulnerable patients: identification of their health conditions, clinical and psychological support as well a immediate point of contact with a MD that make them feeling not abandoned.

This paper is focused on the development of a possible technology able to provide the above-mentioned actions.

The methodology decision support tool is based on doctor - patient relationships that is very well expressed in the Japanese study of Virtual Medical Doctor System. In particular, we assume the integrated design of the platform based on different levels (layers), which may be interconnected through the information flows (or links) as presented in the following figure 1.

- The first level (low) involves the construction of a VDS platform that will act as the core of the entire technology.
- VDS will have an avatar base interface (a) able to create a realistic representation of the patient both form the physical and the psychological point of view (b), the avatar based interface will be responsible to create an empathy (d) among the patient and the MD (real or simulated one) necessary to guarantee the perfect modelling of the real patient health condition.

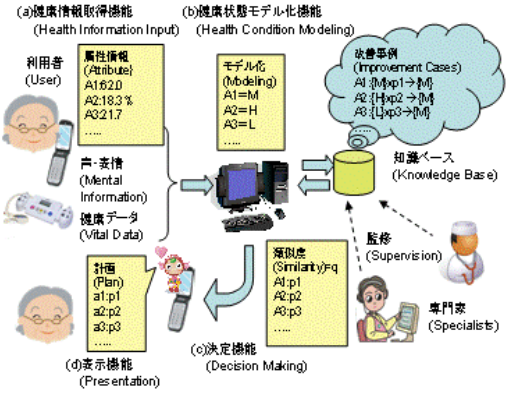


Fig. 1. Logical Scheme of the platform

Japanese VDS is designed to provide two main ontologies: physiological (vital data) and psychological (mental data) representation of the patient, Italian team will ad a third one specifically designed to represent the environmental condition of the patient living situation (i.e. pollution, radiation, temperature, humidity, noise, etc.). This systems will allow the possibility to remotely monitor the conditions of the patient from a single, and even a distributed, point of control. Real Medical Doctors (MDs) will monitor initially the conditions of the patients and, using the mental cloning methodology, will be possible to create a complex decision support system (DSS) able to support MDs to address patients correctly. The main purpose of this project will be

not to substitute MDs but to help them to deal with more patients more effectively providing first level screening and a single point of contact with their patients. A patient will have its well known and trusted avatar as representation of his MD and beyond him several MDs and specialist will monitor him reducing the stress of visiting several MDs when a specialist will be necessary. This system has also a self improvement and a learning loop since the knowledge base will be constantly improved by adding more and more medical records and more and more cases. This huge amount of medical records will be also available for researcher since various patients will be correlated with their physiological and environmental living condition resulting in a complex Virtual Clinic with thousand of patients under monitor. Based on this system is possible to foresee a number of extensions such as social network for scientific research designing, a platform for investigating possible pandemic scenario or simply a coordination point for delivery healthcare remotely in case on natural disaster.

This point is very well addressed considering its technological design: based on Cloud Computing and designed to take advantages from portable devices (i.e. Android OS), this application will have great potential in case of a natural disaster. Within few hours from the event is possible to re-establish 3G networks allowing elder patients to connect back to the VDS. A scared, and potentially injured patient, could connect back to its MD avatar, communicating his vital, mental and environmental parameters and receiving first aid instruction, psychological support (i.e. I am not alone, some one is taking care about me, etc.) and inform relatives about his current conditions. At the same time MDs, supervising the VDS, from a safe place (and potentially distributed) could be addressed on the more critical patients leaving general support to the non critical ones directly to the Knowledge Based Artificial Intelligence.

3 The VDS and Related Technologies

The VDS system is designed to work together with the corresponding human medical doctor in comprehensive coherency using the VDS before to outpatient diagnose and after to classify these diagnosis into classes. So called Simple cases classes could be addressed directly by VDS that would take conclusion and set the diagnosis procedure and appropriate action (e.g., issue drugs to the patient) supervised by the MD using a set of reports. More Complicated cases will require to have the MD to participate in the final decision. In such cases, the system sends the diagnosis reports to the MD and provides an appointment to the patient in the hospital queue. The system reads the queue data at the management centre of the hospital reception assigning the patient to the MD queue, if the Doctor later found that the assignment was appropriate (check mark OK) then the system learned that the decision is appropriate, however, by certain feedback from the doctor the system can learn from the doctor's feedback. The system will provide a window at the doctor office for performance evaluation in order to enforce the learning procedure for the system and, at the same time, such evaluation would provide an effective learning mechanism to increase the reasoning procedure for the diagnosis. Another important issue will be related to the MD Profiling.

Using MD Profiling it will be possible to create different specific representation of the various MD, as in real life. Since knowledge management would be based (i.e., mimic) on a specific Medical Doctor (i.e. Dr. John Doe), therefore, it would be stored in a specific Ontology Management System. So when another MD will doing the outpatient diagnosis (i.e. Dr. Jack Smith) then his profile of decision making related to that Doctor would be used (i.e., recalled). In this way will be possible to accurately mimic and categorize various physicians' actual practices. The VDS System would to simulate patient - doctor interactions using a virtual face (avatar) of an actual MD with pre- assigned virtual version of that medical doctor able to communicate and interact with the patient. The actual facial real-time created images of the MD will be synchronized with a spoken language in the same style of the actual physical doctor is created. The style mimics the actual doctor emotional expression as well his/her diagnosis style. Also the MD speaks in natural accent with emotions based on the patient mental mode, estimated by the patient profile (age, gender, ego data), and his/her situation automatically measured by data resembles (blood pressure, body weight, body temperature, and thermal analyzer), at the same time several environmental variable will be collected in order to create a broader view of the patient living situation. These devices are assembled to the patient desk chair where it would sit on, allowing measurements to be collected and transferred through network connection to the virtual doctor system.

Of course the diagnosis and treatment done by the VDS will be strongly based on the actual diagnosis and guidelines specified by real doctor and validated on based scenarios collected in advance. Such specification, which will be specifically made under Medical supervision, will be implemented in the VDS using specific Syntax Languages that is an open standard for representation of medical knowledge. In such ways the MD diagnosis guidelines are represented as a collection of medical logic modules (MLMs): each MLM will represents a single decision that is grouped into three categories: maintenance, library, and knowledge used by the inference engine to better mimic the MD attitude and diagnosis.

4 System Description

4.1 Analysis and Requirement Collection

Requirement Analysis play a fundamental role developing the system and it will characterize the architecture and data organization. We will conduct a careful system requirement analysis through existent documentation of the VDS and a series of interview at doctors and/or biologists to better understand analysis parameter and expected results. We will also conduct a study in order to gather, to organize, and to analyse parameters from the environmental context that VDS will provide us. The output of this phase will be a document specification used as guideline in order to develop this work. Requirements play an important role so we must understand and use each domain application specific term.

4.2 Data Source Identification

During this phase is essential to know where data are stored and how can be used inside the DW. Different data can be extracted from different source with different technology and different representation model. Each data knowledge of stored data. Fundamental principle of data warehousing is the concept of integrated data that allow us to transform general data into end user useful information. This result will be obtained through reconciliation process that consist of integration, cleaning and transforming all available data in a consistent manner.

4.3 System Design

Design phase will be divided into different steps:

- First phase we design the backend, that consists of ETL (Extract Trans for Load) modules.
- In the second step we conceptually design the DW, data organization and structure data into the DW. Main study of this part will be dimension analysis and attribute.
- In the third step we have to logically design the DW, to choose the Star or Snowflake structure and the dynamicity of each dimension in a proper manner.
- Last step is the system design analysis from OLAP, through Mining and Reporting depending on requirements.

4.4 Proof-of-Concept Implementation

At the end of this research study, we will use all the specifications gathered during requirements collection to make some real cases analysis hypothesis; this is useful in order to provide wide and complete view of system functionalities and to obtain the Proof of Concept.

5 Design of a Decision Model Tool

The first step of research activity will concern the VDS network and validation of its operations through the implementation and testing in medical care areas, at the public/private laboratories. Patients and MDs specific behaviours as well as with some critical situations, related to the definition of patient's diagnosis parameters, will be tested in both countries: Japan and Italy pointing out similarities and differences.

Following the VDS Validation and Verification phase against real-life situations, VDS will be used to train doctors, using the platform as a simulation environment, providing Physiological, Psychological and Environmental data of real and simulated patients to the MDs. The VDS will be validated with reference to real case studies in an experimental campaign specific for each country; it will be tested to assess the impact on various performance measurement such as: reduction of waiting times in the laboratory, variation of user number in queue, variation of service's level and readiness of medical answers for some diseases like flu, bronchitis and other diseases

that do not require specialist visits. In addition it will be tested the possibility of using the system to route through specialist if the investigated parameters are not aligned with those characteristic of normal situations, or can not be restored to normality by simple medical knowledge.

A further development is related to the design of platform extensions in order to enable:

1. Collection and acquirement of new knowledge about new therapies or new diseases.
2. Collection and acquirement of information from the field about symptoms related diseases diagnosed.
3. Collection and acquirement of new knowledge of possible side effects about therapy
4. Remote monitoring about patients' current health condition in case of an emergency or a natural disaster, case classification, first-aid support.
5. Simulation on possible effects of pandemic infections (i.e. SARS, A1-N1, etc.) on most fragile population.

This research work will extend the VDS original Ontology by adding more information regarding the environmental variables. In this way the VDS will be enabled also to increase the number of parameters used by VDS to perform processing and to monitor the phenomena identifying the possible environmental hazards (i.e. pollution, radiation, etc.).

In the first case we might expect to collect in the layers the geospatial information that will enable the VDS to expand its knowledge correlating this information in the form of cause/effect diagrams as well as performing regressive analysis. In the second case, depending on the symptoms experienced by the patient or from the diagnosis made in a specific period in a certain area, it will be possible to aggregate this information and make analysis of correlations (with events that occurred) to study phenomena and, potentially, to forecast the possible evolution. The design of these layers will necessarily assume the use of an appropriate simulation approach enabling the possibilities of a multi-scenarios analysis.

It's clear that these applications will affect the design of platforms for sharing the research ideas, like social networks, which may share information through blog and forums. The platform may share experiences of different national and international scientific communities in the field Medical creating a network with existing platforms. The scientific documentations, such as papers published in various scientific panel, diagnostic studies and data might be contained in the cloud environment, shared and accessible to researchers in order to enhance the quality of the DSS's internal inference engine. Beside the core, made by VDS, this project would like to define a platform where researchers may have access, upon registration, with an ability to modify and to upload the results of the latest innovations in medical science and, at the same time, where other users, such as operators and local medical officials may have easy access to check for updates and various international experiences linking to the results obtained using the VDS as a "Virtual Clinic" in order to help MD to identify best experiences and/or drugs to cure and to prevent certain forms of disease.

6 Case Study

Since the Italian Health Care expenses are always constantly increasing this implies the necessity that such expenses must be controlled. It means that having a model tool able to help in managing for a great amount of these healthcare expenses is very important. In addition, considering the great number of elderly people in Italy as regards the whole Italian population our study focus on build a model at low cost able to manage from home all patients having pathologies where hospitalisation isn't required , but such patients can be managed in "remote" manner from a system-model able to satisfy the user and to solve associated problems.

This users-patients target, as it will be analysed, in normal conditions enters in health care emergency system trying to receive answers, but instead always in the critical patients management path are included, implementing a resources waste of healthcare system.

The following data are showed depending on patients age target, kind of 118 emergency calls system, and so forth are referred to Liguria Region regards 2008 year (only 2010 for Savona province). To implement this analysed model on an area such Liguria region is very significant since the high percentage of elderly people living.

At the present time:

- great reduction of financial transfers from central administration;
- need to found new asset in order to manage the hospitals healthcare expenses;
- excess of "required" healthcare services;
- absence of charge for medical health services eliminating the value perception to the system itself;
- absence of a "filter" system to the patients directly sent to the hospital rather than an absence of minimum culture of self-management of more banal symptoms and pathologies manageable at home;
- media aiming to "dramatize" situations of medical need;
- absence of synchronization of system and diagnostic process;
- duplication of diagnostic tests, time and resources waste in the patient management during its investigation path on its disease state;
- over esteem trend to the diagnostic investigation; avoiding accurate differential diagnosis (since emergency department resources are overwhelmed), but the use of as many as possible clinical test hoping to find the problem in a probabilistic way through the big amount of the carried out clinical test.

Outcomes:

- lot of resources waste in patient handling from home to the hospital;
- overburdened of diagnostic services and patients afferent departments (DEA emergency room), overcrowding phenomenon;
- great increase of medical health expenses not targeted to patient treatment and to the effective improvement of itself.

Patient path from the problem perception to its resolution.

At the present time, every system and organization managing (118 Emergency call, doctors on duty) information flow of medical support applicant knows that in the majority of cases, the problem is perceived heavy from the user (red and yellow codes for the 118) doesn't correspond to the real patient clinical state.

This absence of criticality is highlighted even before the beginning of hospital diagnosis process, but from the arrival on the site (patient home) of basic means of aid (bma) or advanced means of aid (ama).

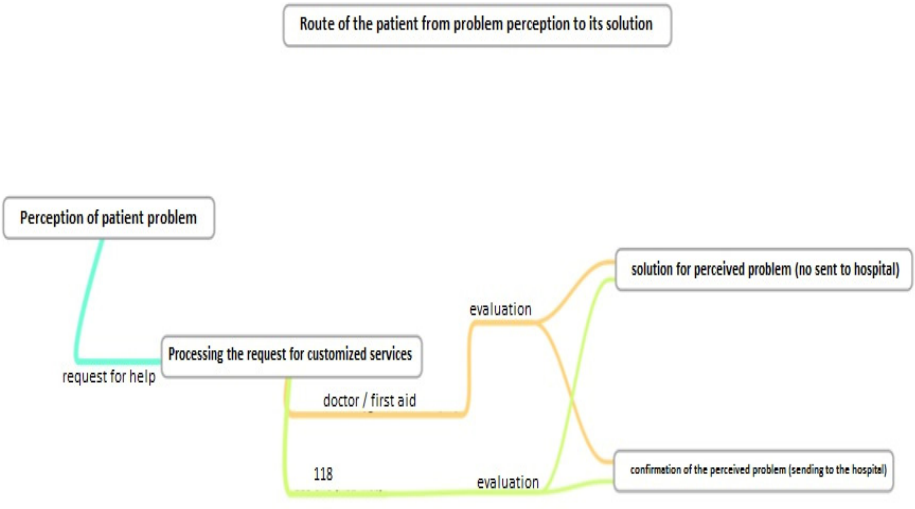


Fig. 2. Route of the patient from the problem to its solution

The particular impact of elderly people for emergency requests highlights that always is an "help" request.

With "help" the customer need to be supported , as for example on advices on assumed drugs dosage, reassured as to the over esteemed symptomatology, it means readdressed to a more correct self assessment of its perceived health state, is intended.

Case study:

It is considered a patient relatively elderly and normally in therapy for pathologies linked to the age, having a clinical stable state living at home alone or with a consort/partner in the same clinical state or even more complex one.

The problem beginning produces the start of user path, that not always has in its time scheduling and procedures a coherency fluency and quickness execution from request and answer.

1. The patient has a problem/perceives a problem; the first difficulty is who's to call? to whom to refer?

Normally the flow diagram for problems linked to the health care field (not important or emerging one (should follow) the following path:

Call to family doctor (from Monday to Friday from 8.00 to 20.00 and for other times and days, call to doctor on duty), that for lots of call could solve on the telephone questions, doubts, and so forth without the need of medical home or patient transfer to the hospital.

Criticalities:

2. Call at the "Family doctor"

- I don't find the family Doctor; I don't know his office study telephone number or other "Health Aggregates Offices", where the family doctor has other offices.
- I speak with the secretary not with the doctor.
- I won't be recalled by the doctor.

3. Call at the "Doctor on duty"

- I don't know how to call the Doctor on duty (I don't' know his telephone number, city number or toll free one).
- His telephone number is always busy and I must wait long time before receiving an answer. I think: "Nobody will respond to me", "Nobody has interest for me", with consequent "anxiety" rising.
- I don't know working days and hours of "Doctor on duty".

Initially the patient had only one doubt, one request of explanation, the desire to have advice; the lack of response and the reinforcement of the need of having it, changes the parameters of the request that are noticed as urgent and imminent: "I have a problem, I must talk with somebody; somebody must listen to me".

4. call to the 118 (only number for national territory emergency) (immediate answer to the user) that usually becomes the user "shelter".

The 118 service as structural and prerogative organization takes care of getting all calls from the medical emergency; another of 118 activities it is included also to give advice but principally it is oriented to codify, in the less possible time the gravity of the problem and send the most suitable help for the patient in the less time possible so that he could be treated and stabilized already in his territory, and then transfer him towards the hospital for the subsequent treatment (diagnostic work up, diagnosis, therapy, recovery and dismissal).

The celerity of the answer, and the possibility of talking with someone, is certainly one of the elements that the user that has no need of "emergency" (bma and ama) believes important. Indeed, the 118 is not the place where to ask only advice; as previously mentioned, the first of its prerogatives is the dispatch of suitable and qualified rescue vehicles, in the minor time possible. As a matter of fact one of the management software used in the operative centrals follow a list of questions particularly aimed to understand the principal pathology and most critical, making a photograph of the most possible scenery it is being described and send rapidly rescue.

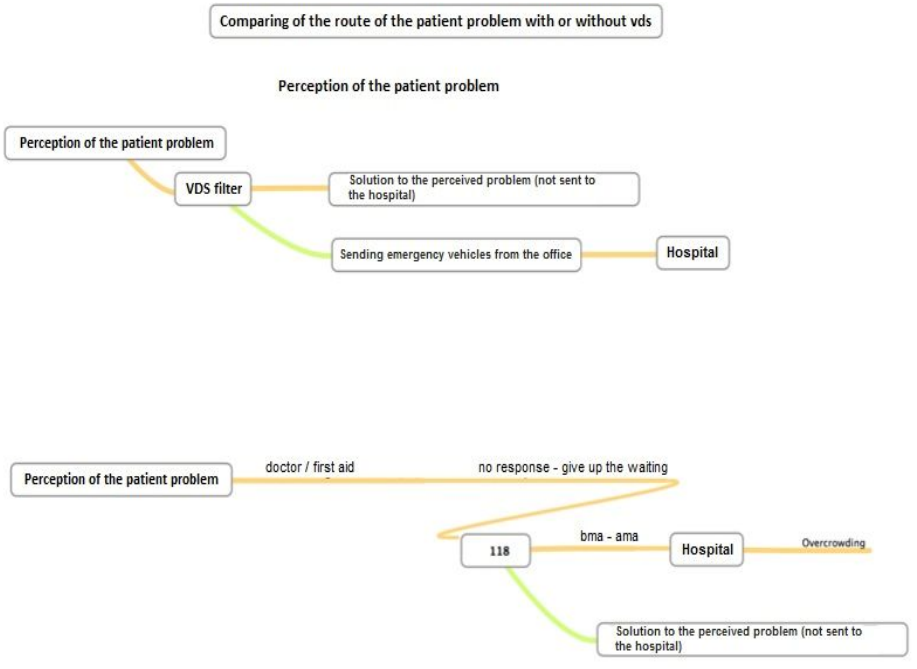


Fig. 3. Comparing of the route of the patient problem with or without VDS

For example, on the usage of central operative software that use the procedure and protocol for dispatch (i.e. protocol proQA):

- This, as one of the first questions they have to ask is: "118 good morning, where do I send the ambulance", starting the conversation with a lot of questions based on the protocol for the presumed pathology; and not "good morning 118 how do you feel today, I am here to listen to you". The result for the patient that finally has reached to talk with someone, even after various attempts with other suitable offices that could give an answer/advice (doctor), is that the fear of not being heard will make the client to overestimate the own symptomatology with the hope and need of someone that comes to help him (in listening) taking the risk of "cheating" the 118 system.

Even though the new 118 operative protocols have created questions that do not "induce" the patient to the affirmative answer, i.e. "do you have chest pain?" (if the patient has not understood one question he will answer yes). We have to remember that the patient has called because he needs help and for a question that he has not well understood, answering "yes" implies a better condition regards a negative answer "no", since for the patient in this moment "yes"= help, and "no"=no help.

Open questions (new 118 Operative Protocols) that don't advise the patient a predictable easy answer (for example where do you have pain?) is preferable.

However, forcing the system is relatively easy since the different Operative Protocols give some quick code procedures and/or the over estimation of the clinical situation.

In case of doubt, it is possible to assign to the patient a greater severity (red or yellow), with the subsequent dispatch of emergency vehicles (bma-ama).

It means that a great percentage of 118 phone calls: dispatch of emergency vehicles of ambulance to the place where the emergency occurred (example patient home) and consequent back to the hospital for other clinical exams, could be managed, without big problems, at the beginning through a system tool, based on a quick answer of a "Patient Service-VDS", dedicated and customized on patient.

In the example of the use of integrated VDS it is possible to consider the following scheme:

An intervention that can be solved with one phone call but basically could instead involve 3-6 persons of health care service and 1-2 emergency rescue vehicles only for first emergency and during hospitalization phase at least 7 persons:

- 2 nurses (1 triage 1 ps);
- 2 doctors (i.e. doctor, medical specialist);
- 1-2 technicians (i.e. lab, radiology);
- 1 operator social health.

Besides the use of various equipment for the diagnosis definition.

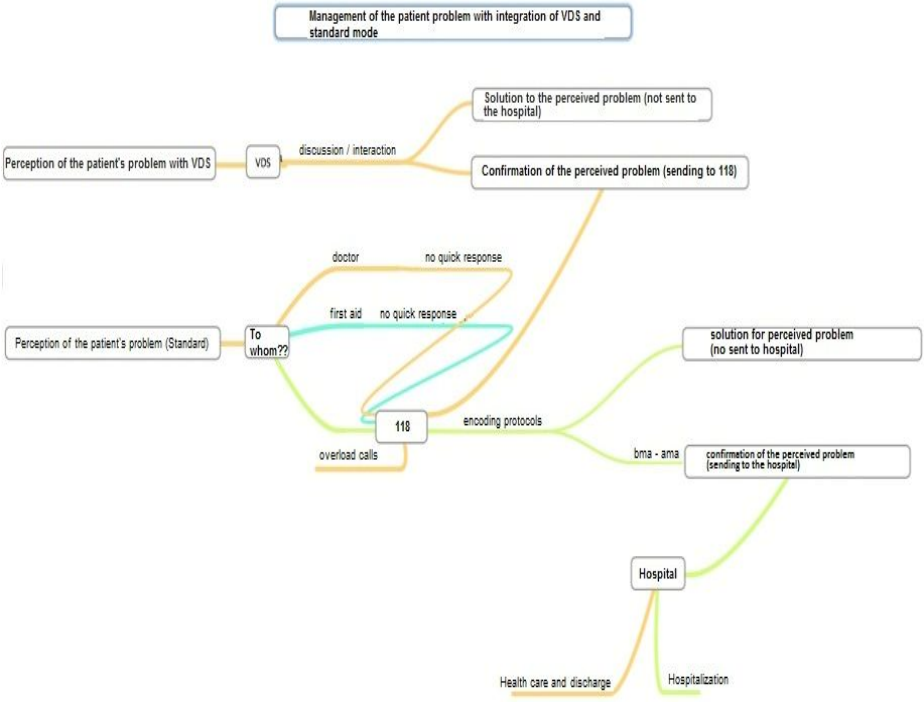


Fig. 4. Management of the patient problem with integration of VDC and standard mode

7 Results Analysis

In this way the results created by the use of VDS as a “Virtual Clinic” will be implemented in the VDS knowledge base resulting in an innovative form of medical interaction between MD and patients. In other word it will be possible to outline an innovative “shared mental model" made of several interacting layers consisting of:

1. A “Virtual Clinic” of local/particular medical experiences obtained from or simulated by mean of the VDS able to reproduce the MD behaviour as well as the doctor - patient interaction;
2. An high-level global network designed to enable sharing, amplification and catalyst for new ideas where result of experience and trials conducted can be shared, updated and directed towards the effective treatment of different diseases;
3. A technological platform where the interactions between the two layers consisting mainly of information exchanges and experiences both at the top level and at the low level will positively affect the experiences of MD and scientist to be prepared to response to unexpected natural or anthropogenic disasters with the help of simulation, case studies and direct experimentation.

8 Conclusion and Future Developments

All these activities will consider other further development to improve the system. First, the system could be published in a Cloud environment to be fully available from every part of the world. The queries could also be submitted through mobile devices such as PDAs or smart phones, ensuring a delocalized accessibility. Through this study you can also encourage the processes of de-hospitalization with significant improvement in patient welfare and reduced costs to the health system. The system could also be used from National Civil Protection to monitor people health in case of particular emergency (Flooding, earthquake). Results of the system as well memorized into the DW could be published in an aggregate manner through social networks for scientific research in order to share knowledge with other colleagues.

References

1. Kurematsu, M., Kinoshita, O., Hakura, J., Fujita, H.: An approach to implement listeners estimate emotion in speech. In: New Trends in Software Methodologies, Tools and Techniques (SoMeT 2009). Frontiers in Artificial Intelligence and application series, vol. 199 (2009)
2. Sasaki, J., Yamada, K., Tanaka, M., Funyu, Y.: A proposal of Life Support Network Architecture for Elderly People Living an a Rural Area. In: SoMeT (2008)
3. van Kasteren, T.L.M., Englebienne, G., Krose, B.J.A.: An activity monitoring system for elderly care using generative and discriminative models. Personal and Ubiquitous Computing (2010)

4. Fujita, H., Hakura, J., Kurematsu, M.: Virtual Medical Doctor Systems. In: HEALTHINF (2010)
5. Ball, T.: Designing integrated healthcare delivery systems. Managing Change (Winter 2009)
6. Evans, R.G., Cardiff, K., Sheps, S.: High Reliability vs. High Autonomy. Health Care Policy 1(4) (2006)
7. Guerra, L., Lombardo, F., Monica, L., Murino, T.: E-Supply Chain: sistemi di protezione dati per piattaforme on-line dedicate al e-collaboration. ANIMP - XXXII Convegno Nazionale Ingegneria e Impiantistica Italiana, Rimini (2005)

Detecting Changing Emotions in Natural Speech

Wojtek Kowalczyk and C. Natalie van der Wal

VU University Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
c.n.vander.wal@few.vu.nl
http://www.few.vu.nl/~cwl210

Abstract. The goal of this research was to develop a system that will automatically measure changes in the emotional state of a speaker, by analyzing his/her voice. Natural (non-acted) human speech of 77 (Dutch) speakers was collected and manually splitted into speech units. Three recordings per speaker were collected, in which he/she was in a positive, neutral and negative state. For each recording, the speakers rated 16 emotional states on a 10-point Likert Scale. The Random Forest algorithm was applied to 207 speech features that were extracted from recordings to qualify (classification) and quantify (regression) the changes in speaker's emotional state. Results showed that predicting the direction of change of emotions and the change of intensity, measured by Mean Squared Error, can be done better than the baseline (the mean value of change). Moreover, it turned out that changes in negative emotions are more predictable than changes in positive emotions.

Keywords: automatic speech classification, emotion recognition, natural human speech.

1 Introduction

Imagine that your telephone can continuously recognise the emotions you feel, through classifying acoustic features in your voice. The possibilities would be endless. You could use your phone to get insights in your own emotional well-being. Your telephone could be your therapist by listening to your voice and assess your emotional well-being. It could perhaps prevent you from depression by contacting your friends or a professional to help you or give you assignments to feel better. Your phone could also be a social coach, giving you lessons and ratings in how to use your voice to come across positive and enthusiastic instead of lethargic and negative. The current research stems from this vision.

The field of emotion recognition by machines is called affective sensing [11]. Besides the envisioned application indicated above, many other applications of emotion recognition are possible, for example emotion recognition in speech is used in call centers to detect anger in the voice of employees and to give them appropriate feedback [14]. Emotion recognition is useful in real-time conversations with embodied

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 491–500, 2012.

agents in human computer interaction, for example in computer games, but also in (web) applications with virtual therapists.

In the multidisciplinary research of emotion, many definitions of emotions exist. In this article, emotion is considered as elicited by a particular stimulus and relatively intense and short lived [6].

Many approaches and techniques are available in emotion recognition research. Emotional expression can be investigated in many different modalities, like gesture, posture, facial expression and speech; e.g.: [3], [10], [11], [13], [14]. In [11], the authors claim that affective sensing systems can recognize emotions in human voices and facial expressions better, if semantic features are analyzed as well, besides the standard analysis of sound patterns (like prosody and energy levels), and pattern matching or statistical machine learning techniques in facial expressions. Besides adding semantics to the emotion recognition process, it seems that multimodal emotion recognition gains higher accuracy than uni-modal emotion recognition. For example in [3] it is shown how emotion recognition through the multiple modalities: facial expressions, body gesture and speech, produces higher accuracy than through any of the single modalities. There are still big steps to be made in research in unimodal emotion recognition, for example in speech recognition. In [14] the main challenges in automatic emotion recognition from speech are discussed; how to segment audio files, how to extract the relevant features in these speech units and how to classify and train databases with emotional speech. In [1] the authors go deeper into one of these issues: which features in speech are the most important for high accuracy in emotion recognition systems? All of these emotion recognition systems analyze individual uni- or multimodal emotional fragments of human speech, facial expressions or body gestures.

In this article, the focus is on a single modality, namely speech. One reason for this is that speech is easy to capture/collect. A second reason is that it is easier to process, compared to images/video, which is important for the envisioned application that uses a telephone processor. The ultimate goal is to develop a system that will automatically measure changes in the emotional state of a speaker, by analyzing his/her voice. To achieve this goal the following research questions are addressed: 1) How accurate can the designed intelligent agent predict if a certain emotion in human speech is becoming weaker or stronger? 2) How accurate can it predict how substantial the change is?

The paper is organized as follows: in Section 2 the approach to the problem is described, explaining the key steps: data capturing, data segmentation, feature extraction and data modelling. Section 3 describes the results. In Section 4 the results are interpreted and some conclusions are made. In Section 5, challenges, possible extensions to and other applications of this research are discussed.

2 Method

In the current research the standard approach that is used in pattern recognition was followed, [5]: capturing data from sensors (in this case: microphones), data segmentation (in this case: manual splitting or recordings into meaningful pieces and labelling

them), feature extraction (converting each recording into a vector of features of fixed length), preparation of training sets, developing classification and regression models. In the current experiments, Random Forests were used, [2, 7], for their superior accuracy, good generalization properties and ease of training. An alternative method to Random Forests could be Support Vector Machines (SVM) with some well chosen kernel functions, [12]. However, algorithms for training SVM are very sensitive to the choice of learning parameters and are computationally more demanding than Random Forests.

2.1 Data Collection

The speech samples used in this research were collected from 77 participants (Dutch university students and employees, 44.2% women, mean age: 26.77 years) during a laboratory study in which the participants were induced with positive, neutral or negative emotions. The speech of the participants was recorded with a ZOOM H1 voice recorder, through a clip-on microphone, with the automatic gain control on. Participants were informed that their speech was recorded during the experiment, so that the researcher could write out their answers afterwards. However, the participants were not informed that their speech was recorded for the purpose of emotion recognition by a machine. In this way, naturally occurring speech was collected. This is beneficial for this study, because it does not have the pitfalls of acted emotional speech. Acted speech elicits how emotions should be portrayed, not necessarily how they were portrayed. Also, acted emotions do not emerge in the body and mind of the individual as naturally occurring emotions do. Another advantage of this data collection is that the participants own ratings of their emotions were collected. Each of the 77 participants recorded 3 short messages (containing their name, age and the city), while being in one of the 3 emotional states: "neutral" (entering the lab), "positive" (after watching a short "positive" movie), and "negative" (after watching a short "negative" movie). After their recording, each subject rated his/her emotional states on an Emotion Report Form containing the following 16 emotions: sadness, fear, shame, contentment, guilt, happiness, disgust, despair, positivity, enjoyment, irritation, hope, anger, pride, negativity, anxiety. Ratings were made on a 10-point Likert scale (1 = none, 10= a great deal).

2.2 Data Segmentation

The speech units were cut manually, starting exactly at the first phoneme spoken by the subject and ending directly after the last phoneme was spoken; periods of silence at the beginning and at the end of each recording were manually removed. The units differ in length for each subject, varying from 2 to 8 seconds. Each speech recording, originally stored in a wav format (stereo, 44.100 Hz frequency sampling), was converted into a single vector of numbers (signals from the two channels were combined into one: s_mono=(s_left + s_right)/2) and stored in Matlab.

2.3 Feature Extraction

In order to apply any classification or regression algorithm to our recordings, each recording had to be converted to a vector of fixed length of sound features. It was difficult to say beforehand, which features would be most successful. Therefore almost all features that could be found in the available literature on analysing affective speech: [1], [3], [8], [10], [9], [11], [13], [14], were used. These features were calculated in two steps:

(1) each recording was split into a sequence of short segments (10 milliseconds long), and a number of procedures were applied to each segment to calculate, among others, features like: fundamental frequency, energy, formants, cepstral coefficients.

(2) the values of the computed features were aggregated over the duration of the whole recording. The aggregates included mean, median, standard deviation, skewness.

The following table provides a complete overview of speech properties and aggregates that were used in the current experiments.

Table 1. All 207 speech features used by the classification algorithm

23 Speech Properties	**9 Aggregates for each Speech Property**
F0 (Fundamental Frequency)	Mean
I (Sound Intensity, measured on the logarithmic scale)	Median
E (Sound Energy, measured in sound units)	Standard deviation
F1, F2, F3 (first 3 formants)	Skewness
B1, B2, B3 (bandwidth of first 3 formants)	Kurtosis
MFCC0-MFCC12 (13 Mel Frequency Cepstral Coefficients)	Q1 (mean of the smallest 20% of values)
SR (speech restarts)	Q5 (mean of the biggest 80% of values)
	Shimmer (period to period variability)
	Rise (percentage of times next value is bigger than the previous one)

2.4 Training Sets

After calculating all features for each recording, the final training set was prepared for developing classifiers or regression models. The following 2 problems were addressed:

Classification: Develop, for each of the 16 emotional states, a classification procedure, which, when applied to two recordings S1 and S2 will determine if the emotional state in S2 is "more present" than in S1 (has higher intensity value).

Regression: Develop, for each of the 16 emotional states, a regression procedure, which, when applied to two recordings S1 and S2 will estimate the difference of the intensity of the emotional state between S1 and S2.

Additionally, it was assumed that both recordings, S1 and S2, are coming from the same speaker and the input for the classification and regression procedure consists of the differences between feature vectors of S1 and S2, and not the original values. In other words, the change in emotional state has to be predicted from the change in feature vectors. The main reason for this assumption was caused by the scarcity of data: for each subject there were only 3 recordings and (big) interpersonal differences in speech features would dominate the subtle changes that reflect emotional states.

The training set was constructed as follows. Let S1, S2, S3 denote 3 recordings of the same subject. This triplet leads to 3 input vectors: S1-S2, S1-S3, S2-S3, and the corresponding output values: E1-E2, E1-E3, E2-E3 (in the regression task), or +1 or -1 (in the classification task), depending on the sign of the difference (cases where the difference was 0 were ignored). In total, the training set used for developing our regression and classification models had 225 records.

2.5 Modelling Data with Random Forests

The concept of Random Forests was introduced by Leo Breiman in 2001 [2], and since then it became one of the most prominent technique for solving classification and regression problems [7]. The key idea behind this technique is a construction of many (hundreds) of de-correlated classification or regression trees and then aggregating their predictions. This leads to models with very good accuracy. The construction of a random forest of K trees for a set on N records is as follows:

> Repeat steps (1) and (2) K times to develop K trees:
>
> (1) draw a random sample of N records (with replacement) from the available data,
> (2) develop a classification or regression tree for the data sample in the following way: (a) whenever a splitting attribute has to be chosen, consider all possible attributes and select at random one of the top L best attributes, (b) whenever a node covers M (or less) records don't split it anymore.

When a Random Forest is applied to new data, outputs of all trees are either averaged (in case of regression), or the most frequent output label is chosen as a result (in case of classification). The Random Forest procedure involves 3 parameters: the number of trees to be developed, K, the number of best splitting attributes L, and the limit on the leaf size, M. Usually, the values of these 3 parameters are established experimentally with help of plots such as Figure 1. The figure represents the evolution of errors of 6 families of trees that are developed for the leaf size limit set to 1, 2, ..., 6 and L=50. It can be seen that M=4 and K=150-200 are the best parameters. In practice, the choice of value of K (the number of trees) is not critical: it has to be sufficiently big and further increase of this value has no impact on the tree accuracy. In this case, K was set to K=200. The optimal value of L was determined experimentally, by trying values L=25, 50, 75, 100; the best results were achieved for L=75. Finally, the choice of the minimal number of records in a leaf, M, was most difficult: it strongly depends on

the emotional state that we wanted to model (different states needed different values of M). To choose this value, the following heuristic was used. For each emotion, 6 values of M (1, 2, ..., 6) were tried, choosing the one that was best (had smallest average error) in the interval of 150-200 trees.

To avoid data overfitting the out-of-bag error estimates were used, as described in [7]. These estimates are computed as follows. Each tree from a random forest is trained on a sample of data (a "bag"). Because data is sampled with replacement, some records are not used in the training (they are "out-of-bag") so they can be used as a test sample to estimate the accuracy of the trained tree. Clearly, each tree is trained and tested on a different sample; therefore the average accuracy of all trees, measured on "out-of-bag" samples, gives a very reliable estimate of the true accuracy of the random forest.

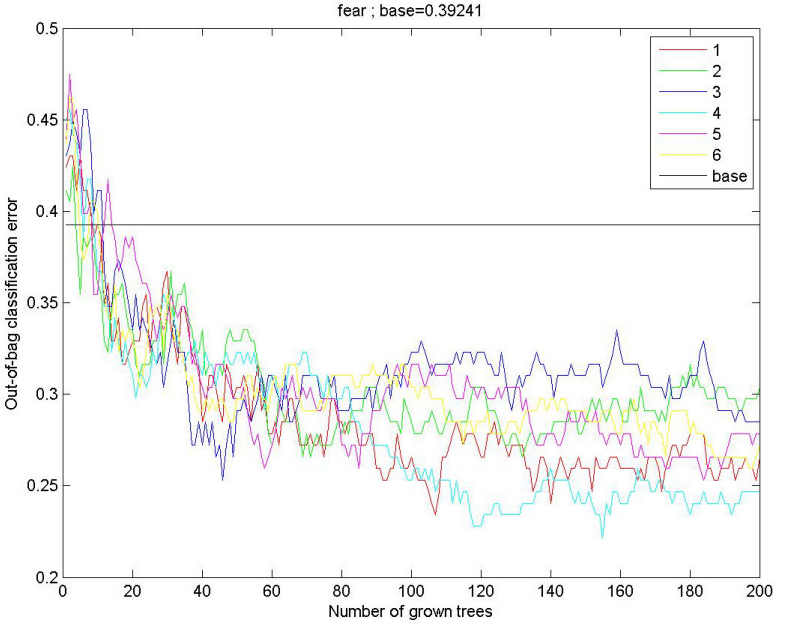


Fig. 1. The evolution of errors of 6 families of trees

3 Results

The results of our experiments are summarised in Tables 2 and 3. In case of classification models, the error was measured by the percentage of misclassified cases. This error was then compared to the baseline error: the error made by the base classifier that always predicts the most frequent category. In case of regression models, the error measure was the Mean Squared Error, MSE. The baseline model was defined as a constant function equal to the mean value of the predicted variable.

Finally, the Relative Error Reduction (RER) was calculated, which is defined as the ratio (BaselineError-ModelError)/BaselineError.

Table 2. Results of the classification problem. Error is measured by the ratio of misclassified records (misclassification rate).

Emotion	Baseline	Model	RER
sadness	0.4024	0.3629	13.24%
fear	0.3924	0.2764	24.19%
shame	0.4397	0.2543	45.10%
content	0.3109	0.3083	1.67%
guilt	0.3578	0.3792	-7.69%
happiness	0.4341	0.3943	-2.25%
disgust	0.2628	0.2596	2.44%
despair	0.3889	0.3545	18.37%
positivity	0.3418	0.3299	2.99%
enjoyment	0.399	0.3875	7.41%
irritation	0.3113	0.2671	6.38%
hope	0.3242	0.369	-11.86%
anger	0.3514	0.2995	0.00%
pride	0.2442	0.2529	-4.76%
negativity	0.3714	0.2981	7.69%
anxiety	0.3168	0.2629	9.80%

Table 3. Results of the regression problem. The Mean Squared Error measure is used (MSE).

Regression	Baseline	Model	RER
sadness	9.1392	8.4424	6.82%
fear	12.1157	10.4567	15.17%
shame	2.7825	2.4092	12.45%
content	12.7282	11.4304	8.61%
guilt	4.2478	4.1774	5.71%
happiness	15.7575	14.2415	9.79%
disgust	16.5909	14.3896	12.85%
despair	7.5872	6.8613	10.93%
positivity	15.3961	14.167	8.54%
enjoyment	17.1388	16.0236	5.18%
irritation	8.2867	6.9223	12.22%
hope	10.0526	9.8482	5.96%
anger	10.4548	9.3617	8.69%
pride	7.8706	7.4807	4.75%
negativity	13.6128	12.0076	14.86%
anxiety	15.0084	13.086	12.53%

3.1 Most Important Features

The Random Forest algorithm provides a powerful mechanism for measuring the importance of attributes that are used in the modelling process [7]. To measure the importance of an attribute, its values are permuted and the original accuracy of the model (measured on out-of-bag samples) is compared to the accuracy of the model on the modified data (on the same out-of-bag samples). The observed difference between

both accuracies is strongly related to the importance of the attribute: the bigger the difference the more important the attribute.

With the help of this method, the importance of each attribute was found and calculated, for each emotion. Due to lack of space only the most frequent attributes that were used by Random Forests will be listed. More precisely, for each model, the five most important attributes are listed, all 32 lists (16 for classification and 16 for regression) were concatenated and frequencies of the attributes on the list were computed. The most frequent attributes are listed in Table 4. The identification of the most informative features has 3 objectives: (1) verification of our findings with the existing literature, (2) simplification of the implementation of the automated system for monitoring emotions, (3) better understanding of the emotion-speech relation.

Table 4. Most important attributes in classification and regression models of detecting changing emotions and their intensity

Most important attributes	**Frequency**
M2std: the standard deviation of the 2nd cepstral coefficient	38 times
M6mean: the mean value of the 6th cepstral coefficient	16 times
PU: the period to period variability(shimmer) of speech energy	12 times
I: the speech intensity	12 times
M6: the 6th cepstral coefficient	12 times

4 Conclusion

In this research, the goal was to develop a system that will automatically measure changes in the emotional state of a speaker, by analyzing his/her voice. Natural human speech was collected in a laboratory study, from microphones and manually splitted and labelled into meaningful pieces. In total, 207 speech features were extracted. The Random Forests algorithm was used to address a classification problem (out of 2 recordings, which one has the highest emotional value?) and a regression problem (from recording S1 to S2, estimate the increase in each emotional state).

Results showed that predicting the direction of change of emotions can be done about 7% better than the baseline (the most frequent class label), while predicting the change of intensity, measured by the Mean Squared Error, can be done about 9.7% better than the baseline (the mean value of change). Moreover, it turned out that changes of intensity in negative emotions are more predictable than changes in positive emotions: the relative error reduction rate for these two groups was 11.2% and 7.1%, respectively. At first sight, these error reductions could seem relatively small, but in fact they are not. Given the modest size of the training set (only 3 recordings per person) these improvements are quite remarkable, together with the fact that the performance is significantly better than a random guess. Moreover, in the field of affective sensing these improvements are quite substantial, for example see significant relative error reduction rates of 7% in [4].

In general, it is much easier to detect changes in negative emotions than in positive ones. This is beneficiary for the envisioned application, where an intelligent agent needs to detect negative mood, to prevent the user from depression or to council the

user into a positive mood. Occurrence of emotions like fear, shame, and despair can be predicted, on average, 29.2% better than the baseline; change in their intensity can be predicted about 12.9% better than the baseline. Some emotions are very difficult to predict. For example, guilt, happiness, hope, and pride seem to be not predictable at all (see Table 2).

5 Discussion

The biggest challenge in our research was a very scarce set of recordings: just 3 records per subject. We believe that with the increase of the size of available data the accuracy of our models would dramatically improve. In practice, this should be easy to achieve: potential users of the final system will have to "tune" it to their specific voice and emotional states by providing, numerous speech samples with labels in the training phase.

A validation experiment has been started, to investigate how humans perform on the same data. Will humans perform better or worse than the machine learning algorithm? For which emotions can changes be detected better (or worse) by humans than machines? We expect the machine algorithm to perform better on these data, because humans can not find semantic cues in these recordings. Also, the acoustic emotional cues are difficult for humans to hear, because the recordings sound like the speakers are repeating a memorised sentence, a bit monotonic/non-emotional.

Other possible applications are: a warning system or a voice-based monitor of physiological functions. For example, when a person is angry, aggressive or just furious, the smart phone could generate alerts or warnings like: "you are too excited to drive a car, operate heavy machinery or talk to your children". We hypothesise that these warnings/alerts make the biggest impact if they are composed by the user itself. Furthermore, we expect that there is a direct relation between voice characteristics and physiological states of a person, like breath rate, blood pressure, heart rate, sugar level and cholesterol level. We would like to experimentally verify and quantify this assumption, so we could build a very cheap monitoring device which would translate observed characteristic of speech into values of physiological parameters. Moreover, the machine learning algorithm could be used in a computer game to acquire information about the emotional state of the player via his/her voice to either verify if the intended effect of the current game level/environment is really there in the player, or to adjust the game level/environment to the current emotional state of the player.

The current research stems from the vision of an application, where speech is captured, while the person is communicating through a phone, to detect the current mood of a person. Most smart phones offer the possibility to capture the facial expression as well, via the video camera. If this could be incorporated in the envisioned application, multimodal emotion recognition, through speech and facial expression, would be possible. Processing videos requires a lot of computing power though. Therefore, on the short term, real-time multimodal processing seems only feasible by applying facial recognition techniques to detect emotions in photos/still images taken of the user.

Moreover, in a follow up study, more audio files of the same speaker will be acquired, for better accuracy of the system. Besides the speaker's self evaluation of his or her emotional state, it would be interesting to supplement the self-evaluation with a psychophysical measurement, like the Galvanic Skin Response. Finally, trying Support Vector Machines as an alternative modelling method is also part of future work.

References

1. Batliner, A., et al.: Whodunnit – Searching for the Most Important Speech Feature Types Signalling Emotion-related User States in Speech. Computer Speech and Language (2010), doi:10.1016/j.csl.2009.12.003
2. Breiman, L.: Random Forests. Machine Learning 45, 5–32 (2001)
3. Castellano, G., Kessous, L., Caridakis, G.: Emotion Recognition through Multiple Modalities: Face, Body Gesture, Speech. In: Peter, C., Beale, R. (eds.) Affect and Emotion in HCI. LNCS, vol. 4868, pp. 92–103. Springer, Heidelberg (2008)
4. Zhang, C., Wu, J., Xiao, X., Wang, Z.: Pronunciation variation modeling for Mandarin with accent. In: Proc. ICSLP 2006, Pittsburgh, USA, pp. 709–712 (2006)
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley (2000)
6. Frijda, N.H.: The laws of emotion. Lawrence Erlbaum Associates Publishers, Mahwah (2007)
7. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, ch. 15, 2nd edn. Springer, New York (2008)
8. Kurematsu, M., Amanuma, S., Hakura, J., Fujita, H.: An Extraction of Emotion in Human Speech Using Cluster Analysis and a Regression Tree. In: Fujita, H., Sasaki, J. (eds.) Proceedings of the 10th WSEAS International Conference on Applied Computer Science, pp. 346–350. World Scientific and Engineering Academy and Society (WSEAS), Stevens Point (2008)
9. Laukka, P., Neiberg, D., Forsell, M., Karlsson, I., Elenius, K.: Expression of Affect in Spontaneous Speech: Acoustic Correlates and Automatic Detection of Irritation and Resignation. Computer Speech and Language 25, 84–104 (2011)
10. Li, X., et al.: Stress and Emotion Classification Using Jitter and Shimmer Features. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2007, pp. 1081–1084 (2007)
11. McIntyre, G., Göcke, R.: Towards affective sensing. In: Jacko, J.A. (ed.) HCI 2007, Part III. LNCS, vol. 4552, pp. 411–420. Springer, Heidelberg (2007)
12. Schölkopf, B., Smola, A.J.: Learning with Kernels. Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press (2001)
13. Tawari, A., Trivedi, M.: Speech Based Emotion Classification Framework for Driver Assistence System. In: Intelligent Vehicles Symposium (IV), June 21-24, pp. 174–178. IEEE (2010), doi:10.1109/IVS.2010.5547956
14. Vogt, T., André, E., Wagner, J.: Automatic Recognition of Emotions from Speech: A Review of the Literature and Recommendations for Practical Realisation. In: Peter, C., Beale, R. (eds.) Affect and Emotion in HCI. LNCS, vol. 4868, pp. 75–91. Springer, Heidelberg (2008)

An Ambient Agent Model for Support of Informal Caregivers during Stress

Azizi Ab Aziz, Jan Treur, and C. Natalie van der Wal

Agent Systems Research Group, Dept. of Computer Science, VU University Amsterdam
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
{mraaziz,treur,cn.van.der.wal}@few.vu.nl

Abstract. Caring for a depressed person may have substantial impact on the health and well-being of the caregiver. In this paper, an ambient agent model is proposed that supports caregivers, to prevent or decrease the burden in them and promote their well-being. The agent integrates a domain model of the functioning of the caregiver and the care recipient and their interaction, and exploits model-based reasoning to assess the caregiver's state in order to generate dedicated actions that are tuned to the circumstances.

Keywords: Integrative ambient agent model, caregiver stress, caregiving interactions, intelligent support.

1 Introduction

Ambient Intelligence applications in the health area usually focus on providing support for persons suffering from some disease or mental disorder (e.g., [1]). For the mental health area applications have been designed to monitor and support persons suffering from depression (e.g., [2]). However, often also persons in the daily environment of a depressed person are affected and may experience a heavy burden as an informal caregiver. In the therapeutic area also support for such informal caregivers, such as partners or family members has been developed; see, for example [7]. This paper focuses on these informal caregivers.

An ambient agent model is presented to provide support to caregivers, based on monitoring and assessing the situation of both the caregiver and care recipient, and determining dedicated support actions. The ambient agent model uses a computational model for caregiving interactions, adopted from [3], and exploits model-based reasoning to monitor and assess the situation, and guidelines adopted from [7] in order to generate support actions (based on these assessments) that are tailored to the persons and their states.

In the paper, first in Section 2 the integrative ambient model to support caregivers is introduced. Next, in Section 3 the effective treatments incorporated in the support model are discussed. In Section 4 simulation results for different types of scenarios are discussed. Section 5 addresses formal verification of the simulation results. Finally, Section 6 ends the paper with a conclusion.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 501–513, 2012.

2 The Integrative Ambient Agent Model

This section focuses on the integrative ambient agent model used to support caregivers. A basic element in the ambient agent model is the integration of the domain model within it. By incorporating the domain model, an ambient agent gets an understanding of the processes of its environment [1], [2]. Basically, there are two different ways to integrate a domain model within the agent model [4]. First, the domain model is used as a basis to perform analysis of the human's states and processes by reasoning about observations and specific sensors (analysis model). Second, the domain model is used as a foundation to provide support for the human (support model). These two models are used within the two corresponding components within the ambient agent model. Fig. 1 (dotted arrows, left hand side) shows these two types of integration of the domain model in the ambient agent model. A third way of using the domain model is to simulate human behaviour in order to test the ambient agent model (dotted arrow in Fig. 1, right hand side).

In Fig. 1, the solid arrows indicate information exchange between processes. In the ambient agent model, another component is introduced, namely a support action repository. This additional component keeps track of the generated support actions given by the ambient agent to the caregiver. Note that there are two incoming arrows into the analysis component. The first arrow provides information about the environment (care recipient stress, personality and resources), the second arrow provides information about already provided support to the caregiver (from the support action repository). The outcome of the analysis component has the form of assessments, and is used as input for the support component, another incoming arrow for the support component provides the already selected support actions and their frequency from the support action repository. The outgoing arrows from the support component define provided support actions to the caregiver and the support action repository. The support action repository will update the frequency of provided support action from this information. In the next section, the details of the analysis and support component will be discussed.

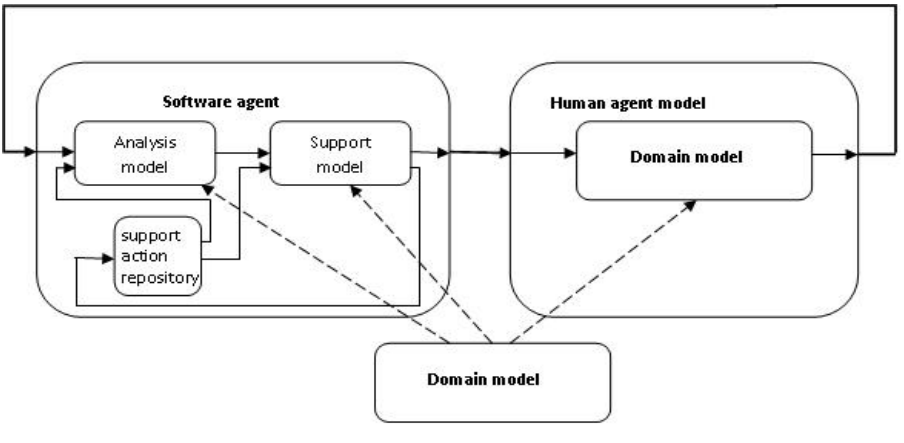


Fig. 1. The integration of a domain model within an agent model

2.1 The Analysis Component

First the analysis component is addresses; see Fig. 2. To be able to analyse the dynamics of the caregiver's and care-recipient's conditions, an ambient agent should be equipped with a domain model. In this case the domain model introduced in [3] is used.. Based on this knowledge, the ambient agent is able to have some understanding of the human processes and actions. Hence, the model for analysis should in principle include approximately the equivalent concepts as in the domain model. Note that not all concepts that exist in the domain model can be physically observed by the ambient agent [4]. For example, the level of 'experienced personal gain' is not something that is explicitly observable in the real world. To overcome this issue, the agent approximates values for such non physically observable variables by using beliefs derived using the integrated domain model. To capture important essences in analyzing caregivers' states, the following concepts are needed: (1) observations of primary stressors, caregiver personality and personal resources, (2) beliefs in (problem and emotional focused) coping characteristics, (3) beliefs in emotional exhaustion (short and long term), (4) beliefs in burden, (5) beliefs in experienced and expected personal gain, (6) beliefs in stress (short and long term), and (7) beliefs in social support. As can be seen, these concepts are similar to the concepts explained in [3], but as a form of integration embedded in observations or beliefs. For example, the concept of belief about a value V at time t for the variable of the domain model named long_term_stress is named belief(long_term_stress, V, t) in the analysis component. Using these embeddings of domain concepts, the ambient agent model is able to assess a caregiver's conditions and provide this information as inputs to the support component, using dynamical relations between such beliefs based on the corresponding dynamical relations in the domain model. For example, suppose in the domain model the following relation is given, specifying how state variable y depends on state variables x_1, x_2, x_3: $y(t+\Delta t) = y(t) + f(x_1(t), x_2(t), x_3(t))\,\Delta t$. Then this is integrated in the analysis model as (where $\rightarrow$ denotes a temporal causal relation):

$$\text{belief}(x1, V1, t) \wedge \text{belief}(x1, V2, t) \wedge \text{belief}(x3, V3, t) \rightarrow \text{belief}(y, f(V1,V2,V3), t+\Delta t)$$

Fig. 2 provides an overview of such dynamical relations in the analysis model. Note that for simplicity of notation here the values of the states are not mentioned.

2.2 The Support Component

The support model (see Fig. 3) can be specified in two different manners. First, the ambient agent can select support using the following causal representation:

$$\begin{aligned}&\text{assessment}(x1, V1) \wedge V1 > \text{threshold\_assessment\_1} \wedge \ .....\\ &\text{assessment}(xk, Vk) \wedge Vk > \text{threshold\_assessment\_k} \wedge\\ &\text{frequency\_provided\_support\_A} < \text{threshold\_frequency\_support\_A} \rightarrow \text{support\_action}(a1)\end{aligned}$$

Here $x_1,\ldots x_k$ represent the assessed conditions, $V_1,\ldots,V_k$ represent observed or estimated values, and a_1 represents a support action. From this representation, the ambient agent will activate support that matches the conditions expressed in the

antecedents. Note that all threshold values can be specified by a user. The frequency of provided support can be obtained from the action repository, and aims to discontinue from providing a specific support if the caregiver shows no improvement after previously receiving the same support. It provides a mechanism to diversify support provided by an ambient agent.

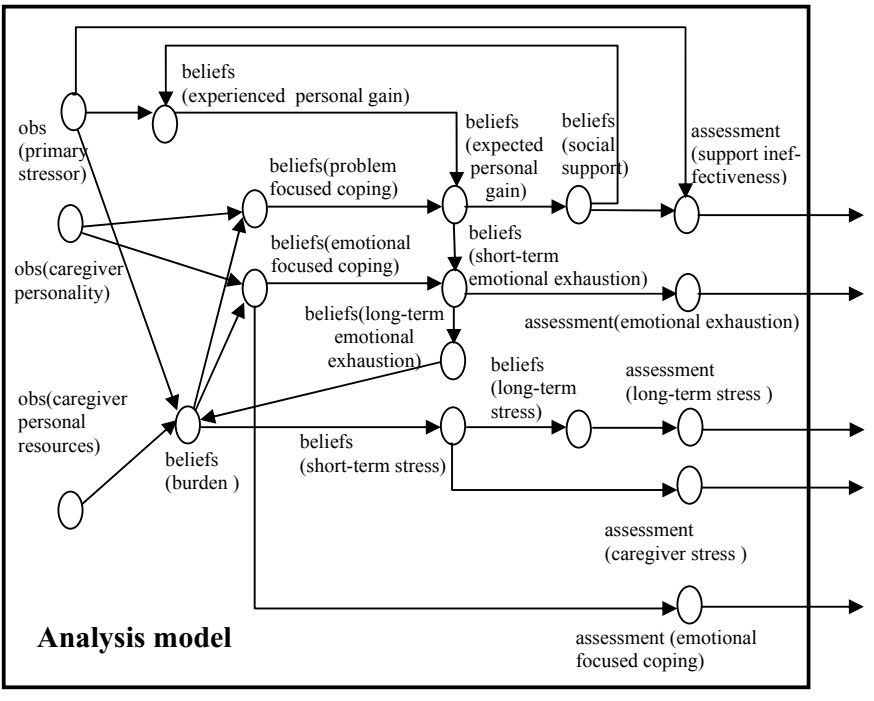


Fig. 2. Overview of the Analysis Model for the Caregiving Processes

Another approach to specify a support model is in a numerical manner, using the weighted networks. For this approach, each support action (e.g., a_1) will receive a summation of weighted input (y) from a set of selected assessments (x_i). For this, a continuous logistic function can be used, as in [14].

$$y(t) = \sum_i x_i \,.\, w_i \tag{1}$$

$$f(t) = \left(\frac{1}{1+e^{-\sigma(y(t)-\tau)}} - \frac{1}{1+e^{\sigma\tau}}\right).(1 + e^{-\sigma\tau}) \tag{2}$$

where w is a weight vector, σ is a steepness and τ a threshold parameter. In this choice, a common practice is followed (logistic function) but other types of combination functions can be specified as well. For this approach, the connection between the agent's assessment results and support actions is represented as follows:

$$\text{assesment}(x1,V1) \wedge .... \wedge \text{assesment}(xk, Vk) \rightarrow \text{support\_action}(a1, f(V1, ..., Vk))$$

where $f(...)$ represents a combination function.

Results from the continuous logistic function will be evaluated, where a support action with the highest value will be chosen. However, to allow flexibility in providing support, users could choose more support actions with the second or third highest values. The details of the support component can be found in Section 3.2. Fig. 3 shows the relationship between results from the analysis component (assessments) and support actions.

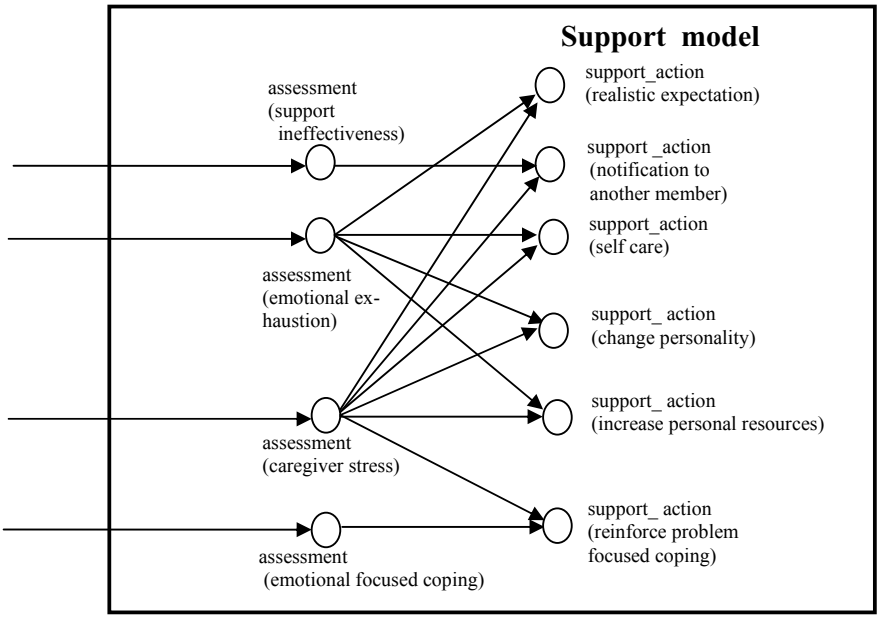


Fig. 3. Overview of the Support Model for a Caregiver

3 Concepts and Effects in Support for Informal Caregivers

This section explains how the proposed model incorporates characteristics of effective treatments for family caregivers in general and those specific to caregivers of depressed persons. By specifying these characteristics of effective treatments for caregivers, the proposed model should be as effective as possible to current standards and knowledge.

3.1 Important Concepts in a Support Model

Zarit and Femia [15] describe four characteristics of effective treatments for caregivers: a psychological approach, multidimensionality, flexibility and sufficiency. The psychological approach refers to practicing new skills and behaviours by caregivers in a group or one-to-one interventions with a psychotherapist. Multidimensional interventions are interventions that address multiple stressors and risk factors that affect the caregiver, instead of just one stressor or risk factor. Flexibility means that an

effective treatment is flexible in its set up: it should not be a scripted protocol intervention, but the intervention should be adjustable to the needs of the caregiver [11]. Sufficiency can refer to provision of ongoing support to caregivers, for example, by ongoing support groups, follow-up sessions of an intervention. These four characteristics have been integrated in the proposed support model for family caregivers of depressed persons as follows:

The psychological approach can be found in the indirect referral to support groups by the ambient agent and in the direct support actions of 'reinforce problem focused coping', 'realistic expectations' and 'increase personal resources caregiver'. The supportive actions are set up in a way that the caregiver is instructed, how to apply general theories to his/her own specific situation and to stay motivated to make plans how to implement these new skills. The ambient agent also gives the caregiver feedback on how he/she is implementing the new skills. The proposed model is also multidimensional, in that it focuses on many possible stressors and risk factors of the caregiver (personality, finances, coping skills, thinking skills, own health). Flexibility in the proposed model can be found in the continuous monitoring of the caregiver by the ambient agent and therefore continuous adjustment of the intervention to the needs of the caregiver. Finally, sufficiency is also integrated in the proposed model by providing ongoing support to the caregiver. Sufficiency and flexibility are the main advantages of the proposed model. The multidimensionality and psychological approach are still open for new insights from research.

Next, it is explained how characteristics of treatments, especially, for caregivers of depressed persons were integrated into the proposed model. Cuijpers [7] describes an intervention for family caregivers of depressed persons, based on his experience. There are eight ways for caregivers to deal with the depressed person they care for, which are shown (translated from Dutch) in Fig. 4. These eight steps are integrated in the proposed model, as well as the seven ways as Cuijpers describes to relieve the burden or stress experienced by the caregiver, shown in Fig. 4 (translated from Dutch) [6].

Eight steps in intervention of family caregivers of depressed persons	**Seven ways to relieve the burden or stress experienced by the caregiver**
1. Gather information 2. Do not try to cure the depression 3. Keep life simple 4. Communicate better 5. Don't give too much criticism, do not get too involved 6. Take good care of yourself 7. Watch relapse signs after recovery 8. Watch out for suicidal signs	1. Dealing with your emotions/feelings 2. Take good care of yourself 3. Learn to think different 4. Learn to relax 5. Make a good time planning 6. Change your social interaction with the depressed person 7. Be assertive

Fig. 4. Eight steps in the intervention of family caregivers of depressed persons and seven ways to relieve the burden or stress experienced by the caregiver

The current support model consists of multiple supportive actions advised by the ambient agent to the caregiver. The first support action is called "increase personal

resources caregiver". This support action is aimed at teaching the caregiver to manage stress, which will decrease the burden. Examples are teaching the caregiver to make a to-do list and becoming more assertive, like in points 5 and 7 of the left column in Fig. 4. This will affect the caregiver's personality (as in changing his/her stress reactions: now he/she gets well organised, and more assertive) and the caregiver's social and financial resources (as in getting financial/practical help from friends/family).

The second support action is called "reinforce problem focused coping caregiver". Here the ambient agent teaches the caregiver how to learn to apply problem focused coping instead of emotion focused coping and gives feedback. Research shows that coping is a learnt behaviour, see a review in: [13]. Examples are: text messages or instruction movies on phone/through emails, in which it is shown how to deal in certain situations or dialogues with the depressed person. Also the ambient agent will ask to plan and report the new skills the caregiver has to apply, so it can monitor the newly developed skills and give feedback to the caregiver. This support action decreases the caregiver's emotion focused coping and increases the caregiver's problem focused coping: increases. These skills fall under points 2-5 in the left column of Fig. 4 and 1,3,6,7 in the right column.

The third and fourth support actions are called "realistic expectations and self-care caregiver". In these actions, the ambient agent gives information about the illness so the caregiver gets an understanding of the behavioural patterns and needs of the depressed person (corresponding to point 1 in the left column of Fig. 4). Also the ambient agent teaches the caregiver to take care of him/herself (physically, emotionally, and mentally) and asks for reports and plans and gives feedback (point 6 in the left column of Fig. 4, points 2, 4 in the right column). Examples are: text messages or movies on phone/through emails, in which examples of the behaviours of other depressed persons are given, like how fast they recover or relapse. Giving tips in self-care, like taking a time-out, finding social support, eating healthy, exercising regularly and learning relaxation exercises. These support actions increase the caregiver's experienced personal gain, because (s)he experiences less disappointments as the caregiver learns to have more realistic expectations towards the depressed person [10]. The caregiver's short term emotion exhaustion will also decrease.

The fifth support action is aimed at other persons than the 'main' informal caregiver, namely other (possible) caregivers, friends of the 'main' caregiver, or a specialist like a doctor or therapist. This support action is called giving warning and refers to the ambient agent giving information to another person than the caregiver it is supporting. This information contains a warning signal that the depressed person and the caregiver both need support from others. The effect of support from an ambient agent to the caregiver will be dealt with in the next section.

3.2 Dynamics Specifications of the Effects from a Support Model

Using the support concepts introduced in the previous sections, it is possible to specify computational properties to visualize the effects from the support provided by a support agent. The dynamic specifications of an agent-based support can be structured pertinent to the purposes of the support, namely; (1) to reduce long-term exhaustion in a caregiving process, (2) to develop problem-focused coping skills, and (3) to improve personality attributes that reduce the physiological signs of stress [6], [8], [13].

Support to Reduce Long-Term Emotional Exhaustion. In this case, the support agent aims to reduce further negative influences that cause emotional exhaustion. From Table 1, the support agent will provide important advices and suggestions to regulate self-care (*Sc*), to increase external personal resources (*Ep*), and to foster more realistic expectations. The effect of short-term emotional exhaustion (*EsH*) after following agent's support is estimated after adding a new support parameter, δ_{SA} and a self-care effect into equation 3. This indicates that when self-care, $Sc(t) \rightarrow 1$ and $\delta_{SA} \rightarrow 1$, then the short-term emotional exhaustion (*EsH*) is recuced to zero. Another important effect after following the support is having more external personal resources. Thus, a new caregiver personal resource (*GgR*) can be expressed as having a combination of existing resources (*GpR*) and external resources (*Ep*).

$$EsH_A(t) = Md_A(t).(1-PgN_A(t)).(1-\delta_{SA}.Sc(t)) \tag{3}$$

$$GgR_A(t) = \delta_{EA}.\ GpR_A(t) + (1-\delta_{EA}).Ep(t) \tag{4}$$

The new value of experienced personal gain (*EpN*) depends on a combination of the previous equation (12) in [3] and a support contribution when a person is capable to achieve realistic expectations (*Re*).

$$\begin{aligned} EpN_A(t+\Delta t) = EpN_A(t) + \vartheta.(\delta_{RA}.[(Pos((Scp_A(t) - GpS_A(t)) - \\ EpN_A(t)).(1-EpN_A(t))) - Pos(-((Scp_A(t) - GpS_A(t)) - \\ EpN_A(t)).EpN_A(t))] + (1-\delta_{RA}).Re(t).(1-EpN_A(t))\Delta t. \end{aligned} \tag{5}$$

Support to Reduce Dependency on Emotional-Focused Coping Skills: In order to visualize the effect when a person follows agent's advices to reinforce problem-focused skills, both new problem-focused (*PfC*) and emotional-focused coping (*EfC*) skills are calculated as follows:

$$PfC_A(t) = GpP_A(t).(1-((1-\delta_{FA}.Rp(t)).Bd_A(t))). \tag{6}$$

$$EfC_A(t) = (1-GpP_A(t)).Bd_A(t).(1-\delta_{FA}.Rp(t)). \tag{7}$$

where δ_{FA} determines the influence of the acceptance in change coping skills and *Rp* represents reinforce problem focused coping skills, *Bd* represents burden and *GpP* represents the caregiver personality.

Support to Reduce Physiological Signs of Stress: For this type of support, changes in both the caregiver personality and resources are needed. In this case, a new caregiver personality (G*pP*) is calculated by combining the existing personality, and the positive personality (*Cp*) from the support.

$$GpP_A(t) = \delta_{PA}.GpP_A(t) + (1-\delta_{PA}).Cp(t) \tag{8}$$

In addition to this, δ_{EA}, δ_{RA} and δ_{PA} are support-acceptance parameters; it represents a person's ability to accept respectives changes from the support.

4 Simulation Results

The ambient agent model presented in Section 3, integrating the domain model as described in Section 2 was implemented in Matlab in order to perform simulation experiments. For the simulations, the functioning of the designed system was explored in interaction with three fictional types of caregivers (caregiver 1, 2 and 3: CG1, CG2 and CG3. Both caregivers 1 and 2 are ineffective caregivers and susceptible for long-term stress in a caregiving process (low in positive personality and resources), while caregiver 3 is an effective caregiver. Caregiver 1 ignores the support provided by the intelligent support agent, and caregiver 2 follows the support. In addition to this, information about the care-recipient's (CR) stress buffer and long-term stress has been used to measure the outcome of the agent support (as in [2]). The care-recipient's stress buffer represents a process of support protecting the care recipient from potentially adverse effects of stressful events (stressors). Therefore, many studies have shown that a high stress-buffer level will reduce the development of care recipient long-term stress level in future [6][9]. In the simulations, the care recipient is experiencing negative events (stressors) and expects supports from a caregiver (also facing incoming stressors).

Table 1. Initial Values for the Simulation Experiments

	Caregiver 1 (CG1)	Caregiver 2 (CG2)	Caregiver 3 (CG3)	Care recipient (CR)
CG personality	0.2	0.1	0.8	-
CG personal resources	0.2	0.1	0.7	-
CG empathy	0.3	0.3	0.7	-
CR personality	-	-	-	0.3
CR coping skills	-	-	-	0.1

Several parameters that can be varied to simulate different characteristics were set as: $t_{max} = 1000$ (to represent a monitoring activity up to 42 days), $\Delta t = 0.3$, regulatory rates = 0.5, flexibility rates = 0.2, and support-acceptance rates = 0.3. These settings were obtained from several experiments to determine the most suitable parameter values for the model. In addition, the weighted network is implemented in the support model to select the most appropriate support. All caregivers receive support by the agent after half of the simulation period.

Below, the outcome of the simulation experiment with CG2 and the CR is shown. The other simulations with CG1 and CG3 can be found in Appendix A [16]. In the following case (see Fig. 5) an ineffective caregiver follows the provided support from a support agent. After following the recommended advices, the caregiver improves his / her ability to provide support. One of the precursors to explain this outcome is the increasing caregiver's personal gain. It is consistent with the findings that suggest that caregiving satisfaction encourages a caregiver to provide more support [10], [12]. In addition to this, by following the specific advices, the caregiver is helped to apply more focused-coping skills, which later on influence the development of positive relationship focused coping. In many reports in the literature, problem-focused coping skills give a positive outcome in a caregiving process, for both caregiver and care recipient.

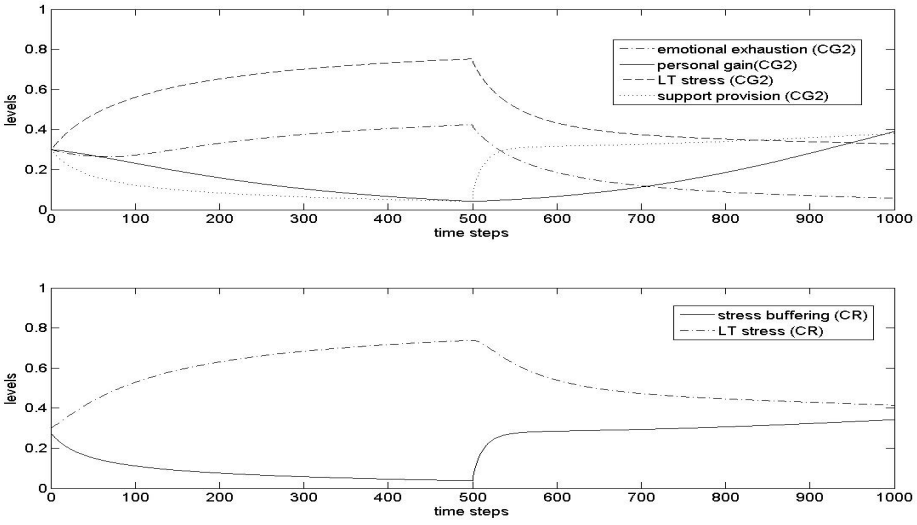


Fig. 5. An ineffective caregiver (CG2) with support, and a bad care recipient (CR)

5 Verification of the Simulation Results

In order to verify whether the model indeed generates results that adhere to psychological theories, a set of properties have been identified from related literature. These properties have been specified in a language called Temporal Trace Language (TTL). TTL is built on atoms referring to states of the world, time points, and traces. This relationship can be presented as holds(state(γ, t), p) or state(γ, t) |= p, which means that state property *p* is true in the state of trace γ at time point *t* [5]. It is also comparable to the *Holds*-predicate in Situation Calculus. Based on this, dynamic properties are formulated using a hybrid sorted predicate logic approach, by using quantifiers over time and traces and first-order logical connectives such as ¬, ∧, ∨, ⇒, ∀, and ∃. A number of simulations including the ones described in Section 4 have been used as basis for the verification of the identified properties and were confirmed. Note that tb and te are the initial and final time point of the simulation.

VP1: Monotonic Decrease of Long-Term Stress
For all time points t1 and t2 between tb and te in trace γ1

if at t1 the value of the caregiver's long-term stress is R1 and at t2 the value of the caregiver's long-term stress is R2 and t1 < t2, then R1 ≥ R2

∀γ: TRACE, ∀R1, R2: REAL, t1,t2:TIME
[state(γ,t1) |= long_term_stress(cg, R1) & state(γ,t2) |= long_term_stress(cg, R2) &
tb ≤ t1 ≤ te & tb ≤ t2 ≤ te & t1 < t2 ⇒ R1 ≥ R2]

By checking property VP1, one can verify whether a caregiver's long term stress decreases monotonically over a certain time interval. For example, the caregiver's long-term stress turned out to decrease over the second half of the trace for caregivers that have received and accepted the provided support or for an effective caregiver.

VP2: Decrement of a Caregiver's Long-Term Stress Below a Certain Level x
A time point t exists such that for all t1>t the value of long-term stress is at most level x.

$\forall\gamma 1$: TRACE, $\exists t\ \forall R$: REAL [tb < t < te & $\forall$t1:TIME>t
[t ≤ t1 ≤ te & state(γ,t1) |= long_term_stress (cg, R1) $\Rightarrow$ R1 ≤ x]

Property VP2 can be used to verify whether a variable eventually approaches some (given) value. In the experiments reported here, x = 0.3 was used as a borderline value for long-term stress to assume a caregiver is effective to provide social support. In many cases, after following the advices, the caregiver will reach this borderline value. A number of more specific other properties have been identified and verified, such as the following ones, which compare cases with a specific type of support and cases without. Note that formalisation of such comparison properties makes use of the possibility to explicitly refer to traces in the language TTL; this is not possible in the usual temporal logical languages.

VP3: Effect of Problem Coping Skills on a Caregiver's Long-Term Stress
After a caregiver has followed the programme to improve problem focused coping skills for some time, the long-term stress level is more reduced than for a caregiver who does not.

$\forall\gamma 1,\gamma 2$: TRACE, $\forall$R1, R2: REAL, t1,t2:TIME
[state(γ1, t1) |= support_problem_coping & state(γ2, t1) |= not support_problem_coping &
state(γ1, t2) |= long_term_stress(cg, R1) & state(γ2, t2) |= long_term_stress(cg, R2) &
t1 < t2 $\Rightarrow$ R1 < R2]

VP4: Effect of Realistic Expectation on Emotional Exhaustion
After a caregiver has followed the support programme to reduce unrealistic expectation, the long-term emotional exhaustion is more reduced than for a caregiver who does not.

$\forall\gamma 1,\gamma 2$: TRACE, $\forall$R1, R2: REAL, t1,t2:TIME
[state(γ1, t1) |= support_realistic_expectation &
state(γ2, t1) |= not support_realistic_expectation &
state(γ1, t2) |= long_term_emotional_exhaustion(cg, R1) &
state(γ2, t2) |= long_term_emotional_exhaustion (cg, R2) &
t1 < t2 $\Rightarrow$ R1 < R2]

VP5: Effectiveness of Support on Provided Support to the Care Recipient
A caregiver who follows the suggested support by an agent will provide better support to the care recipient than a caregiver who does not.

$\forall\gamma 1,\gamma 2$: TRACE, $\forall$R1, R2, d: REAL, t1,t2:TIME
[[[state(γ1, t1) |= support_realistic_expectation & state(γ1, t1) |= support_problem_coping &
state(γ1, t1) |= support_add_personal_resources] &
state(γ2, t1) |= not support_realistic_expectation | state(γ2, t1) |= not support_problem_coping |
state(γ2, t1) |= not support_add_personal_resources] &
state(γ1, t2) |= long_term_stress(cg, R1) & state(γ2, t2) |= long_term_stress(cg, R2) &
t1 < t2 $\Rightarrow$ R1 < R2]

6 Conclusion

In this paper, an ambient agent model was proposed that supports caregivers for depressed persons and promote their well-being. Caring for a depressed person may entail a serious risk for the health of the caregiver. The designed ambient agent integrates a domain model of the functioning of the caregiver and the care recipient and their interaction. It exploits model-based reasoning to monitor and assess the caregiver's state using this computational model. Based on these assessments dedicated support actions are generated that are tuned to the circumstances, thereby taking into account guidelines from psychological theories about social support and stress buffering.

Although some applications have been designed to support persons with a depression, automated support for caregivers has not been addressed, as far as the authors know. The model introduced here was evaluated by conducting a number of simulation experiments for different scenarios and types of caregivers, and formal verification of the outcomes of these experiments. These outcomes show that using the advices provided by the ambient agent results in improvement in the situation in comparison to not using such advices; for verification of this type of comparison properties (which are not representable in other temporal languages; see also [5]) the language TTL and its software environment [5] has been succesfully used.

References

1. Aarts, E., Harwig, R., Schuurman, M.: Ambient Intelligence. In: Denning, P. (ed.) The Invisible Future, pp. 235–250. McGraw Hill, New York (2001)
2. Aziz, A.A., Klein, M.C.A., Treur, J.: An Integrative Ambient Agent Model for Unipolar Depression Relapse Prevention. Journal of Ambient Intelligence and Smart Environments 2, 5–20 (2010)
3. Aziz, A.A., Treur, J., van der Wal, C.N.: Modelling Caregiving Interactions during Stress. In: Yao, Y., Sun, R., Poggio, T., Liu, J., Zhong, N., Huang, J. (eds.) BI 2010. LNCS (LNAI), vol. 6334, pp. 263–273. Springer, Heidelberg (2010)
4. Bosse, T., Hoogendoorn, M., Klein, M.C.A., Treur, J.: An Ambient Agent Model for Monitoring and Analysing Dynamics of Complex Human Behaviour. Journal of Ambient Intelligence and Smart Environments 3, 283–303 (2011)
5. Bosse, T., Jonker, C.M., van der Meij, L., Sharpanskykh, A., Treur, J.: Specification and Verification of Dynamics in Agent Models. International Journal of Cooperative Information Systems 18, 167–1193 (2009)
6. Cohen, S., Wills, T.A.: Stress, social support, and the buffering hypothesis. Psychological Bulletin 98, 310–357 (1985)
7. Cuijpers, P.: Depression: Guide for Family Members. Baarn, HB uitgevers (2004) (in Dutch)
8. Folkman, S.: Personal Control, Stress and Coping Processes: A theoretical analysis. Journal of Personality and Social Psychology 46, 839–852 (1984)
9. Groves, L.: Communicating Social Support. Social Work in Health Care 47, 338–340 (2008)
10. Kramer, B.J.: Expanding the Conceptualization of Caregiver Coping: The Importance of Relationship Focused Coping Strategies. Journal of Family Relations 42, 383–391 (1993)

11. Musil, M.C., Morris, D.L., Warner, C., Saeid, H.: Issues in Caregivers' Stress and Provider's Support. Research on Aging 25, 505–526 (2003)
12. Ostwald, S.K.: Caregiver Exhaustion: Caring for the Hidden Patients. Adv. Practical Nursing 3, 29–35 (1997)
13. Shankland, R., França, L.R., Genolini, C.M., Guelfi, J.D., Ionescu, S.: Preliminary study on the role of alternative educational pathways in promoting the use of problem-focused coping strategies. European Journal of Psychology of Education 24, 499–512 (2009)
14. Treur, J., Umair, M.: A Cognitive Agent Model Using Inverse Mirroring for False Attribution of Own Actions to Other Agents. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 109–119. Springer, Heidelberg (2011)
15. Zarit, S., Femia, E.: Behavioral and Psychosocial Interventions for Family Caregivers: Characteristics of effective approaches and flaws in study design. American Journal of Nursing 108, 47–53 (2008)
16. See, http://www.few.vu.nl/~wai/Papers/IEA2012_supportcaregiver_appendix.pdf

Interruption Theory for Improving Work Efficiency by Reasoning Workflow

Kohei Sugawara[1] and Hamido Fujita[2]

[1] Iwate Prefectural University,
Faculty of Software and Information Science, 152-52 Sugo, Takizawa,
Iwate-gun, Iwate, 020-0173, Japan
[2] Sangikyo Corporation,
4509, Ikebe-cho, Tsuzuki-ku, Yokohama, 224-0053 Japan
sugawarak@sangikyo.co.jp

Abstract. corporation is operated based on the strategy for having an enterprise to be sustainably growing. The workers who are employees and managers run several tasks simultaneously based on the workflow reflected on their experience and knowledge. Such workflow's tasks affect each other interactively due to invocation of an interrupt. When the multiple workflows are overlapped, the continuity of the active task is interrupted. The evaluated interruption invokes positive or negative effect according to the context of the tasks. In this paper we discuss the attributes of the tasks interruption, to provide an assistance in resuming or sustaining (i.e., interrupted) task work from interruption by the system to achieve suistabanle workflow and possible optimization.

1 Introduction

Interruption is occurred by breaking the continuity of the current operation based on workflow. In other words the interruption involves in partial participation within the engagement between the task and the agent employing that task. This disengagement can participate to produce another workflow that would make the interruption task be part of the workflow. Interruption is formalized through two parametric views that are time and context. The time parametric view is related the interruption impact on the work processing time. This time parametric view is represented as positive (i.e., affect in forwarding manner on the workflow) or negative (i.e., affecting the backward manner on the workflow). The representation of this time parametric view is reflected through a cognitive map model. The cognitive map model indicates semantic relation between causal interruption and impact to the workflow. The parameter related to the context is defined as the subjective relation of the interruption to workflow affect. Interruption is classified in the knowledgebase according to the company case studies collected from company doing business. The context of the workflow is different by strategy of the enterprise. Interruption invokes new activity that has process related to the context. The workflow consists of a set of tasks to achieve the objective by steps. The steps for goal in Lewin theory represent "cognitive structure" as workflow [1].

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 514–520, 2012.

The interruption on the cognitive structure expects success or failure. The tasks are action or process for operating the workflow such as development, communication, checking other work or searching information. Business enterprise consists of a set of workflow that we called multiple workflows. These multiple workflows are component of the whole enterprise decision making. The multiple workflows are operated simultaneously by the workers who are employees and managers. These workflows contribute to construct business or enterprise workflow that is used to do decision making based on input parameters (or values), specifying the appropriate decision making. The values of these parameters are articulated from knowledgebase decision making strategies. These strategies are collected from knowledgebase expertise (experts), and represented as input values to workflow for producing relative decision making. When an interruption is occurred, the priority is changed by the context of the interruption. A workflow has several innate values that are profit or acquisition of the desire in the future or recent appeared. Individual human profile that is skill, experience, role or environment is related as essential items for the workflow to get high value result from the achieved workflow. The criticality of the interruption affects the priority of the workflow. If the value of affected workflow by the interruption is higher than the value of the original workflow, the criticality of the interruption is high. Interruption invokes new process and it generates new scenario for adjusting actual environment that includes new process. The generated tasks progress or reduce the values of the active workflow by the interruption. The value for organization is calculated by aggregating the values related to workflows. The estimated total value for each workflow effects the decision making for the criticality of the interruption. When a non-critical interruption is occurred, a worker can decide whether the generated process by the interruption is started or this interruption is rejected. If the critical task is occurred by the interruption, the continuity of the actual operation is disturbed by the occurred interruption and the interruption imposes to produce and to operate new task as prior task. Interruption also invokes to reduce the human working memory with time in cognitive psychology as one of the backward manner on the workflow [2]. The working memory is a limited-capacity system in which some resource is shared between processing and storage based on trade-off: Performance decreases when the concurrent memory load increases by interruption, and any increase in difficulty of processing results in a loss of information from short-term storage memory [3]. The relation of both the time and the degree of the decreasing memory is represented by Ebbinghaus's Forgetting Curve [2]. The negative effect of the lacked working memory is appeared when the past task is resumed from interruption tasks. Authors are previously proposed the GUI for assisting the recalling past activities by indicating the historical desktop environment [4]. The optical short period interruption is also frequently occurred by overlapping each object window in computer operation [5]. Changed situation by interruption generates the mismatch of the workflow because the environment is influenced by negative or positive interruption. Even if the interrupted work is just resumed without reducing the short-term memory, the objective of original workflow may not be achieved or sustained in different environment. In order to improve the gap of between different workflows, this paper discusses the effect of the interruption and how to assist the decision making for selecting appropriate action by considering the context of the interruption.

2 Consideration of Interrupted Work

In order to simplify the context of the interruption, the document submission in the office work is assumed. The workflow of the document submission work can be classified the hierarchical layers by the span of the workflow such as monthly, weekly, daily and so on. Figure 1 shows the relation of each layer workflow and its contents. The upper layer workflow consists of the detailed workflows and the order number for operating detailed workflows is assigned. Operating time and limited time in the workflow of this situation are parameter as criteria for defining the context of each achievement. The operating time indicates the consumed time to execute each task on the workflow such as document creation time, correcting information time, thinking time or communication time. The limited time is predetermined for the objective or the rule based on enterprise strategy by each task. Each task is ordered appropriately according to the workflow to complete the tasks by time limit. The negative interruption in this situation means consuming the operating time to exceed the time limit. The recalling time for the progress of the past task before the interruption is included into this negative effect in the time aspect on the workflow. The positive interruption in time parametric view means extending usable time for the limited time of the workflow. The usable time is extended by shorten the operating time and to extend the time limit. The positive interruption for the tiredness that generates the delay operation improves the usable time by eliminating the extending factor of the operating time such as resting or refreshing. The negative interruption for

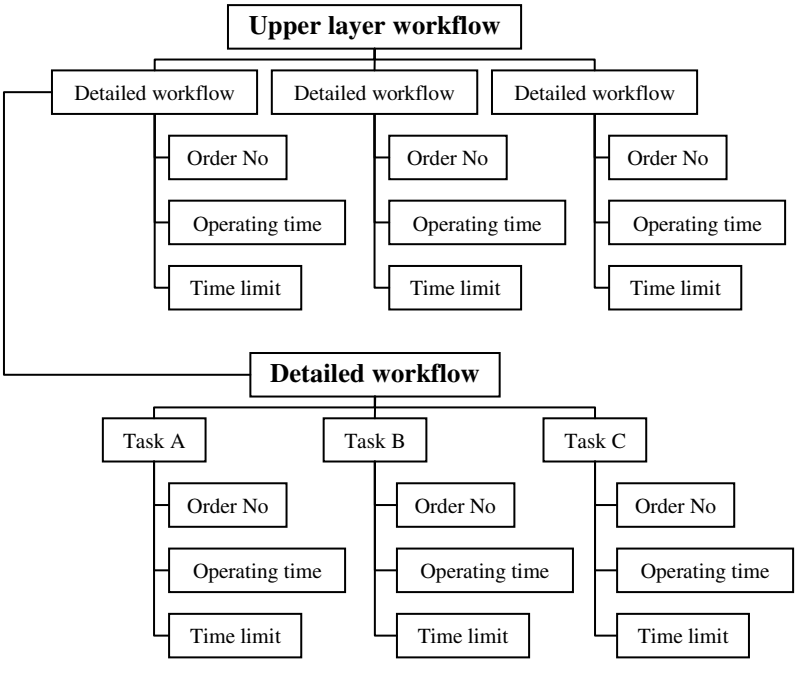


Fig. 1. The relation of the workflows

the detailed workflow influences the other detail workflow to advance the upper layer workflow. If the needed time to operate each task is shorten by the negative interruption, the schedule of the surrounded workflows must be adjusted by skipping remaining tasks, extending the limited time of the active detailed workflow or sacrificing the time and tasks of the other detailed workflow.

For example, the work for monthly document submissions is assumed as the upper layer workflow. This workflow has three sub workflows that are to create daily report, working time report and reimbursement form. Each sub-workflow has following parametric condition based on defined workflow in assumed situation and each these workflows have each some tasks and order. The time limit of the daily report is 17:30 and the estimated operating time for the creation time is two and a half hours. The creation time of the working time report and the reimbursement form is a half hour. Both time limits for these two works are 13:00. In order to complete all works from 8:30 to 12:00 that is the starting time for lunch break, these workflows must be scheduled. Firstly the daily report is created for reference from the working time report and the reimbursement form. After finished first workflow, the other workflows are stared in random order. If the operation for creating the daily report is interrupted by meeting period of thirty minutes, the objective of the upper layer workflow cannot be achieved because of consuming enough time for operation. In order to resolve this issue, previous workflows must be changed. For example, the operating time is shorten by adjusting the scheduled tasks, the starting time for daily report is moved to the afternoon, the operating time is extended by using the time for lunch break or demonstrating requisition for extending the time limit.

3 Structure for Supporting Decision Making

When the situation of the active workflow is changed by the interruption, the similarity of the situation between the operating workflow and the effected environment is reduced by interruption. The objective of the workflow may not be achieved by operating with mismatch workflow in restarting the interrupted work. This issue can be resolved by recommending similar expertise workflow from knowledgebase. Figure 2 shows the proposed system for assisting appropriate adaptive workflow to actual situation. The active workflow is progressed from process 1 (A-P1) to process 3 (A-P3) on the workflow A (WA). The workflow B (WB) is started after completed process 4 (A-P4) in prescheduled order as an upper layer workflow. The workflow C is interruption that has positive and negative effect to other workflows. The system observes and retrieves the changed situation in time aspect among the workflows. In order to recommend the appropriate action from the similar workflow to the recent situation, the similarity of the interruption context is computed. That similarity for the interruption context is computed by applying regularized Singular Value Decomposition (SVD) [6]. If the interruption that has negative impact to the actual workflow on the context, the appropriate workflow is reasoned according to the similarity computation from the expertise workflow. That expertise workflow is contained to the knowledge database with classifying case studies collected from enterprise experiences. The context of the interruption is

categorized for classify the similar case in the knowledgebase by giving weight to the important factor based on the expertise. The worker's situation in interruption is used as input data to compute the similarity of the adaptive scenario according to the weight. The ranking of the similar scenario is recommended from the system for supporting worker's decision making. The weight is adjusted dynamically for appropriate decision making by repeating evaluation in the recommendation. The cognitive map model for the interaction of the interruption effect improves the accuracy to reason the similarity by applying the relation of the causal and effect. The set of the scenario policy is recommended from the system by estimating the reasoning with actual situation and similar situation. This recommendation from knowledgebase assists for human decision making on the interruption. The adjusted new process can be created by merging simultaneous workflow and aligning the cognitive map as multi criteria. The similarity of the active workflow and the other workflow is not changed by the positive interruption to the workflow because the positive interruption sustains the original workflow even the situation is changed. Therefore, the workflow of the positive interruption (positive workflow D) is added to similar situation workflow as a recommended relevant task through the feedback to the knowledge database.

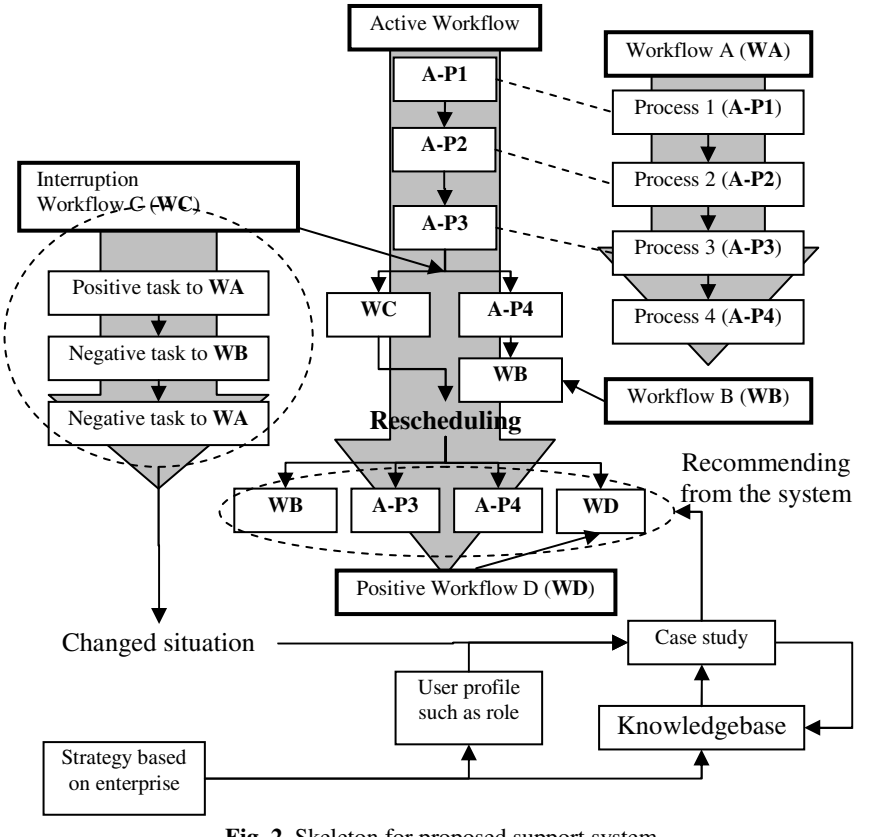


Fig. 2. Skeleton for proposed support system

4 Conclusion and Future Works

Interruption is encountered regular activities and it is occurred frequently. A beginner worker has less experience on decision knowledge to resume workflow after an interruption. Even if the work of an expert worker is interrupted, the accuracy of the decision making for appropriate action is reduced according to the context of the interruption or the worker's negative condition such as reducing memory. The decision based on small amount estimation invokes the gap of the upper layer workflow. For example the total limited time is reduced by concentrating the detailed workflow. The expert knowledge and experience is needed to estimate appropriate next action in dynamic changed situation by the occurring interruption. The proposed approach is an improvement for the changed situation of the workflow by utilizing knowledgebase. The knowledgebase is used to narrow the target area to retrieve adaptive information for the worker by reasoning similarity between a worker's profile and relevant contents in the interruption. The knowledgebase can have the criteria for improving the efficiency of the interrupted workflow by computing the relative appropriate knowledge relevant to the need of workers. Therefore the worker can get the adaptive information for the interruption of the original workflow from the system without contriving ways to request adaptive information from the knowledgebase. Although this paper focused on the time parametric view, there are several types for the context of the interruption such as mentality of the workers or physical issues related to the workflow. The degree of the effect by the interruption differs greatly in individuals. In order to improve the usability of the recommendation for changed workflow, the system must include the interaction of several contexts and retrieve individual parameter for adjusting similar workflow. Interruption that has a criticality is discussed for the purpose of improving work efficiency. However, a non-critical interruption may important for other workflow or individual mental satisfaction. In forthcoming paper we would formalize interruption theory model and present concrete results as case studies in real business.

Moreover, in the near future, this study will consider further attributes to expand the interruption context by retrieving many case studies collected from real enterprise. The balance of selecting these attributes depends on the priority raking of these attributes that we need to investigate and report in next paper.

References

1. Prentice, W.C.H.: The Psychological Review - The Interruption of Tasks. Psychological Review 51(6) (November 1944), http://interruptions.net/literature/Prentice-PsychologicalReview44.pdf
2. Wherry, R.J.: The Curve of Forgetting: Its Statistical Application. Journal of Educational Psychology 23(8), 621–624 (1932), http://psycnet.apa.org/index.cfm?fa=buy.optionToBuy&id=1933-01266-001, doi:10.1037/h0070645

3. Barrouillet, P., Bernardin, S., Portrat, S., Vergauwe, E., Camos, V.: Time and Cognitive Load in Working Memory. Journal of Experimental Psychology: Learning, Memory, and Cognition 33(3), 570–585 (2007)
4. Sugawara, K., Fujita, H.: An Approach for Smoothly Recalling the Interrupted Tasks by Memorizing User Tasks. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 21–26. Springer, Heidelberg (2011)
5. Sugawara, K.: Approach for Handling Multiple Objects Without Breaking the Continuity of User's Thinking. In: Fujita, H. (ed.) Proceedings of the 9th SoMeT 2010, New Trends in Software Methodologies, Tools and Techniques, pp. 218–225 (2010)
6. Turney, P.D.: Similarity of Semantic Relations. Journal Computational Linguistics 32(3) (September 2006)

Action Selection Using Theory of Mind: A Case Study in the Domain of Fighter Pilot Training

Mark Hoogendoorn[1] and Robbert-Jan Merk[2]

[1] VU University Amsterdam, Department of Computer Science
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
m.hoogendoorn@vu.nl
[2] National Aerospace Laboratory, Training, Simulation, and Operator Performance
Anthony Fokkerweg 2, Amsterdam, The Netherlands
robbert-jan.merk@nlr.nl

Abstract. Theory of mind based reasoning is crucial for humans that interact with each other. Also in the domain of multi-agent systems the importance of theory of mind based reasoning has been stressed, for instance in the process of selecting appropriate actions. In this paper, a theory of mind based approach is presented which goes beyond the capabilities of currently existing agent-based theory of mind approaches by adding certainties to predicted states, and predicting over a longer period of time thereby generating multiple predictions using the theory of mind model. This approach has been applied to the domain of fighter pilots whereby intelligent opponents are developed to facilitate dedicated training for F16 fighter pilots.

1 Introduction

In multi-agent system, the problem of individuals selecting an appropriate action given the current state of affairs of the world and the behavior of other agents is a far from trivial task. Agents have to consider the actions that will most likely be performed by other agents, and select their own action accordingly. Castelfranchi [5] states that agents can take models of other agents actions into account in various ways: it can be used to anticipate to the behavior of the other agent (for instance exploiting the other agent's action) or it can be used to influence the other agents behavior (for instance inducting the other agent to drop a certain goal). Imagine a scenario whereby there are two fighter planes that are flying towards each other, called red and blue respectively. If red anticipates that blue will fire a missile, red might take precautions. On the other hand, if red knows that performing a certain maneuver will likely avoid the missile from being fired by blue he might perform that maneuver.

A variety of authors have developed models that try to come to coordinated efforts. For instance, Parker [9] introduces an approach that takes into account previous actions performed by other agents. Veloso, Stone, and Bowling [10] present an algorithm to anticipate on the needs of teammates in RoboCup Soccer and select actions accordingly. However, these approaches do not have an explicit representation of the

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 521–533, 2012.

behavioral model and internal decision process of the other agents. Humans are known to use such behavioral model to coordinate their behavior, which is commonly referred to as *theory of mind* (see e.g. [2]). Within several research domains, *theory of mind* models have been utilized to obtain good coordinated effort between agents. In the (serious) gaming domain for instance, various approaches have been proposed (see e.g. [8; 6; 7]).

The main purpose of this paper is to go beyond the current state of the art in such theory of mind based approaches and to apply this approach to agent-based training environments of fighter pilots. To be more specific, in this paper an action selection mechanism is introduced based upon theory of mind that extends the approaches mentioned above by: (1) the ability to express a BDI-based behavioral model of the other agent with explicit certainties of the states (modeled by means of numerical values); (2) is able to look ahead for a large number of steps, thereby combining the *theory of mind* model with a world model as well as the prospected actions of the agent itself, and (3) provides a thorough evaluation of the approach in the domain of fighter pilots, whereby agents are developed as part of a training environment for F16 fighter pilots and can act as an intelligent opponent.

This paper is organized as follows. The modeling approach used throughout this paper is explained in Section 2. Section 3 expresses the fighter pilot case study. In Section 4 the model itself is explained whereas in Section 5 the results obtained using the proposed model are presented. Section 6 presents the verification of these results, and finally, Section 7 is a discussion.

2 Modeling Approach

In order to model the approach as introduced an expressive language is used, called LEADSTO [4], based on a variant of Executable Temporal Logic [1]. The basis of this language consists of causal relations of the format $\alpha \rightarrow\!\!\!\rightarrow_{e, f, g, h} \beta$, which means:

if state property α holds for a certain time interval with duration g,
then after some delay (between e and f) state property β will hold for a certain time interval of length h.

where α and β are state properties of the form 'conjunction of literals' (where a literal is an atom or the negation of an atom), and e, f, g, h non-negative real numbers. In order to evaluate the proper function of the model, the language TTL is used [3]. This predicate logical language supports formal specification and analysis of dynamic properties, covering both qualitative and quantitative aspects. TTL is built on atoms referring to states, time points and traces. A *state* of a process for (state) ontology Ont is an assignment of truth values to the set of ground atoms in the ontology. The set of all possible states for ontology Ont is denoted by STATES(Ont). To describe sequences of states, a fixed *time frame* T is assumed which is linearly ordered. A *trace* γ over state ontology Ont and time frame T is a mapping $\gamma : T \rightarrow$ STATES(Ont), i.e., a sequence of states γ_t ($t \in T$) in STATES(Ont). The set of *dynamic properties* DYNPROP(Ont) is the

set of temporal statements that can be formulated with respect to traces based on the state ontology Ont in the following manner. Given a trace γ over state ontology Ont, the state in γ at time point t is denoted by state(γ, t). These states can be related to state properties via the formally defined satisfaction relation $\models$, comparable to the Holds-predicate in the Situation Calculus: state(γ, t) $\models$ p denotes that state property p holds in trace γ at time t. Based on these statements, dynamic properties can be formulated in a formal manner in a sorted first-order predicate logic, using quantifiers over time and traces and the usual first-order logical connectives such as $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\forall$, $\exists$. A special software environment has been developed for TTL, featuring both a Property Editor for building and editing TTL properties and a Checking Tool that enables formal verification of such properties against a set of (simulated or empirical) traces.

3 Case Study

As this paper is part of a larger research effort to generate more human-like agents to improve the quality of tactical training simulators for the Royal Netherlands Air Force, the model is tested against a case study in the domain of air-to-air combat.

In the case study, two aircraft of opposing forces fly their missions. One aircraft, flown by a human, is designated as Blue. The opponent, designated Red, is controlled by an agent equipped with the theory of mind model as developed in this paper. Both aircraft have radars to detect each other. In the case study, simple radar models have been used that always detect other aircraft in a 360^0 area around the aircraft within an unlimited distance. Blue and Red have different missions: Blue has a defensive mission while Red has an offensive mission. The goal of Blue is to defend its airspace, while Red's goal is to get past Blue to its objective. Both Blue and Red want to survive and to kill the opponent if possible. However, Blue has orders on how and when to attack any detected aircraft expressed by means of so-called Rules of Engagement (RoE). RoE are used to reduce the chance of friendly fire or other incidents.

An important aspect of the scenario is the Forward Line of Own Troops or FLOT. The FLOT basically is a line dividing the friendly territory from enemy territory. The Rules of Engagement of Blue dictate that Blue can only attack aircraft that cross the FLOT and come to Blue's side of the FLOT. It can be reasonably assumed that such aircraft are hostile and thus valid targets for an attack. Blue's behavior, to a large extent, is therefore determined by his RoE which can therefore be used as a basis for the mental model that Red uses to make predictions on the other agent's actions. It will not attack Red if Red remains on the far side of the FLOT and only attacks if Red crosses the FLOT.

4 Model for Action Selection Using Theory of Mind

The Action Selection model which utilizes theory of mind based reasoning consists of a number of processes, submodels and states, as illustrated in Figure 1.

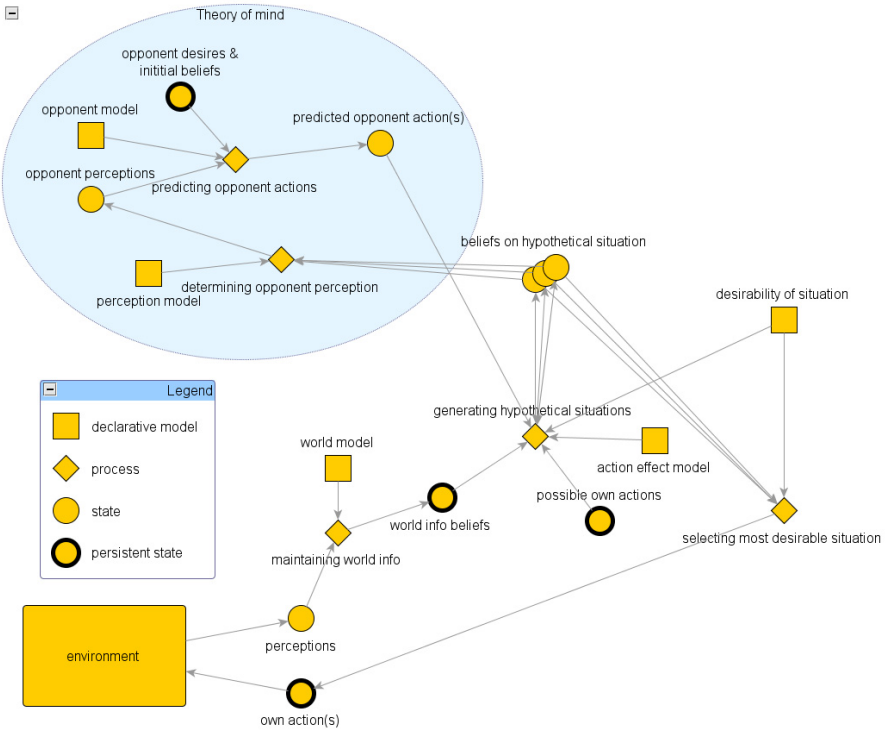


Fig. 1. Overview of model

Essentially, the decision cycle of the agent starts with the updating of its current beliefs about the world which is performed by means of the process *maintaining world info*. Thereafter, this information is used to determine what situations the agent could end up in after this situation, given his own, and the other agents' actions and their influence upon the world, done in the process *generating hypothetical situation*. Then out of these hypothetical situations, the most desirable is selected in the process *selecting most desirable situation*. Once this most desirable situation is selected, the action accompanying this situation is performed.

The *theory of mind module*, the key element of the model presented in this paper, is used to generate the hypothetical situations. As the opponent has such a large impact on these hypothetical future situations, the theory of mind module is essential to predict his actions and thus his influence on the hypothetical situations. The search through the state space of potential situations is guided by a desirability function, which indicates how good a situation is from the viewpoint of the agent. As the agent has to operate in real time, the generation of new hypothetical situations is halted after a certain time limit has been reached. Below, first the three processes high-level processes are explained whereby in the generation of hypothetical situation the Theory of Mind module is explained in more detail.

4.1 Maintaining World Info

The process of action selection starts with building up the beliefs on the world state using observations. The model receives from the environment periodically a set of predicates that represent the perception of the agent. These percepts are modeled as object-attribute-value triples, so that there is a uniform, expressive representation of the outside world available to the agent. An example is that if the speed of the agent's aircraft is 100 m/s, the agent receives the predicate observation(red, speed(ownship, 100)).

From these percepts a set of viewpoint-independent world info beliefs are derived, using a world model. These beliefs are of the form belief(AGENT, WORLD_INFO), where AGENT is an identifier referring to whom the belief belongs to and WORLD_INFO an object-attribute-value triples, like with perception. An example of a belief of the agent red on world info is the position of an object, represented as belief(red, position(Object, X, Y)), with X and Y being the absolute coordinates of the object. The world model is a set of rules that dictate how certain world info elements can be derived from percepts and other world info elements.

4.2 Generating Hypothetical Situations

To select a new action, as many hypothetical situations as possible are generated within the time limit that is set. Formally this can be expressed as (note that in case the *e,f,g,h* values accompanying the LEADSTO arrow are omitted, standard values of 0,0,1, and 1 are assumed):

LP1: As long as the time limit has not been reached, the process continues
A:AGENT, D:REAL, T:REAL, Tmax:REAL
desire(A, generating_situations) ∧ belief(A, time_since_start(T)) ∧ belief(A, maximum_time_for_generating(Tmax) ∧ T < Tmax
→→ desire(A, generating_situations)

A hypothetical situation is a predicted world state based on the actions of the relevant actors and the predicted influence of those actions on the world state. For example, the position of an aircraft in a future hypothetical situation depends on its previous position, its speed, its heading and the action of its pilot (does he turn left or right, etcetera). A hypothetical situation is identified by an ordered list of the actions of each of the agents involved. For example, a possible hypothetical situation is represented by the predicate situation([transition([action(red, turn_left), action(blue, turn_right)])], which represents the hypothetical situation after Red turns left and Blue turns right one time step from the current moment. For each hypothetical situation a set of beliefs is formed on the state of the world in that situation, which can then be used to determine the desirability of this situation. These are the same kind of beliefs as used to represent the current, real world state but their actual values are interpolated using an *action effect model*. This model is a set of rules that describe how a certain belief on world info changes, given a previous situation and an action that is performed. The beliefs that are derived are of the form belief(AGENT, holds_in(SITUATION, WORLD_INFO)), meaning that AGENT believes that WORLD_INFO, an object-attribute-value triplet, holds in SITUATION.

Note that not all hypothetical situations are formed (i.e. every possible combination of sequences of actions), as that would lead to combinatory explosion. Instead, the generation of (beliefs on world info for) hypothetical situations is reduced by predicting the opponent's action and by selecting only the most desirable hypothetical situation to continue reasoning with.

Starting with the current situation, situation([]), the "hypothetical" situation that is the result of zero transitions, the first step is to determine the opponent's action, which is predicted via the theory of mind module as explained below. The action that follows from the theory of mind module is then coupled with every possible action of the agent itself and for each of these possible actions, a new hypothetical situation is considers and the beliefs on its world info are derived. More formally:

LP2: generating hypothetical situations from the current situation
A:AGENT,A_{own}:ACTION, A_{enemy}:ACTION, Info:WORLD_INFO, S_0:SITUATION, S_1:SITUATION
desire(A, generating_situations) $\wedge$ belief(A, possible_actions(A_{own})) $\wedge$ belief(A, current_situation(s_0)) $\wedge$
belief(A, predicted_enemy_action(s_0, A_{enemy})) $\wedge$ set_of(A_{own}, A_{enemy}, Set) $\wedge$
belief(A, action_effect(S_0, Set, world_info(Info))) $\wedge$ list_of(S_0, Set, S_1)
$\twoheadrightarrow$ belief(A, holds_in(S_1, world_info(Info)) $\wedge$ belief(A, considered_hypothetical_situation(S_1))

Each hypothetical situation S for which beliefs on world info are derived is stored as belief(A, considered_hypothetical_situation(S_1)). The next step is to derive what the opponent's action would be (using the theory of mind module explained in Sections 4.2.1 and 4.2.2. This results in a single action from the opponent. Eventually, after the first iterations, there are N + 1 considered hypothetical situations, where N is the number of possible actions for the agent. Each of these hypothetical situations is assigned a desirability value in the range [0, 1]. In each subsequent iteration, the most desirable considered hypothetical situation that has not been selected before is selected. For this selected situation, N more hypothetical situations are generated for each possible action of the agent, the opponent action in the selected situation is predicted using the module again, and the world info beliefs resulting from these actions are derived. The selected situation is marked so that it will not be selected for generating more situations anymore. More formally:

LP3: Generating further hypothetical situations from other hypothetical situations
A:AGENT A_{own}:ACTION, $A_{opponent}$:ACTION, I:WORLD_INFO, S_n:SITUATION, S_{n+1}: SITUATION
desire (A, need_for_reasoning) $\wedge$ belief(A, considered _hypothetical_situation(S_n)) $\wedge$
belief(A, most_desirable(S_n)) $\wedge$ not(belief(A, has_been_selected(S_n))) $\wedge$ belief(A, possible_action(A_{own})) $\wedge$
belief(A, predicted_opponent_action(s_n, $A_{opponent}$)) $\wedge$ set_of(A_{own}, $A_{opponent}$, Transition) $\wedge$
action_effect(S_n, Transition, world_info(I)) $\wedge$ list_of(S_n, Transition, S_{n+1})
$\twoheadrightarrow$ belief(A, holds_in(S_{n+1}, world_info(I)) $\wedge$ belief(A, considered_hypothetical_situation(S_{n+1})) $\wedge$ belief(A, has_been_selected(S_n))

This process of selecting the most desirable considered hypothetical situation, generate it successor hypothetical situation by combining all possible own actions and predicting the opponent action and derived world info beliefs continues until the time limit is reached and desire (A, need_for_reasoning) no longer holds.

Below, it is explained how the theory of mind model works in order to derive the actions of the opponent.

4.2.1 Determining the Opponents Observations

In order to determine the observations of the opponent, the viewpoint-independent world info beliefs for the given hypothetical situation are converted to viewpoint-dependent beliefs on opponent perception using the rules of the perception model. An example of such a rule is to derive a relative position of an object (is it to the left or right of the opponent) from the beliefs on the coordinates of that object.

As the beliefs on the opponent mental states have a degree of belief value assigned to them, the beliefs on opponent perception needs to be assigned such degree of belief values. While more complex functions can be used for this assignment, for now the assignment method is simple: beliefs on opponent perceptions that hold are assigned the value 1, all others the value 0.

4.2.2 Determining the Opponents Actions

On the basis of the estimated opponent's perceptions and the opponent model, the various mental states of the opponent can be predicted. For the time being, the focus is on predicting the actions the opponent will take in a given hypothetical situation.

The beliefs on the opponent's perceptions have a degree of belief as described above. The degrees of belief on the opponent's mental states are then calculated via the causal dependencies between the mental states. The degree of belief is a real value in the range [0, 1] indicating how certain the agent is the opponent has these beliefs.

The *opponent model* is a causal model on what kind of mental states the opponent can have and what the causal relations are between these mental states. This model is represented in the agent as a set of beliefs. These beliefs represent the knowledge the agent has on how the opponent thinks and reasons.

The beliefs on mental states are of the form belief(A:AGENT, holds_in(S:SITUATION, M:MENTAL_STATE), D:REAL), where A is the agent having this belief, M the mental state the belief is about, S the hypothetical situation the mental state holds and D the degree of belief. The model follows the Belief-Desire-Intention framework, so the mental states the beliefs refer to are either beliefs, desires or intentions. The terms of type MENTAL_STATE have either the form belief(A:AGENT, I:WORLD_INFO), the form desire(A:AGENT, D:DESIRE), the form intention(A:AGENT, I:INTENTION) or the form action(A:AGENT, C:ACTION), where DESIRE, INTENTION and ACTION are the sets of respectively the possible desires, possible intentions and possible actions agent A can have.

To represent the knowledge the agent has about the interaction between mental states in other agents or humans, the agent has a set of beliefs on causal relations between mental states. As with mental state beliefs, these beliefs consist of three parts, with the first part indicating the owner of the belief and the last part the degree of belief. The second part differs in that it consists of a pair of mental states that are supposed to be causally linked, with the first mental state being a cause to the second mental state. An example of such a belief on causal relation is belief(A, causal_relation (desire(B, engage), intention(B, engage)), 0.5), which represents A's belief with a degree of belief of 0.5 that if B desires to engage, he will have the intention to engage.

Using the degrees of belief of the beliefs on the opponent's observations and of the causal relations between the opponent's mental states, the degrees of beliefs of the opponent's mental states for a given hypothetical situation can be calculated with (1):

$$dob\ (S, B) = \frac{1}{1 + e^{-\sigma (W_{S,B} - \tau)}} \quad (1)$$

In formula (1), *dob(S,B)* is the degree of belief B in situation S, σ is the steepness parameter, τ is the threshold parameter. $W_{S,B}$ is calculated with formula (2).

$$W_{S,B} = \sum_{i=1}^{N} dob\ (c_{i,B})\, dob\ (S, i) \quad (2)$$

Where N is the number of beliefs that cause belief B, $dob(c_i, b)$ is the degree of belief of the causal relation from belief i and belief B and $dob(i)$ the degree of belief of i.

With every generation of a new hypothetical situation, the beliefs on opponent observations are derived from the world info beliefs of that situation, from which the mental states they share a causal relation with are derived using formula (1). After that, all beliefs on mental states with a causal link with the derived beliefs on mental states are updated, and so on, after all beliefs on mental states in the opponent model of are derived.

At the end of the causal chains defined by the opponent model are the beliefs on the opponent actions. The action with the highest degree of belief is selected and marked as the predicted opponent action for that particular hypothetical situation.

4.3 Selecting Most Desirable Situation

As mentioned in the part explaining *generating hypothetical situations*, each hypothetical situation is assigned a desirability value, which is used to guide the search process. This desirability value represents the agent's subjective appraisal of a certain (hypothetical) situations and consists of both rational and emotive aspects. Also, at the end of the reasoning process, when the time limit has been reached, the most desirable hypothetical situation and the first agent action in the list of transitions that establishes this hypothetical situation is then executed, so that the current situation is nudged a bit towards this highly desirable hypothetical situation. After this the whole process starts again with maintaining world info, generating hypothetical situations in which opponent actions are predicted and so on.

Selecting the most desirable hypothetical situation is simply done by taking the one with the highest desirability value. Calculating the desirability value is done with (2):

$$d(S) = \sum_{i=1}^{N} w(des_i)\, d(S, des_i) \quad (2)$$

d(S)	Total desirability of situation S
$d(S, des_i)$	Desirability of situation S in regards to desire des_i
$w(des_i)$	Weight for desire des_i
N	number of desires

The idea is that the desirability of a situation is fully dependent on what the agent's desires are and thus that this desirability can be divided into multiple values, one for

each desire, that are then combined using a weighted summation to get a final value in the range [0, 1].

The exact number and nature of the desires is domain specific, as not every possible human desire is relevant for each application. For example, the desire for survival plays a large role in the case study of this paper, but in other (non-military) domains, it might be advisable to choose to model other desires.

5 Results

The model described in this paper has been implemented and tested in a tactical fighter simulator for the use case described in Section 3. Hereby the setting was that the automated agent using the theory of mind module controlled red which flew against a human-controlled aircraft, designated Blue. The exact specification of the Theory of Mind model utilized can be found in appendix A[1]. To both test the added value of the theory of mind aspect of the model and its robustness, four different configurations of the use case have been used. In configuration A, the human pilot is instructed to comply to the mission briefing, i.e. respect the Rule of Engagement of only attacking Red when it crosses the FLOT and to not cross the FLOT himself. Also, the theory of mind aspect is inactive; instead the agent assumes in every hypothetical situation that Blue does not turn and fires a missile whenever Red is in Blue's weapon range. In configuration B, the theory of mind aspect is inactive as well, but the human pilot is instructed to not comply to the mission briefing, i.e. to ignore the FLOT and directly engage Red. In configuration C and D the theory of mind aspect is made active in order to predict opponent actions, with the human pilot being compliant in configuration C and non-compliant in D.

5.1 Parameter Settings

In the opponent model, the reasoning of the human pilot regarding the FLOT, his RoE and other factors have been modeled. For the sake of brevity the details of the opponent model has been left out of the paper.

For the use case, three desires have been defined: desire(red, engage), desire(red, survive) and desire(red, objective). The weights for the desires are: W(desire(red, engage)) = 0.1, W(desire(red, survive)) = 0.3, W(desire(red, objective)) = 0.6.The values for the desirability for each desire are calculated using the threshold function $th(\sigma,\tau, V)$, which is defined as:

$$th(\sigma,\tau,V) = \left(\frac{1}{1+e^{-\sigma(V-\tau)}} - \frac{1}{1+e^{\sigma\tau)}}\right)(1+e^{-\sigma\tau)}) \tag{3}$$

- σ Steepness parameter
- T Threshold parameter
- V Input value

[1] http://www.cs.vu.nl/~mhoogen/tom/appendix_iea_aie_2012.pdf

Below the functions for the three desires used for the test scenario are described.

Desire to engage: Red has the desire to engage enemies. The closer Red is to its opponent and the more the opponent is in front of Red, the better the situation is for Red with regards to its desire to engage. The detailed formula used to calculate this value is expressed as follows:

$$d(S, engage) = r(1 - th(\sigma_d, 0.5\tau_d, V_d)) + (1 - r)(1 - th(\sigma_b, 0, V_b)) \quad (4)$$

- r The ratio between how the distance and the bearing influence the desirability.
- V_d The distance between Blue and Red, derived from the world info on the positions of Blue and Red.
- τ_d The max distance from Blue where it is still possible for Red to reach Blue.
- σ_d $= 10 / \tau_d^2$
- V_b The number of degrees between Red's nose and Blue's position (bearing), based on Red's belief on world info.
- σ_d $= 10/180$[3]

Desire to fulfill objective: As Red has the mission to get to an objective, the closer Red is to that objective, the higher the desirability to fulfill objective. The detailed formula is shown in Appendix A.

Desire to survive: As an agent mimicking human behavior, Red has a desire to survive. Blue can kill Red by means of missiles, so it is desirable for Red to have the distance between it and any missile Blue fires as large as possible. If no missile has been fired, Red is save in that situation. Again, the details are shown in Appendix A.

5.2 Behavior

Figure 2 shows the behaviour of Red and Blue in the four configurations.

In each configuration, Red starts north of the FLOT (the horizontal line in the middle) and flies southeast towards the objective south of the FLOT. Blue starts to the south of the FLOT, facing westward while circling clockwise to patrol the area between the FLOT and the objective.

In configuration A, Red moves away from Blue as soon as Red gets into Blue's weapon range. As Blue is compliant and thus does not cross the FLOT, both aircraft hover around the FLOT. In configuration B, Blue heads straight for Red, who avoids Blue as without theory of mind it will not assume that Blue will not attack him.

The behavior of Red with the theory of mind aspect activated behaves markedly different, as the agent will not see Blue as a threat while being on the north side of the FLOT. In configuration C, Red only starts moving away from Blue when Red crosses the FLOT. Later on, nearer to the objective, Red decides to engage Blue who, after

[2] The steepness parameter needs to be divided by the threshold parameter as keeping the steepness at 10 would mean the threshold function is very flat.

[3] 180 degrees is the maximum deviation from an aircraft's nose.

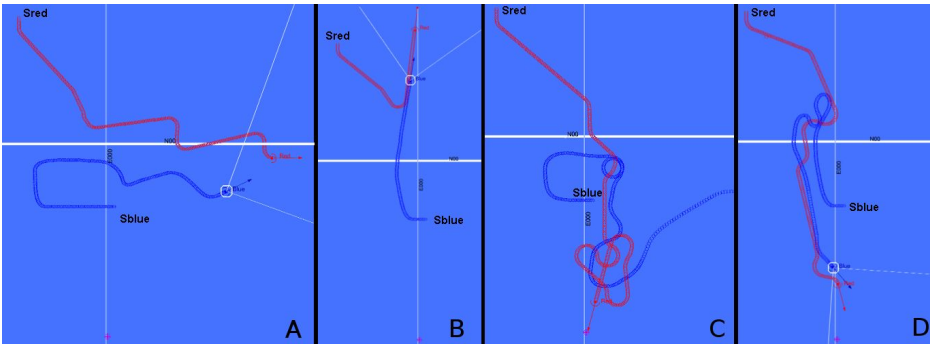


Fig. 2. Flight paths of Red (agent) and Blue (human) in different configuration. White horizontal line is the FLOT, Red's objective is at the bottom of the white vertical line, S_{red} and S_{blue} indicating the starting positions of Red and Blue.

A: no theory of mind, compliant opponent.
B: no theory of mind, non-compliant opponent.
C: theory of mind, compliant opponent
D: theory of mind, non-compliant opponent.

Red gets successfully behind him a few times, flies away to the west, after which Red continues to the objective. In configuration D, Red once again sees no threat in Blue north of the FLOT and manages to get a lead in the race towards the objective, with an evasive maneuver at the end.

These examples show that using the theory of mind module contributes to a more intelligent behavior of the agent, as 1) the agent gets much closer to the objective while using the theory of mind module than without using it and 2) the agent reacts more appropriate to Blue's behavior.

6 Verification

In this section, it is investigated whether the model indeed shows the behavior as intended in the design of the model. These properties are used to verify the internal correctness of the model For the sake of brevity, not all formalizations are shown.

P1: Updated world model lead to predicted hypothetical situations
In case the agent obtains an update with respect to the world model, then hypothetical situations are generated using the theory of mind module. Formally:

$\forall\gamma$:TRACES, t:interval
[state(γ, t) |= world_model_update
$\Rightarrow \exists$t2:interval $\geq$ t [(state(γ, t) |= hypothetical_situation_generated) &
$\forall$t':interval > t & t' $\leq$t2 [state(γ, t) |$\neq$ world_model_update]]]

P2: Hypothetical situation lead to action being selected
Once a number of hypothetical situations have been generated, eventually an action will be selected by the agent.

P3: Actions lead to updated world model
Once a new action has been selected, the agent will receive an updated set of perceptions.

All the properties expressed above were formally verified against the traces generated in the simulation runs (i.e. the four traces representing the different settings), and were shown to hold.

7 Discussion

In this paper, a theory of mind based model has been developed which can be utilized to create intelligent agents that can anticipate on the actions of other agents. Hereby, several aspects were represented in the model that move beyond the current state of the art in agent-based theory of mind modeling: the model about which reasoning takes place incorporates certainties, and the model is used to reason ahead a number of steps. The model has been applied to the domain of fighter pilots, thereby showing that it can truly contribute to more intelligent behavior.

Future work is to create a more elaborate evaluation of the approach by means of real fighter pilots, to see how they judge the intelligence of the behavior of the agents. Furthermore, the idea is also to make a coupling to more sophisticated models for instance to represent the agent's awareness of the situation Finally, applications in different domains are also envisioned.

References

1. Barringer, H., Fisher, M., Gabbay, D., Owens, R., Reynolds, M.: The Imperative Future: Principles of Executable Temporal Logic. John Wiley & Sons (1996)
2. Baron-Cohen, S.: Mindblindness. MIT Press (1995)
3. Bosse, T., Jonker, C.M., van der Meij, L., Sharpanskykh, A., Treur, J.: Specification and Verification of Dynamics in Agent Models. International Journal of Cooperative Information Systems 18, 167–193 (2009)
4. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by Simulation. International Journal of Artificial Intelligence Tools 16, 435–464 (2007)
5. Castelfranchi, C.: Modelling social action for AI agents. Artificial Intelligence 103, 157–182 (1998)
6. Harbers, M., van den Bosch, K., Meyer, J.J.: Modeling Agent with a Theory of Mind. In: Baeza-Yates, R., Lang, J., Mitra, S., Parsons, S., Pasi, G. (eds.) Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Agent Technology, pp. 217–224. IEEE Computer Society Press (2009)
7. Hoogendoorn, M., Soumokil, J.: Evaluation of Virtual Agents Attributed with Theory of Mind in a Real Time Action Game. In: van der Hoek, Kaminka, Lesperance, Luck, Send (eds.) Proceedings of the Ninth International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2010, pp. 59–66 (2010)

8. Laird, J.: It Knows What You're Going To Do: Adding Anticipation to a Quakebot. In: Andre, E., Sen, S., Frasson, C., Muller, J.P. (eds.) Proceedings of the 5th International Joint Conference on Autonomous Agents, pp. 385–392. ACM Press (2001)
9. Parker, L.E.: Adaptive action selection for cooperative agent teams. In: Meyer, J.-A., Roitblat, H., Wilson, S. (eds.) Proceedings of the 2nd International Conference on Simulation of Adaptive Behavior, pp. 442–450. MIT Press, Cambridge (1992)
10. Veloso, M., Stone, P., Bowling, M.: Anticipation as a key for collaboration in a team of agents: A case study in robotic soccer. In: Proceedings of SPIE Sensor Fusion and Decentralized Control in Robotic Systems II, vol. 3839 (1999)

Intelligent Decision-Making Approach Based on Fuzzy-Causal Knowledge and Reasoning

Alejandro Peña-Ayala[1,2,3] and Riichiro Mizoguchi[3]

[1] WOLNM,
31 Julio 1859, No. 1099-B, Leyes Reforma, 09310, DF, Mexico
[2] ESIME-Z IPN
apenaa@ipn.mx
http://www.wolnm.org/apa
[3] Osaka University, Institute of Scientific and Industrial Research, Suita Campus
miz@ei.sanken.osaka-u.ac.jp

Abstract. Our intelligent decision-making approach (IDMA) is an instance of cognitive computing. It applies causality as common sense reasoning and fuzzy logic as a representation for qualitative knowledge. Our IDMA collects raw knowledge of humans through psychological models to tailor a knowledge-base (KB). The KB manages different repositories (e.g., cognitive maps (CM) and an ontology) to depict the object of study. The IDMA traces fuzzy-causal inferences to simulate causal behavior and estimate causal outcomes for decision-making. In order to test our approach, it is linked to the sequencing module of an intelligent and adaptive web-based educational system (IAWBES). It is used to provide student-centered education and enhance the students' learning by intelligent and adaptive functionalities. The results reveal users of an experimental group reached 17% of better learning than their peers of the control group.

Keywords. Fuzzy-causal reasoning, psychological models, decision-making, knowledge-base, cognitive map, ontology, student model, content model.

1 Introduction

Cognitive computing aims at developing a coherent mechanism inspired by the mind's capabilities [1]. Moving from ignorance to knowledge, dealing with uncertainty, problem solving, and decision-making are a sample of cognitive reasoning [2]. Causality, our awareness of what causes what in the world and why it matters [3], represents a baseline for predicting and explaining certain consequences given specific conditions. Both causes and effects represent instances of common sense, the knowledge that every person assumes his/her neighbors also posses [4]. Common sense reasoning, a sort of reasoning we all perform about the everyday world [5], is the basic engine to deduce causal outcomes. Both knowledge and reasoning are often qualitatively characterized (i.e., expressed by natural language terms and sentences to acquire, state and reveal the meanings people give to their world and their experiences [6]).

This context is taken into account for the realization of our IDMA in the form of intelligent system for decision-making applicable in sciences ranging from social to applied, such as education. Thus, we present next a sample of related works, our conceptual model, its underlying items, a trial to test our approach, and the conclusions.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 534–543, 2012.

2 Related Works

With the purpose of shaping the context of our IDMA and distinguishing its contributions, a set of similar works is given in this section. So we identify some works related to decision making in the field of cognitive computing and cognitive reasoning, followed by causality and qualitative knowledge applications, besides the use of CM.

Wallach et al. adapt the Learning Intelligent Distribution Agent, a model of human cognition, to tailor both affective and rational features of moral decision making [7]. They show how moral decisions are made in many domains using the same mechanisms that enable general decision-making. Likewise, our IDMA is able to depict different kinds of features for decision-making. We also pursue to apply our IDMA in several domains by a general framework to depict knowledge and make reasoning.

Senglaub et al. outline a hybrid reasoning model for human decision-making, based on the C.S. Peirce's philosophical work, for computational representations or emulations of human cognition [8]. In the same way, we sketch a hybrid model because combines causality with fuzzy knowledge in order to reasoning about an issue.

Concerning the use of common sense for decision-making, Ramachandran applies Markov decision processes to scan KB, represented by semantic networks, for enabling reinforcement learning agents to learn optimal behaviors [9]. But in IDMA, we tailor CM and an ontology as the causal and semantic KB repositories respectively.

Osman and Shanks study how people weight base-rate information when they estimate causal strength and make judgments or inferences about the likelihood of an event [10]. However, our IDMA encourages people to express causal bias and judgments about how a concept exerts another in linguistic terms.

Premchaiswadi et al. define a framework for transforming decision-making statements into a set of qualitative sentences and probability inequality constraints for inference in a Bayesian Network [11]. In contrast, our IDMA applies a deterministic and inductive reasoning mechanism based on fuzzy logic and causality.

Al shayji and El Zant apply fuzzy logic to cope the uncertainty and reasoning problems for decision-making. They extract vague information in the form of fuzzy concepts and relationships to state them in an ontology [12]. Likewise our IDMA represents the meaning of fuzzy terms and causal relationships by means of an ontology.

Dubois and Prade propose a model of fuzzy relation equations and causal reasoning based on possibility theory and the twofold fuzzy sets for handling uncertainty in diagnosis problems [13]. In comparison, the IDMA adds an underlying principle of causality to estimate uncertainty as a variation of the consequent fuzzy linguistic term.

Tsadiras and Kouskouvelis design a fuzzy CM to analyze the integration of Turkey into the European Union [14]. Nevertheless, such a CM version is not a real fuzzy logic implementation, as the one used in the IDMA. Our IDMA applies a fuzzy rules-base CM version that blends fuzzy logic with the principle of causality.

Sobecki and Fijałkowski develop an approach to schedule courses, where students express their needs and preferences by several kinds of parameters, so the system recommends the sequence of courses to be taken [15]. Similarly, our IDMA supports the sequencing module of an IAWBES to evaluate the available options of content to teach a concept and choose the most promising option for being delivered to student.

3 Conceptual Description

In this section we set the architecture and performance of our IDMA. The IDMA is deployed as a multi-agent system (MAS) [16], which is able to interact with end-user and systems by the reception and delivering of messages based on a sort of agent communication language [17]. The IDMA plays *requester* and *responder* roles.

As requester, the IDMA demands information and services to *functional systems* to acquire, represent, and access features of the object to be modeled, such as: user, learning object (LO). As responder, *applicative systems* (i.e., those that provide specialized services to end-users, such as education) request evaluate alternatives and make decisions to the IDMA (e.g., respectively: estimate the learning impact that different LO produce on the user' learning, choose the best LO to be delivered).

Such roles are fulfilled by four kinds of agents: 1) *interface*: it performs requester and responder roles; 2) *management*: it holds support functions to feed, administrate and access the repositories of the KB such as the ontology; 3) *mapping*: it tailors a CM to shape the causal model to be analyzed; 4) *engine*: it triggers a simulation of fuzzy-causal effects to study the behavior of the model and estimate causal outcomes.

The performance of the IDMA is illustrated in Fig. 1 as a workflow, where messages and access to KB repositories (e.g., CM, ontology, and working memory of the fuzzy-causal simulation) are pictured respectively by wide and thin arrows.

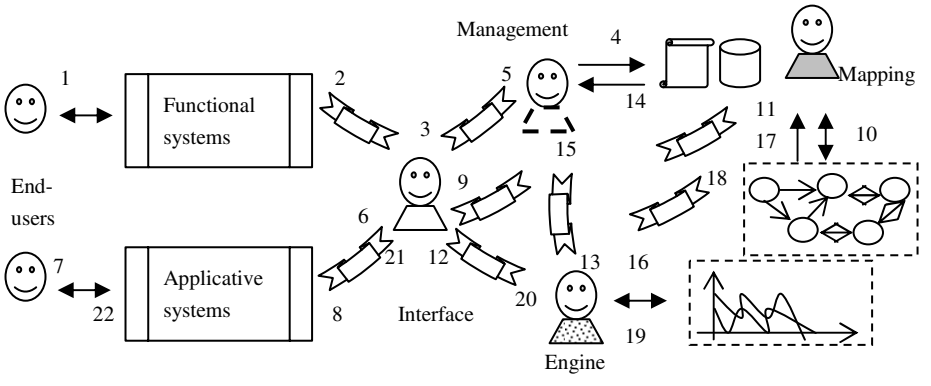


Fig. 1. The workflow of the IDMA is accomplished as follows: 1) End-user provides personal features to functional systems; 2) functional systems forward personal features data to interface agent; 3) it forwards data to management agent; 4) it stores data and update KB repositories; 5) it commits and informs to interface agent; 6) it informs "IDMA is ready" to applicative systems; 7) end-user interacts with applicative systems; 8) they request evaluation and decision-making to the interface agent; 9) it request the generation of a CM for each option to be evaluated to mapping agent; 10) it automatically tailors a CM for each option to be evaluated; 11) it commits and informs to interface agent; 12) it triggers engine agent; 13) it requests KB data to management agent; 14) it access requested data from KB; 15) it forwards KB data to engine agent; 16) it requests the CM of each option to be evaluated to mapping agent; 17) it access requested CM; 18) it forwards CM to engine agent; 19) it evaluates each CM by fuzzy-causal simulation and choose the best option; 20) it commits and informs the decision to interface agent; 21) it forwards the decision to applicative systems; 22) they interpret decision and achieve the corresponding action during their interaction with end-user.

4 Application Domain

Our IDMA is designed as a general mechanism to analyze and reason for intelligent decision-making applications. However, the description of its components and the explanation of its performance are illustrated by the application of the IDMA to the educational domain. So we shape a profile of the application domain in this section.

The student-centered paradigm pursues to enhance the apprenticeship of students by taking into account their cognitive strengths and weakness, needs, preferences, and goals. Thus, the paradigm advices to tailor educational curricula, lectures, support, assessment, and evaluation in such way they satisfy students' requirements.

IAEWS is an artificial intelligence approach to deploy the student-centered paradigm. It intelligently adapts interfaces, content, trials, assistance, and examination based on students' likings and constraints in order to increase students' learning [18].

The sequencing is a key module of the IAWEBS for a successful teaching-learning cycle, because of it decides what the student is going to do next. For instance, it chooses the LO that offers the most profitable students' learning achievement. This function claims the evaluation of the available LO options and the decision of which of them is the best LO option for being delivered to student. Such *applicative tasks* are candidates to be made by our IDMA, as it is explained in the rest of the paper.

5 Knowledge Acquisition

Before making a decision, it is necessary to model the subject of decision. Thus in this case, the applicative tasks are described by the features of the student and the content. They are collected by a knowledge acquisition framework. It is implemented by functional systems and the interface and management agents of the IDMA as follows.

In the case study, four domains of features are considered to depict the student: learning preferences, cognitive skills, personality traits, and knowledge. Whereas, two domains of features are used to set the educational content: concept and LO option.

With the aim at achieving certitude of the nature, attributes, and measures of the features, the functional systems account psychological models, pedagogical criteria, and web graphic design items to outcome a *student model*, a *content model*, and an *ontology*, which are reflected as part of a KB, by means of the next instruments [19]:

- The Gardner's Multiple Intelligence model (GMIM) is used to estimate eight instances of the learning preferences domain (Gardner, 1983).
- The Wechsler Adult Intelligent Scale (WAIS) is applied to measure eleven skills of the cognitive domain (Wechsler, 2002).
- The Minnesota Multiphasic Personality Inventory version 2 (MMPI-2) is accounted to analyze forty three traits of the personality domain.
- The Taxonomy of Learning Objectives (TLO) is enhanced to identify seven levels of domain knowledge mastered by students.
- A set of guidelines for using learning technologies with multimedia (GULTM) is accounted to depict properties of the concept to be taught and LO option domains.

6 Knowledge Base Organization

By means of the interaction between user (e.g., student and pedagogue) and the five functional systems, the features from student and content are acquired. The features are characterized by means of: *concepts*, *relations*, and *measure-values*. Such knowledge is added into the KB through the steps 1 to 5 of the workflow set in Fig. 1. Thus in this section, we describe the organization of the KB based on [20].

The KB is composed of several repositories to organize the application domain knowledge, such as: the student model, the content model, the ontology, the CM of each LO option to be evaluated, and results of the simulation decision-making process. Most of the repositories store raw data of the student, LO, CM and results by means of eXtended Markup Language (XML) documents. But the semantic definition of the items stored in those repositories is stated in the ontology repository. The semantic definition is made by sentences of the Web Ontology Language (OWL).

As regards with the student model, it contains three XML repositories: 1) *record* depicts personal data of the student; 2) *assessment* records the behavior, responses, and outcomes developed by the student during the sessions held with the IAWBES; 3) *knowledge* reveals the prior and the learned domain knowledge gained by the student during the teaching-learning experiences (lectures) provided by the IAWBES; 4) *profile* represents the main features of the learning preferences, cognitive, and personality domains. This repository is complemented by ten more (e.g., one for preferences, four for cognitive, and five for personality) that offer detailed knowledge of such domains.

Concerning the content model, it owns two repositories: 1) *meta-taxonomy* shapes the hierarchy of classes and properties to characterize LO; 2) *taxonomy* describes the features of a LO option, so there is one taxonomy repository for each LO option.

The CM repository shapes the topology of a CM by the *id* that identifies the items (e.g., cause and effect concepts, fuzzy rules-base) of the causal relationships. The results repository stores the linguistic values that instantiate the *level* and *variation* states of the concepts held in the CM along each point of time t_i of the simulation.

Essentially, the ontology is encoded by four kinds of OWL sentences: *class*, *DatatypeProperty*, *FunctionalProperty*, and class instances. The first is used to define a class and inheritance relationships with ancestors classes. The second and third are two options to define properties and attach them to a class. The fourth is applied for defining a specific object of a class, whose properties have instantiated values.

The ontology represents the meaning of fuzzy rules-bases (i.e., they have a cause-effect rule per each linguistic value that instantiates the antecedent), concepts (i.e., the features used to describe the student and content models), causal relationships (i.e., they identify the cause concept and the effect concept, besides the fuzzy rules-base that describes the relationship), series of linguistic terms (i.e., they mold the universe of discourse "UOD" of a fuzzy variable), linguistic terms (e.g., the fuzzy value attached to a linguistic variable), the type of relationship (e.g., fuzzy, causal), the type of concept (e.g., level, variation), terms (i.e., they are used to label concepts), associate terms (i.e., composed terms), version (i.e., special attributes used to depict LO).

The management of the KB is accomplished by repository management agents, such as the ontology agent that is fully detailed in [21].

7 Cognitive Mapping

The applicative tasks (e.g, evaluation of LO options and selection of the best one) is fulfilled by two processes: cognitive mapping and fuzzy-causal reasoning simulation. The earliest is explained in this section; whilst the latest is pointed out in the next one.

Conceptually speaking, the application domain is claimed as a teaching-learning cycle, where LO represent the *cause*, the student's mental faculties the *effect* (i.e., because of they are stimulated) and also the *cause* (i.e., due to they are used to learn), and the student's apprenticeship the *effect* (e.g. a specific concept to be learned).

Such a conceptual environment is shaped as a CM, whose topology embraces three tiers of concepts (e.g., like a three-layer artificial neural network). Where the first tier contains concepts that characterize the LO to be evaluated; the second tier holds concepts that describe the student; the third tier owns the concept to be taught (learned).

Based on the steps 7 to 11 of the workflow pictured in Fig. 1, the mapping agent receives the *id* of the LO options, the *id* of the student, and the *id* of the concept to be taught as parameters wrapped in a message. Afterwards, it automatically tailors the CM structure that corresponds for the LO to be evaluated. The cognitive mapping process draws the topology of the CM according to the concepts and relationships stated in the repositories of the student model, content model and ontology. A sample of the CM's topology tailored by the mapping agent is explained as follows:

The first level contains two types of concepts to represent the LO: *general* and *specific*. General concepts are features that describe the concept to be taught (e.g., in this case they are five adjectives: *abstract*, *abundant*, *complex*, *technical*). Specific concepts characterize a given option of LO authored for teaching such a concept (e.g., in this case they are eight adjectives: *dynamic*, *static*, *constructive*, *declarative*, *linguistic*, *non-linguistic*, *sonorous*, *visual*).

The second level models the student by a sample of concepts corresponding to three domains: 1) learning preferences: *auditory*, *logical*, *linguistic*, *visual*; 2) personality: *hysteria*, *psychasthenia*, *social introversion*, *depression*; 3) cognitive: *causal reasoning*, *intelligence quotient* (IQ), *auditory memory*, *visual memory*. Concerning the third level, it only includes the concept to be learned, as the goal to be achieved.

As regards with the topology characterized by causal relationships, concepts of the first layer only exert concepts of the second layer. Thus, there are 156 unidirectional causal relationships (13 * 12) between both layers. Concepts of the second level bias the concept of the third level, so there are 12 causal relationships (12 * 1) between both tiers. However, there are more causal relationships, such as feedback and auto-feedback between concepts of the second level and the one of the third. It means that: concepts of the second tier influence each other and themselves, so there are 144 causal relationships (12 * 11 + 12); and the concept of the third tier exerts concepts of the second level and itself, thereby there are 13 causal relationships (12 + 1).

In resume, the topology of the CM sample contains: 3 tiers, 26 concepts, and 325 causal relationships, where all the relationships are defined by their respective fuzzy rules-base. The CM holds unidirectional and bidirectional causal relationships between couples of concepts, besides of itself causal relationships for many concepts. Thus, the CM is a tool for modeling system dynamics, whose attributes are set in [22].

8 Decisions Making Engine

Decision-making represents a chance towards a given goal. It implies the existence of at least two options or action courses to accomplish an objective. So it is needed the evaluation of the available alternatives based on some criteria and profit. In this section, we show how the IDMA makes decisions through the evaluation of the LO options authored to teach a specific concept and the selection of the best LO. Therefore, once the LO options relying on their respective CM, the steps 12 to 22 of the workflow drawn in Fig. 1 are fulfilled to evaluate the LO and compare their results.

First of all, the state of the concepts is initialized with *level*, *variation*, or both values. Such values are measured during the application of the knowledge acquisition framework and are stored in the KB. A *level* is a qualitative value assigned to the state of a concept to show how intense is its presence in a point of time t_i. A *variation* reveals a sense and a qualitative degree of change after a short time t_j- t_i, ($j = i + 1$). Thus, the state of concepts that describe the LO are only instantiated by levels because of the content is static along the lecture; whereas the state of concepts that characterizes the student and the concept to be learned are qualified by both level and variation due to they are altered during the lecture. So a UOD with six levels (e.g., {*so low*, *low*, *medium*, *high*, *so high*}) are linked to the state of concepts that describe LO and the student; a UOD with seven levels (e.g., {*ignorance*, *knowledge*, *comprehension*, *application*, *analysis*, *synthesis*, *evaluation*}) is associated to the state of the concept to be taught; a UOD with eleven variations (e.g., decreases: *so much*, *much*, *regular*, *little*, and *so little*, *holds*, *increases*: *so little*, *little*, *regular*, *much*, *so much*)) is joined to the state of concepts that depict the student and the concept to be taught.

Thereafter, the simulation of causal effects is started and along discrete increments of time $t_1 \ldots t_n$ the causal effects on the state of the *effect* concepts (i.e., concepts biased by others) are estimated until a stable status is reached (i.e. when the state of each concept does not change, a loop of values appears) or a chaotic situation is faced.

Thus, during each point of time t_1, *causal inferences* (CI) and *fuzzy inferences* (FI) are computed according to the mathematical foundations detailed in [22], as follows: 1) the levels attached to the state of concepts in the first tier exert the state of concepts in the second tier as a variation estimated by CI; 2) the variations given to the state of concepts in second and third tiers bias the state of concepts in second and third tiers as a variation computed by CI; 3) the levels assigned to the state of concepts in second and third tiers exert the state of concepts in second and third tiers as a variation made by CI; 4) the variation and level provided to the state of concepts in second and third layers bias the level allocated to the state of concepts in second and third tiers by FI.

The state of concepts in the first level never changes; but, the level and the variation appointed to the state of concepts in second and third levels could change along the simulation. It means, the former level and variation (i.e., those used for initializing the state in t_0) given to the state of the concept to be learned (i.e. the one of the third level) could be altered (e.g., from *analysis* to *synthesis*, from *holds* to *increase much*).

Finally, the question: which is the best LO option? The decision is made by choosing the LO whose simulation achieved to its CM outcomes at t_n the highest level and the longest positive variation estimated as the final state of the concept to be learned.

9 Case Study

The support for decision-making given by our IDMA has been tested in the educational arena. It has been the responsible for deciding the most profitable LO to teach the student. Thus, according to the framework sketched in Fig. 1, it receives messages sent by the sequencing module of an IAWBES (e.g., the applicative system) before delivering a lecture to the student. The application of the workflow is illustrated next.

Our institution encourages graduates to make research as thesis and projects. But, most of them are living in several cities of the country. So we decided to use Internet to make a call for participation and provide a course.

Firstly, we launching a campaign to invite subjects interested in updating their domain knowledge related to the "scientific method". Several hundreds of people were aimed for participating of the free e-training. However, just 200 graduates enrolled by filing an e-form with personal data, which was stored in the record repository.

Secondly, applicants answered the preferences, personality, cognitive, and domain knowledge tests by interacting with the functional systems. The collected data were stored in student model repositories. Unfortunately, during the process 75% of the universe gradually deserted and only 50 people made up the population (N).

Thirdly, a course about "philosophy of the science" was delivered to the population in order to train them in the use and interaction with e-learning systems. Simultaneously, an introductory course was designed. It holds ten *basic concepts* (e.g., *hypothesis*, *law*, *theory*). Four options of LO were authored to teach each concept. The first stimulates objectivism learning, the second the constructivism learning, the third privileges sonorous linguist content, and the fourth prefers the visual messages.

Fourthly, only 18 participants fulfilled the training. Thus, they set the sample (n) and were randomly split into two comparative groups: *control* (C) and *experimental* (E). Moreover, a pre-measure concerning their prior knowledge of the basic concepts (PK) was made to the participants. The functional system measured the level of mastering held by the individual through the UOD with seven levels given by the TEO. But, the linguistic terms were converted to digits (e.g., 0 for *ignorance*…). So a participant got a PK sum between [0, 60], and the PK sum of a group was [0, 540].

Fifthly, the stimuli were provided to the participants through the delivering of just one LO option for each basic concept. However, the LO learned by C members were randomly chosen, and the LO taught to E members was selected by the IDMA.

Sixthly, based on the instruments and criteria used in the pre-measure, a post-measure was made to participants for estimating their final knowledge of the basic concepts (FK). Thereby, the difference between FK and PK reveals the learning gained (LG) by the participant and the whole group.

Seventhly, a statistical process was computed and the results were interpreted. As a sample of such outcomes several parameters concerning to the PK, FK, and LG are outlined in Table 1, besides three concepts of the student model.

Finally, although C group held more PK, 44% *high* IQ, and greater intercept (a) than the ones estimated for E group; at the end, E group overcame the disadvantage and achieved a higher learning FK and LG, and better Person's coefficient (r), linear regression slope (b) and overcoat the statistical significance (p).

Table 1. Main statistical measures outcome by comparative groups during the test

Criterion	Control group	Experimental group
Pre-measure	PK: Sum 42; mean 4.67	PK: Sum 38; mean 4.22
Post-measure	FK: Sum 174; mean 19.33	FK: Sum 198; mean 22
Learning gained	LG: Sum 132; mean 14.67	LG: Sum 160; mean 17.78
Logical preference	44.4% so high, 55% high	44% quite high, 44% high, 11 medium
Depression trait	22% high, 22% medium, 55% low	11% high, 22% medium, 66% low
IQ skill	44% high, 22% medium, 33% low	11% high, 22% medium, 66% low
Pearson's coefficient	r = 0.554	r =0.828
Statistical significance	p =0.122	P =0.0059
Linear regression	Post = a 13.7 + b 1.22 Pre-measure	Post = a 7.72 + b 3.28 Pre-measure

10 Conclusions

Decision-making is an activity that human beings and organizations daily make. Intelligent support systems are concerned with research, theory, modeling and practice of a better decision-making. Our IDMA is an instance of such kind of systems, whose underlying items are fuzzy and causal knowledge and reasoning. It includes a framework to guide the collection of features, the organization and management of knowledge, the modeling of the features as a CM, the evaluation of options and the selection of the best one, as it was shown in the case study: 17.5 % better learning was achieved for people who were benefited from the IDMA. Automatic acquisition of features and the application of the IDMA to different domains are being planned.

Acknowledgments. First author gives testimony of the strength given by his Father, Brother Jesus and Helper, as part of the research projects of World Outreach Light to the Nations Ministries (WOLNM). This work holds a partial support from grants: CONACYT-SNI-36453, CONACYT 118962-162727, CONACYT 118862, SIP-20120266, IPN-COFAA-SIBE, IPN-SIP-EDI: SIP/DI/DOPI/EDI-0505/11.

References

1. Modha, D.S., Ananthanarayanan, R., Esser, S.K., Ndirango, E., Sherbondy, A.J., Singh, R.: Cognitive Computing. Cognitive Computing Communications of the ACM 54(8), 62–71 (2011)
2. Anshakov, O.M., Gergely, T.: Cognitive Reasoning: A Formal Approach. Springer, Heidelberg (2010)
3. Pearl, J.: Causality: Models, Reasoning, and Inference. Cambridge University Press, Cambridge (2000)
4. Panton, K., Matuszek, C., Lenat, D., Schneider, D., Witbrock, M., Siegel, N., Shepard, B.: Common Sense Reasoning – From Cyc to Intelligent Assistant. In: Cai, Y., Abascal, J. (eds.) Ambient Intelligence in Everyday Life. LNCS (LNAI), vol. 3864, pp. 1–31. Springer, Heidelberg (2006)
5. Muller, E.T.: Common Sense Reasoning. Morgan Kaufman Publishers, California (2006)
6. Bartolomei, J.E.: Qualitative Knowledge Construction for Engineering Systems: Extending the Design Structure Matrix Methodology in Scope and Procedure. PhD thesis, Massachusetts Institute of Technology (2007)

7. Wallach, W., Franklin, S., Allen, C.: A Conceptual and Computational Model of Moral Decision Making in Human and Artificial Agents. In: Wallach, W., Franklin, S. (eds.) Topics in Cognitive Science, special issue on Cognitive Based Theories of Moral Decision Making, pp. 454–485. Cognitive Science Society (2010)
8. Senglaub, M., Harris, D., Raybourn, E.M.: Foundations for Reasoning in Cognition-Based Computational Representations of Human Decision Making. Technical report, Sandia National Laboratories (2001)
9. Ramachandran, D.: Using Common Sense for Decision Making in an Adventure Game. AAAI Spring Symposium Series. AAAI, California (2007)
10. Osman, M., Shanks, D.R.: Individual Differences in Causal Learning and Decision Making. Acta Psychologica 120, 93–112 (2005)
11. Premchaiswadi, W., Jongsawat, N., Romsaiyud, W.: Bayesian Network Inference with Qualitative Expert knowledge for Group Decision Making. In: 5th IEEE International Conference on Intelligent Systems (IS), pp. 126–131. IEEE Press, New York (2010)
12. Al Shayji, S., El Zant, N.: Building Fuzzy-Logic Ontology for Political Decision-Makers. Int. J. Mathematical Models and Methods in Applied Sciences 5(5), 991–1001 (2011)
13. Dubois, D., Prade, H.: Fuzzy Relation Equations and Causal Reasoning. Fuzzy Sets and Systems 75(2), 119–134 (1995)
14. Tsadiras, A.K., Kouskouvelis, I.: Using Fuzzy Cognitive Maps as a Decision Support System for Political Decisions: The Case of Turkey's Integration into the European Union. In: Bozanis, P., Houstis, E.N. (eds.) PCI 2005. LNCS, vol. 3746, pp. 371–381. Springer, Heidelberg (2005)
15. Sobecki, J., Fijałkowski, D.: Student Automatic Courses Scheduling. In: Nguyen, N.T., Trawiński, B., Jung, J.J. (eds.) New Challenges for Intelligent Information and Database Systems. SCI, vol. 351, pp. 219–226. Springer, Heidelberg (2011)
16. Wooldridge, M.: The Logical Modeling of Computational Multi-agent Systems. PhD thesis, Manchester Metropolitan University (1992)
17. FIPA-ACL: Agent Communication Language. Technical specifications, Foundation for Intelligent Physical Agents (2000)
18. Canales, A., Peña Ayala, A., Perdo, R., Sossa, H., Gutierrez, A.: Adaptive and Intelligent Web based Education System: Towards an Integral Architecture and Framework. Expert Systems with Applications 33(4), 1076–1089 (2007)
19. Peña Ayala, A.: Student Model based on Psychological Models. In: Uzunboylu, H. (ed.) WCES. Procedia-Social and Behavioral Sciences, vol. 1(1), pp. 1996–2000. Elsevier, UK (2009)
20. Peña Ayala, A., Sossa, H.: Semantic Representation and Management of Student Models: An Approach to Adapt Lecture Sequencing to Enhance Learning. In: Sidorov, G., Hernández Aguirre, A., Reyes García, C.A. (eds.) MICAI 2010, Part I. LNCS (LNAI), vol. 6437, pp. 175–186. Springer, Heidelberg (2010)
21. Peña Ayala, A.: Ontology Agents and their Applications in the Web-based Education Systems: Towards an Adaptive and Intelligent Service. In: Nguyen, N.T., Jain, L.C. (eds.) Intelligent Agents in the Evolution of Web and Applications. SCI, vol. 167, pp. 249–278. Springer, Heidelberg (2009)
22. Peña Ayala, A., Perdo, R., Sossa, H., Gutierrez, A.: Causal Knowledge and Reasoning by Cognitive Maps: Pursuing a Holistic Approach. Expert Systems with Applications 38(1), 2–18 (2008)

Camera Motion Estimation and Moving Object Detection Based on Local Feature Tracking

Andrey Vavilin, Le-My Ha, and Kang-Hyun Jo

Dept. Electrical Engineering and Information Systems, University of Ulsan,
Mugeo-dong, Nam-ku, Ulsan 689-740, Korea
{andy,acejo}@islab.ulsan.ac.kr

Abstract. This paper describes an approach for the camera motion estimation and moving object detection via tracking of the local image regions through the image sequence. The propose algorithm consists of two parts. First image of the sequence is used to compose a triangular grid which vertices are used as a feature points. Grid is optimized in order to increase number of elements in the regions with higher level of details. This grid is then used as an initial for the next frame in order to track local features. Neighborhood of each vertex is used to generate color distribution model which is used as a feature vector for tracking. In the second part of the proposed algorithm grids of two consistent frames used to estimate motion of correspondent grid vertices in order to form a motion field. This field is used to find dominant motions and make assumptions for background-foreground motion. Background motion is then used to estimate camera motion parameters.

Keywords: camera motion estimation, feature tracking.

1 Introduction

Camera motion estimation is an important issue for several video analysis tasks such as indexing and retrieval purposes, motion compensation, scientific film analysis and many others. From an aesthetical point of view, camera motion is often used as an expressive element in film production. Motion content can be used as a powerful cue for structuring video data, similarity-based video retrieval, and video abstraction.

Motion estimation and motion pattern classification problem has been extensively investigated by the scientific community for semantic characterization and discrimination of video streams. Moving object trajectories have been used for video retrieval [1–3]. Camera motion pattern characterization has been efficiently applied to video indexing and retrieval [4–7]. However, the main limitation of the latter methods is that they deal only with the characterization of the detected camera motion patterns, without explicit measurement of the camera motion parameters. As a result, the acquired information is of limited interest, since it can be used primarily for video indexing and retrieval.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 544–552, 2012.

There are different types of camera motion: rotation around one of the three axes and translation along the x and y-axis. Furthermore, zoom in and out can be considered as equivalent to translation along the z-axis. Existing methods can be classified as optical flow methods and feature correspondences based approaches. Let us also mention recursive techniques based on extended Kalman filters [8] which track camera motion and estimate the structure of the scene. In the case of an uncalibrated camera, interesting approaches are described in [9, 10]. The use of optical flow avoids the choice of "good features". In [11] differential approaches of the epipolar constraint are described. In [12], the optical flow computed between two adjacent images in a video sequence is linearly decomposed on a database of optical flow models. The authors of [13] propose a comparison of algorithms which only use optical flow for estimating camera.

In this work we present an approach for recovering grid-based structures from images. This structure is then used for estimating 3D camera motion. Our approach is based on decomposing image into the local regions and fixing geometrical relations between them. Initial grid is updating througth the image sequence.

2 Camera Model

In this work we considered eight-parameter perspective model defined as follows:

$$\begin{aligned} x_i^2 &= \frac{a_0 + a_2 x_i^1 + a_3 y_i^1}{a_6 x_i^1 + a_7 y_i^1 + 1} \\ y_i^2 &= \frac{a_1 + a_4 x_i^1 + a_5 y_i^1}{a_6 x_i^1 + a_7 y_i^1 + 1} \end{aligned} \tag{1}$$

Where $(x_1^1, y_1^1)$and (x_i^2, y_i^2) are the coordinates of the same point in to consequent frames at t_1 and t_2 respectively and $(a_0,\ldots,a_7)$ are the motion parameters. Various motion models can be derived from this mode. For example, in case of $a_6=a_7=0$ it is reduced to affine model, and setting $a_2=a_5$, $a_3=-a_4$ and $a_6=a_7=0$ will give us a translation-zoom-rotation model.

In [14] any vector field is approximated by a linear combination of a divergent field, a rotation field and two hyperbolic fields. The relationship between motion model parameters and symbol-level interpretation is established as:

$$\begin{aligned} Pan &= a_0 \\ Tilt &= a_1 \\ Zoom &= \frac{1}{2}(a_2 + a_5) \\ Rotation &= \frac{1}{2}(a_3 - a_4) \end{aligned} \tag{2}$$

Error in estimation parameters is defined as:

$$\varepsilon(a) = \sum_{i=1}^{N} \left\| p_i^2 - f\left(p_i^1, a\right) \right\| \tag{3}$$

Where N is number of corresponding points, $p_i^1 = \left(x_i^1, y_i^1\right)$ and $p_i^2 = \left(x_i^2, y_i^2\right)$ are the corresponding points in first and second frames, $a=(a_0,\ldots,a_7)$ is a transformation parameters vector and $f()$ is a transformation function defined by (1).

Using this model definition, problem of camera motion estimation could be formalized as the error minimization problem:

$$M = \arg\min_a \varepsilon(a) \tag{4}$$

Where M defines estimated camera motion parameters.

It is well known that, by taking some particular point p as the origin of the coordinate system with coordinates z, any infinitely differentiable function $f(x)$ could be approximated using Taylor series:

$$f(z) = f(p) + \sum_i \frac{\partial f}{\partial z_i} z_i + \frac{1}{2} + \sum_i \frac{\partial^2 f}{\partial z_i \partial z_j} z_i z_j + \ldots \approx c - b \cdot z + \frac{1}{2} z \cdot A \cdot z \tag{5}$$

where

$$c \equiv f(p) \qquad b \equiv -\nabla f \mid p \qquad [A]_{ij} = \frac{\partial^2 f}{\partial z_i \partial z_j} \mid p \tag{6}$$

The matrix A which consists of a second partial derivatives of the function is also called Hessian matrix of the function at p [15].

In approximation of (5) the gradient of f is easily calculated as

$$\nabla f(z) = Az - b \tag{7}$$

In Newton's method gradient is set to 0 to determine the next iteration point.

The gradient of error function ε with respect to parameters a has components

$$\frac{\partial \varepsilon}{\partial a_k} = -2 \sum_{i=1}^{N} \frac{\left[p_i^2 - f\left(p_i^1, a\right)\right] \partial f\left(p_i^1, a\right)}{\partial a_k}, \qquad k = 0,1,\ldots,7 \tag{8}$$

Taking second order partial derivatives gives

$$\frac{\partial^2 \varepsilon^2}{\partial a_k \partial a_j} = 2 \sum_{i=1}^{N} \frac{\partial f\left(p_i^1, a\right)}{\partial a_k} \frac{\partial f\left(p_i^1, a\right)}{\partial a_j} - 2 \sum_{i=1}^{N} \left[p_i^2 - f\left(p_i^1, a\right)\right] \frac{\partial^2 f\left(p_i^1, a\right)}{\partial a_k \partial a_j} \tag{9}$$

It is conventional to remove the factors of 2 by defining

$$\alpha_{kj} = \frac{1}{2} \frac{\partial^2 \varepsilon^2}{\partial a_k \partial a_j} \tag{10}$$

$$\beta_k = \frac{1}{2}\frac{\partial \varepsilon}{\partial a_k} \tag{11}$$

Making [a]=1/2A in equation (7), in terms of which that equation can be rewritten as the set of linear equations

$$\sum_{i=1}^{N} \alpha_{ki} \partial a_i = \beta_k \tag{12}$$

SVD is used to compute transformation parameters form overdetermined set of linear equations (12).

In the proposed work initial translation was estimated prior to pan-tilt-zoom estimation based on center of gravity of corresponding feature points in consistent frames. To remove outliers voting idea was used. After finding correspondences between frames each pair of matching points "votes" for its offset. Then points with small number of offset votes are discarded.

However, prior to camera motion estimation a set of correspondent points has to be determined.

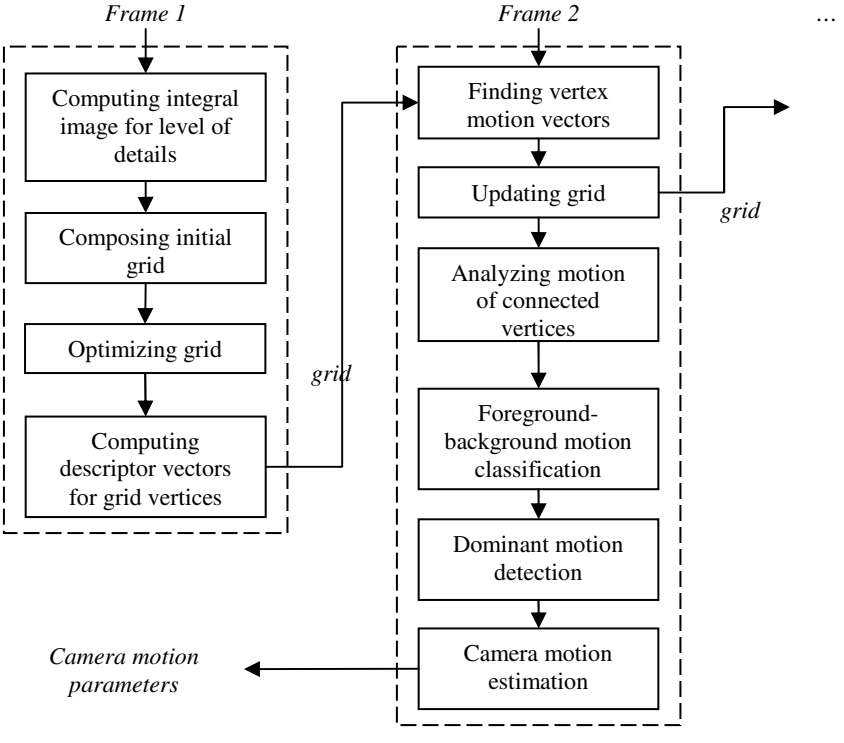


Fig. 1. Main algorithm scheme

3 Algorithm Description

In the proposed work image is described by set of feature points connected with a rigid grid. Vertices of the grid are represent features used for tracking and neighborhood of each vertex forms its description vector.

Main algorithm scheme is shown in Fig.1. First image in a sequence is used to compose the initial grid. Process starts with making a grid composed from equilateral triangles. Then the position of each vertex is updated to the local maximum of level of details [14,15] for some neighborhood. Radius of neighborhood is selected as 1/3 of minimum length among the edges, connected to this vertex. Level of details for neighborhood *B* is defined as follows:

$$D_B = \frac{1}{area} \sum_{(x,y) \in B} \max\left(\Delta I_x(x,y), \Delta I_y(x,y)\right) \tag{13}$$

where $\Delta I_x = \left|I(x+1,y) - I(x,y)\right|$ and $\Delta I_y = \left|I(x,y+1) - I(x,y)\right|$. Example of level of detail image and initial grid are shown in Fig.2(b) and Fig.2(d).

Position of grid vertices is then modified in order to move them to the local peaks of detail density function:

$$F(x,y) = \sum_{(i,j) \in N(x,y)} \max\left(\Delta I_x(i,j), \Delta I_y(i,j)\right) \tag{14}$$

where *N(x,y)* is a neighborhood of point with coordinates *(x,y)*. Example of this function is shown if Fig.2(c).

New coordinates *V'* of vertex *V* were selected from its neighborhood *R(V)* according to the following equation:

$$V' = \arg \max_{(x,y) \in R(V)} F(x,y) \tag{15}$$

Example of updated grid could be found in Fig.2(e). On the same time, if the amount of details presented in vertex neighborhood is higher than a predefined threshold all edges in connected triangles are separated into two parts forming 4 new triangles (Fig.2(f)). Finally, all triangles are optimized in order to match two criteria:

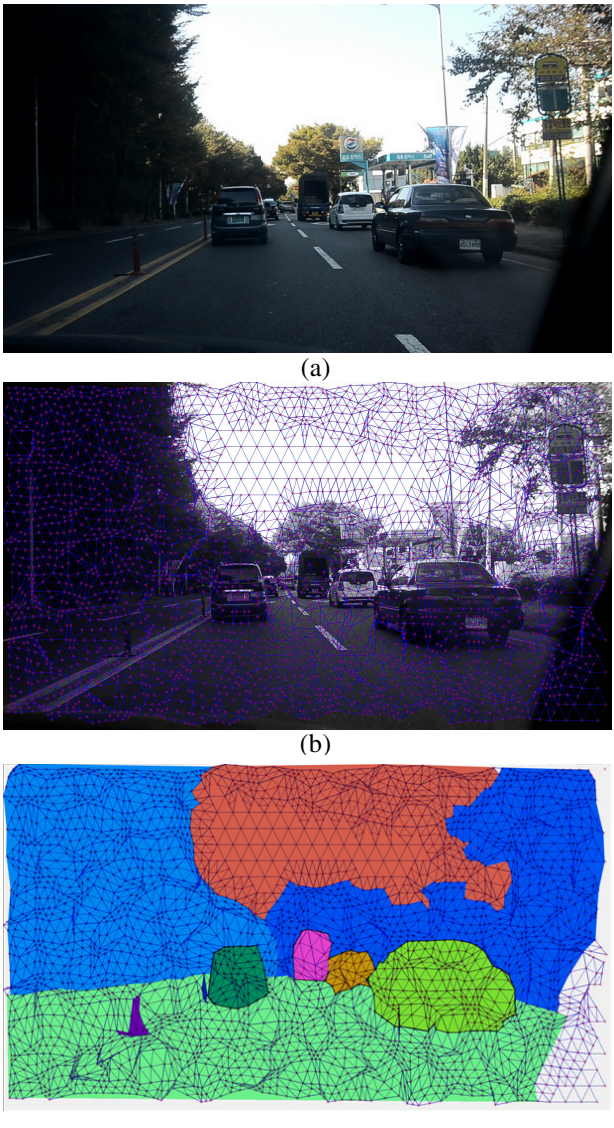
- all angles should be between 40 and 70 degrees
- all edges length rates in triangle should belong to (0.8,1.2)

Process repeats until detail level for each vertex will be less than threshold.

Fig. 2. Processing example: input image (a), gradient image (b), evaluation function for feature selection (c) initial grid, (d) grid with vertices moved to local maximum and (e) grids with disjointed elements for one iteration (f).

To minimize computational time for grid vertex position update and feature vector computation summed area tables [16] which could effectively utilized by using GPU computations [17] are used. Grid computed for the first image is then used as the initial for the second one. Offset between corresponding vertexes in consistent frames gives us motion field. Each motion vector votes for its direction. Connected vertexes with similar motion directions are more likely belong to

Vertex descriptor vector is formed from color probability distribution for its neighborhood. Vertexes are tracked between consequent frames using mean-shift algorithm. Motion vectors computed from correspondent vertices are analyzed in order to find dominant motion and detect local motion patterns. Regions with similar motion vectors are grouped and tracked together (see Fig.3 for detais).



(a)

(b)

(c)

Fig. 3. Motion analysis example: input image (a), tracked grid (b), regions of similar motion (c). White color used to show regions with no motion and red color shows region with not enough information to track

4 Experimental Results

All experiments were done on Intel Core 2 Duo with 2Gb memory. Program was not optimized for the maximum performance, thus, computational time could be decreased. To evaluate matching quality several types of experiments were done. First group of tests considered planar camera motion (translation and rotation around camera optical axis). Image sequences with camera smoothly by shifted by 20 cm in different directions and rotated by 20 degrees around its optical axis were made To evaluate quality of camera motion estimation average and maximum absolute errors were computed. Results for all groups of tests are shown in Table 1.

Second group of tests was used to evaluate algorithm performance for full 3d camera motion with known real camera trajectory. Three kinds of scenes were used: static scenes, scenes with moving objects and scenes with high amount of natural objects (trees, grass etc).

In the last group of tests image sequence with predefined camera motion trajectory was used to evaluate error depending on number of grid vertexes used for matching.

Table 1. Computational error (cm)

	Planar motion	**3D motion**		
		Static scene	Moving objects	Natural scene
Average absolute error	0,59	0,44	0,59	4,61
Maximum absolute error	0,6	3,7	3,93	9,02

Table 1 shows that the proposed method can effectively estimate camera motion for scene with moving objects. However, its weak point is natural scenes with small number of geometrical objects.

5 Conclusions

The proposed paper describes grid-based approach for camera motion estimation. Proposed algorithm could be used in various applications such as stereo matching, camera motion compensation, background model generation etc. Using non-arbitrary triangular grid allows us to add more elements in informative regions with high level of details while preserving small number of tracking points in regions with small level of details which are bad for tracking.

In future work we would like to investigate the effect of different kind of grids (ex. rectangular grid) and vertex tracking methods for motion estimation results. Additionally, context analysis could be used for effective motion estimation in scenes with complex motion (background parts moving with different velocities in addition to multiple moving objects).

Acknowledgements. This work was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2011. This work was supported by University of Ulsan, School of Excellence in Electrical Engineering.

References

1. Lie, W., Hsiao, W.: Content-based video retrieval based on object motion trajectory. In: Proc. IEEE Workshop on Multimedia Signal Processing, pp. 237–240 (December 2002)
2. Hu, W., Xie, D., Fu, Z., Zeng, W., Maybank, S.: Semantic-based surveillance video retrieval. IEEE Trans. Image Process. 16(4), 1168–1181 (2007)
3. Jianfeng, Y., Zhanhuai, L.: Modeling of moving objects and querying videos by trajectories. In: Proc. 10th Int. Multimedia Modelling Conf., Washington, DC, pp. 373–380 (2004)
4. Tan, Y.-P., Saur, D.D., Kulkarni, S.R., Ramadge, P.J.: Rapid estimation of camera motion from compressed video with application to video annotation. IEEE Trans. Circuits Syst. Video Technol. 10(1), 133–145 (2000)
5. Duan, L.-Y., Jin, J.S., Tian, Q., Xu, C.-S.: Nonparametric motion characterization for robust classification of camera motion patterns. IEEE Trans. Multimedia 8(2), 323–340 (2006)
6. Zhu, X., Elmagarmid, A.K., Xue, X., Wu, L., Catlin, A.C.: InsightVideo: Toward hierarchical video content organization for efficient browsing, summarization and retrieval. IEEE Trans. Multimedia 7(4), 648–666 (2005)
7. Lertrusdachakul, T., Aoki, T., Yasuda, H.: Camera motion characterization through image feature analysis. In: Proc. ICCIMA 2005, August 16-18, pp. 186–190 (2005)
8. Yao, A., Calway, A.: Robust estimation of 3-d camera motion for uncalibrated augmented reality. Technical Report CSTR-02-001, Dept of Computer Science, University of Bristol (2002)
9. Faugeras, O., Luong, Q.T., Papadopoulo, T.: The Geometry of Multiple Images. MIT Press (2000)
10. Pollefeys, M., Vergauwen, M., Cornelis, K., Tops, J., Verbiest, F., Van Gool, L.: Structure and motion from image sequences. In: Proceedings of Conference on Optical 3-D Measurement Techniques V, pp. 251–258 (2001)
11. Ma, Y., Kosecká, J., Sastry, S.: Linear differential algorithm for motion recovery: A geometric approach. IJCV 36(1), 71–89 (2000)
12. Park, S.C., Lee, H.S., Lee, S.W.: Qualitative estimation of camera motion parameters from the linear composition of optical flow. PR 37(4), 767–779 (2004)
13. Tian, Y., Tomasi, C., Heeger, D.J.: Comparison of approaches to egomotion computation. In: IEEE Computer Society, Conference on Computer Vision and Pattern Recognition, pp. 315–320 (1996)
14. Rovid, A., Varkonyi-Koczy, A.R., Hashimoto, T., Balogh, S., Shimodaira, Y.: Gradient Based Synthesized Multiple Exposure Time HDR Image. In: IMTC 2007, Warsaw, Poland, May 1-3 (2007)
15. Vavilin, A., Jo, K.-H.: Recursive HDR Image Generation from Differently Exposed Images. In: Proceedings of Graphicon 2008, Moscow, Russia, June 23-27, pp. 156–160 (2008)
16. Crow, F.C.: Summed-area tables for texture mapping. In: Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques, pp. 207–212. ACM Press (1984)
17. Hensley, J., Scheuermann, T., Coombe, G., Singh, M., Lastra, A.: Fast Summed-Area Table Generation and its Applications. Computer Graphics Forum 24(3), 547–555 (2005)

Optimization of Criminal HotSpots Based on Underlying Crime Controlling Factors Using Geospatial Discriminative Pattern

Dawei Wang[1], Wei Ding[1], Tomasz Stepinski[2], Josue Salazar[3], Henry Lo[1], and Melissa Morabito[4]

[1] Department of Computer Science, University of Massachusetts Boston
[2] Department of Geography, University of Cincinnati
[3] Department of Computer Science, Rice University
[4] College of Liberal Arts, University of Massachusetts Boston

Abstract. Criminal activities are unevenly distributed over space. The concept of hotspots is widely used to analyze the spatial characters of crimes. But existing methods usually identify hotspots based on an arbitrary user-defined threshold with respect to the number of a target crime without considering underlying controlling factors. In this study we introduce a new data mining model – *Hotspots Optimization Tool* (HOT) – to identify and optimize crime hotspots. The key component of HOT, Geospatial Discriminative Patterns (GDPatterns), which capture the difference between two classes in spatial dataset, is used in crime hotspot analysis. Using a real world dataset of a northeastern city in the United States, we demonstrate that the HOT model is a useful tool in optimizing crime hotspots,and it is also capable of visualizing criminal controlling factors which will help domain scientists further understanding the underlying reasons of criminal activities.

Keywords: Crime Hotspot, Hotspots Optimization Tool, Geospatial Discriminative Pattern, Footprint.

1 Introduction

The use of crime hotspots—spatial locations of high crime concentration [3]—is a key component in the study of criminal related problems. The existence of hotspots is due to the nature that criminal activities are unevenly distribution over space. The reasons driving the distribution of crime incidents have been explained in relation to the interaction of target and offender and the strength of guardianship [5]. An accurately identified crime hotspot map will significantly benefit police practise such as threat visualization, police resources allocation, and crime prediction, etc. [4].

However, commonly used hotspots identification methods such as point mapping, thematic mapping, and kernel density estimation (KDE) rely on a user-defined threshold and none of them have taken the underlying controlling factors of crimes into account. There is a potential error when using user-specified

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 553–562, 2012.

thresholds because the contrast between hotspots and normal areas may be ill-defined. For example, if a block with more than ten crime incidents a year is identified as a hotspot, then is there a large difference between this hotspot and the blocks that have nine crime incidents a year? A better way to accurately locate hotspots is to identify them not only by the criminal density, but also considering the underlying controlling factors.

In this paper, we introduce a new data mining model, *Hotspots Optimization Tool* (HOT)(Fig. 1), to improve the identification of hotspot by optimizing its boundary through the spatial footprints of patterns of crime driving factors. In the proposed method, a pattern means a combination of values of relevant variables. And patterns capable of identifying hotspots out of non-hot (normal) areas from the spatial perspective are called Geospatial Discriminative Patterns (GDPatterns) [7]. The HOT method adaptively optimizes the crime hotspots while searching for GDPatterns between crime hotspots and normal areas. Using a real world six-year dataset of a northeastern city in the United States, we demonstrate that the HOT model is a useful tool in optimizing crime hotspots, and it is also capable of visualizing criminal controlling factors which will help domain scientists further understanding the underlying reasons of criminal activities.

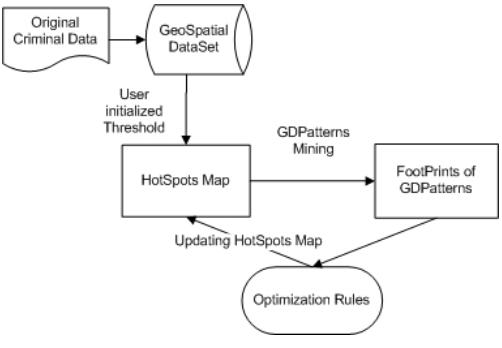


Fig. 1. The framework of *Hotspots Optimization Tool* (HOT). The boundaries of hotspots are updated using GDPatterns according to the optimization rules.

The rest of the paper is organized as follows. In Section 2 related works are discussed. Section 3 introduces the data representation and formal definition of the research problems. The *Hotspots Optimization Tool* is also presented in section 3. Our experimental results are discussed in Section 4. And in Section 5 we conclude the paper and discuss future research directions.

2 Related Work

Classic criminal theories, such as the Routine Activities Theory [5], conclude that three concepts contribute to crime: accessible and attractive targets, a

pool of motivated offenders, and lack of guardianship. The concepts of "tipping point"[10] and "disorder"[17] explain why adjacent areas of crime hotspots are at higher risk. A recent work done by [16] also discusses how an area is affected by the activity scope of offenders.

The Spatial and Temporal Analysis of Crime (STAC) program [2] is one of the earliest and widely used hotspot mapping applications. STAC uses "standard deviational ellipses" to display crime hotspots on a map and does not pre-define spatial boundaries. But some studies [9] show that STAC may be misleading because hotspots do not naturally follow the shape of ellipses. Another popular hotspot representation method is thematic mapping, in which boundary areas (geographic boundaries like census blocks or uniform grids) are used as the basic mapping elements [12]. Compared to point mapping, thematic mapping uses aggregate data, and spatial details within the thematic areas are lost. Also, the identified hotspots are restricted to the shape of thematic units. Kernel density estimation (KDE) [18] aggregates point data inside a user-specified search radius and generates a continuous surface representing the density of points. It overcomes the limitation of geometric shapes but still lacks statistical robustness that can be validated in the produced map [4]. All the above methods focus only on the target crime data and none of them consider underlying controlling factors of crime incidents.

Geospatial Discriminative Pattern applies emerging patterns to the spatial content. Emerging patterns are firstly introduced in [8] and further systematically studied in [14]. In the work of [7] they adopted the relative risk ratio as the measure of pattern emergence and use the method in vegetation remote sensing datasets. In our work GDPatterns are used as a tool to spatially mine the statically significant difference between target crime hotspots and normal areas with respect to its underlying related factors. It is the first time that GDPatterns have been used in the field of crime hotspot study.

3 Methodology

In this section, we will formally define the research problem and then present the HOT algorithm. To find GDPatterns of a target crime and its associated variables, a transaction-based geospatial database needs to be built. A widely used method for representing spatial distribution of entities is grid thematic mapping [11]. In this work we firstly generate a grid mask to cover the studied area. Variable data (both target crime and explanatory variables that contain information about underlying controlling factors of target crime) in the original spatial dataset is plotted onto a grid map with the same dimension as the mask. The cell in the grid is assigned as the count of incidents falling into it.

Since the explanatory variables come from very different sources, the range of their values varies. As with most criminal activities, the counts of cells with same values in each grid map follow a power-law distribution [6]. A better way to fairly represent all the variables in one pattern is to categorize them and change the original values into categorized numbers. Jenks Optimization for Natural

Breaks Classification [13], a method that is based on natural groupings inherited in data is used to divide every variable into categories. Using the Nature Break method the categories' breaks are identified that best group similar values, and the differences between categories are maximized.

Finally, with a user-specified threshold, the cells of the target crime grid can be classified into two classes: hotspots and normal area and a transaction-based geospatial dataset D is built.

Definition 1. *Geospatial database object*: A geospatial database object is a tuple of the form: $\{x, y, V_1, V_2, ..., V_n, C\}$, where x, y indicate the object's spatial coordinates, $V_1, V_2, ..., V_n$ are the categorized values of the explanatory variables, and C is the class label of target crime. C is 0 if the area is not a hotspot (or normal area) and 1 if the area is a hotspot. Using C, objects in D are labelled into the class of D_h (hotspots) if $C = 1$, or D_n (normal area) if $C = 0$.

3.1 Geospatial Discriminative Patterns

Here we give a brief introduction of *Closed Frequent Patterns* [15], GDPatterns and related definitions.

Definition 2. *Transaction and pattern*: In a geospatial database, a transaction T is the group of explanatory variables $(V_1, V_2, ..., V_n)$ in an object. An pattern X is a set of values of explanatory variables (e.g. $V_1 = 1$, $V_3 = 4$). For example, disregarding the class label C, in dataset D each object can be viewed as a transaction in location (x, y) with a fixed-number of variables.

Definition 3. *Support*: A pattern is said to be supported by a transaction when it is a subset of the transaction. For example, given a transaction T { V_1=1, V_2=1, V_3=2, V_4=2, V_5=3, V_6=5 }, patterns X_1 {V_1=1, V_2=1, V_5=3} and X_2 {V_1=1, V_3=2, V_4=2 } are supported by T, though X_3 {$V_1 = 1$, V_5=5, V_6=3} is not because it is not a subset of T. The number of transactions that support an pattern X is called the support count (suppcount) of X. The support of X is the ratio of $X's$ suppcount and the total number of transactions in a geospatial database (Formula 1).

$$sup(X) = \frac{suppcount(X)}{\tau} \tag{1}$$

where $sup(X)$ is the support of pattern X and τ is the number of transactions.

Definition 4. *Closed frequent patterns*: An pattern X is said to be a closed pattern when none of its immediate super-sets has exactly the same support as X. A closed pattern can represent a set of non-closed patterns without losing any support information, because the support of non-closed patterns can be calculated directly from the closed pattern. Using closed patterns will effectively reduce the total number of patterns. Furthermore, X is a closed frequent pattern if the support of X is greater than a user-defined minimum support threshold (ρ). We are only interested in closed frequent patterns because infrequent patterns are likely to be insignificant and may happen by chance.

The patterns we are looking for should meet two requirements: (1) to significantly represent the situation or conditions of explanatory variables in objects in D; (2) to significantly distinguish classes (D_h, D_n) from dataset D. A closed frequent pattern can satisfy the first requirement. To capture the difference of classes, the patterns should be more frequent in one class than in another.

Definition 5. *Geospatial Discriminating Patterns (GDPattern)*: In a geospatial database, a closed frequent pattern X is also a GDPattern if the growth ratio(δ) of X is larger than a user defined threshold. Here, growth ratio of a pattern is defined as the ratio of its supports in different classes.

$$\delta = \frac{sup(X, D_h)}{sup(X, D_n)} \tag{2}$$

where δ is the growth ratio; $sup(X, D_h)$ is the supports of closed frequent pattern X in class D_h and $sup(X, D_n)$ is supports of closed frequent pattern X in class D_n.

Definition 5. *Footprint of a GDPattern*: The footprint of a GDPattern X is the objects that support X in geospatial dataset D (Fig. 2). It is the set of cells whose correspondent objects support X in the grid map of study area. Footprints of GDPatterns provide a way to measure the spatial distribution of those patterns in studied area.

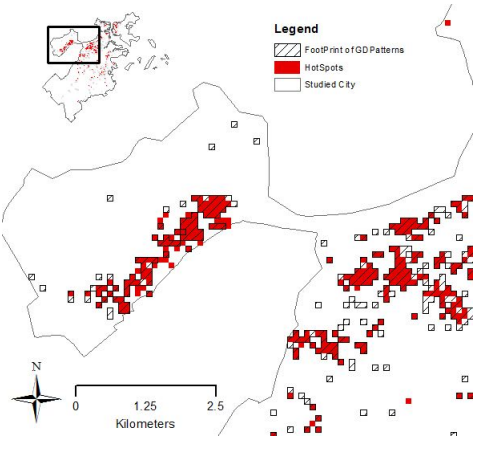


Fig. 2. A example map of GDPatterns Footprints. By selecting Residential Burglary(RB) data as the target crime, nine other variables are used as explanatory variables from the experiment dataset and 1,500 GDPatterns are mined with a growth ratio larger than twenty. The red area are RB hotspots with a user defined threshold and hallow squares with slash lines are footprints of the 1,500 GDPatterns.

Hence, with a rational threshold of growth ratio the GDPatterns mined from D are significantly different between classes and are capable of digging out the meaningful information underlying the spatial distribution of target crime hotspots.

Algorithm 1. *The Hotspot Optimization Tool* takes as input a geospatial dataset D, a hotspot threshold h, a hotspot candidate threshold h', a closed frequent pattern threshold ρ, a growth ratio threshold δ, and returns a new set of hotspots D_h, a set of GDPatterns G, and their footprints ψ.

Data: D, h, h', ρ, δ
Result: D_h, G, ψ
$count = 1$;
Generate D_h, $D_{h'}$ and D_n;
while $count \neq 0$ **do**
  $count = 0$;
  $\mu = \emptyset$;
  $G =$ Mine GDPatterns using D_h, ρ and δ;
  $\psi = footprints(G)$;
  for *cell* $c \in D_{h'}$ **do**
    if *c adjacent to some cell in* D_h *and* $c \in D'_h$ **then**
      $\mu = \mu \cup c$;
    end
  end
  for *cell* $c \in \mu$ **do**
    if $c \in \psi$ **then**
      $D_h = D_h \cup c$;
      $count{+}{+}$;
    end
  end
end

3.2 Hotspot Optimization Tool

As mentioned above, locating hotspots with a user defined threshold is not sufficient. Here we introduce a model, *Hotspot Optimization Tool* (HOT), to emphasize the identification of hotspots by optimizing user-specified hotspot boundaries. The practicality of HOT is based on two concepts: firstly, a hotspot can be considered as a "tipping point"[10] or the source of "disorder"[17] of its adjacent blocks, which means the adjacent areas have the possibility of being affected by crimes happening in hotspots. Also, from the point of view of spatial correlations [1], adjacent areas (cells) of a hotspot cell are more likely to fall into the active range of the same criminals. Therefore these areas (adjacent cells) are potential hotspots, especially those with a relatively high crime density. Secondly, according to the definition, GDPatterns are much more frequent in hotspots than in normal area. Normal areas located in the footprints of GDPatterns are more likely to be hotspots because in these areas the values of explanatory variables are the same.

With a target crime being selected, to find hotspots (D_h) we firstly initialize a threshold of target crime rates. Then we optimize the boundaries of hotspot using

HOT (Algorithm 1) with the intrinsic discriminative information embedded in the GDPatterns:

This algorithm does the following:

- Identify areas with a relatively high crime density ($D_{h'}$, areas with high target crime density that are close to the density in hotspots, line 2);
- Mine GDPatterns based on current hotspot boundaries and draw the footprints of GDPatterns (lines 6 and 7);
- Generate candidate cells(lines 8-12): cells located in $D_{h'}$ and adjacent to some cell in D_h.
- Test the hypothesis for candidate cells (line 14): a candidate cell is inside the footprints of GDPatterns (ψ);
- If the hypothesis is true, the boundaries of the hotspot are modified by changing the current cell into a hotspot cell (from $D_{h'}$ to D_h) (line 15);
- Iterate until all hypothesis tests are fault (line 3 and line 19).

When the boundaries of a hotspot are changed, a new set of GDPatterns will be generated based on the modified hotspots, followed by the change of footprints. If in the current loop the set of GDPatterns is the same as the former loop, it means there are no new footprints and there will be no "true" from the hypothesis test (lines 4-10 in Algorithm 1). The HOT will stop and a new optimized hotspot map is generated.

4 Experiment Results

4.1 Data Preprocessing

The experiments are done using historical data with a time span of six years (2004-2009) from a northeastern city in the United States. The size of study area is 130.1 km^2 and the approximate population is 600,000. As one of the most frequently reported and resource-demanding crimes in the studied city (according to the city police department report), Residential Burglary (RB, burglaries target at residential houses) is selected as the target crime. In addition to RB, total of eight social/criminal features are selected in this study as explanatory variables with the help of a domain expert. Among those are:

- Commercial Burglary (CB, burglaries that target at commercial sites), Street Robbery (SR), Motor Vehicle Larceny (MV, crimes against possession inside vehicles) and Arrest data (AR) are related criminal data that pictured the level of activity of crimes. The rates of CB, MV, and ST reflect the strength of guardianship in the area. Arrest rate is a good indicator for the pool of offenders.
- Foreclosed Houses (FC, houses that are redeemed by mortgage lender) reflect the house vacancy conditions and a vacant house has a higher risk of being broken into than an inhabited one. It is also an indicator of guardianship.

- The spatial density of RB is affected by the density of population (POP) and number of houses units (HU). A hotspot map of RB may simply be displaying locations of high housing density because such areas have a potential higher RB rate than areas with fewer houses.
- The studied city is a hub of higher education and a significant amount of houses near universities or colleges are usually rented by students or scholars, which make them easy targets of burglars during semester breaks. The variable of Distance to Colleges (DC) is used to address this concern.

The original criminal dataset comes as vector maps (points and polygon). A grid map is made as a mask to cover the whole study area and acts as the background map for data preprocessing. The cell size selected is $100m \times 100m$, which results in a number of 12,984 cells in the study area. There are two concepts to consider when choosing an appropriate cell size. Firstly, the cell is approximately half the size of average city block size ($19,873m^2$) in the studied city, which will be a good representative of reality. Secondly, with this cell size the number of cells which fall into the study area is at the same order of magnitude with the number of RB incidents, which minimizes the loss of spatial information during aggregation.

4.2 Hotspots Optimization

An initial threshold of RB hotspots is needed to set the initial classes before the HOT algorithm is used. From the study of [16], a house is under a relatively higher risk if a burglary happened in the nearby area in the past four months. Relatively, if three or more burglary incidents happened in the block in one year, the area is likely a hotspot of burglary. Because the time span of our RB data is six years, we set an area (cell) to be a hotspot if there are eighteen or more burglary incidents ($h \geq 18$).

Using a support threshold of 0.001, 6,327 patterns are mined out of which top 1,500 are selected with a growth ratio more than twenty ($\delta > 20$), which indicate with an at least 95% confidence level (1:20) that these GDPatterns will reveal the difference between hot spots and normal area. We use the threshold of 9 RB incidents($18 > h' \geq 9$), half of the initial value used for hotspots, to define the "potential hot" area ($D_{h'}$). In the 6th loop OHS reaches the final condition and stops (Fig. 3). A final version of the set of patterns is extracted and the growth ratios of top 1,500 GDPatterns are all greater than 50, which is at least twice the initial version.

The new hotspot grid map is projected with satellite images of the studied city and a figure of sample site is extracted and shown in Fig. 4. Using an arbitrary threshold (h) the red cells are classified into hotspots and cells in same blocks (in the colour of blue) have been left out. It is reasonable that houses located in the same block have a similar risk of being broken into. Our optimization method successfully captures these cells and modifies the hotspot boundaries rationally. Also, cells which are mostly covered by natural land, parking lots, roads and highways identified and are not classified into hotspots using our methods.

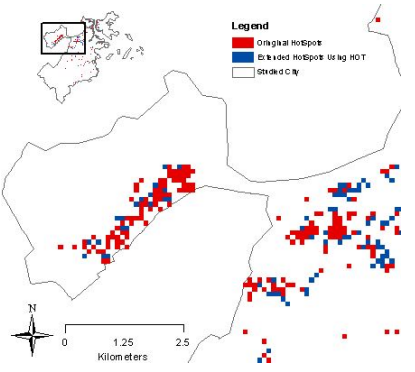


Fig. 3. Optimized hotspots map of the studied city. The purple cells are hotspots initially defined by the user-defined threshold and the blue cells represent hotspots that are added from candidate areas using HOT.

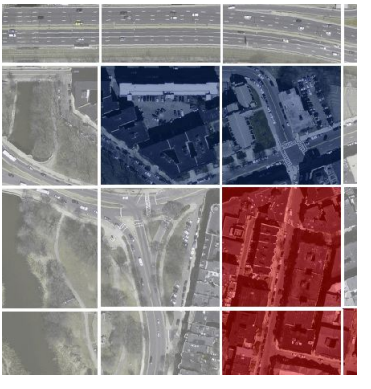


Fig. 4. A re-projection example of hotspots with satellite images. The purple cells are hotspots defined by the original threshold and the red cells are hotspots identified using our optimization method.

5 Conclusion and Future Work

In this paper we present a data mining model –Hotspots Optimization Tool – to optimize crime hotspots using GDPatterns. It is a first time attempt of using GDPatterns in crime hotspots analysis. Using a real world dataset we have proved that our model is capable of identifying crime hotspots by considering the controlling factors of criminal activities. This is important in criminal analysis because we can visualize areas that are in danger of becoming unstable and changing into a pool of criminal activity.

The GDPatterns mined in the process is an information-rich dataset and from which more details of crime driving factors can be extracted. The optimization

process is not only a visualizing of crime itself but also an visualization of controlling factors and will help our understanding of the underlying reasons of criminal activities. In our future work, we will focus on rational structured and re-organized GDPatterns.

Acknowledgement. The work was partially funded by the National Institute of Justice (No.2009-DE-BX-K219).

References

1. Bailey, T.C., Gatrell, A.C.: Interactive spatial data analysis. Longman Scientific & Technical Essex (1995)
2. Bates, S.: Spatial and temporal analysis of crime. Research Bulletin (April 1987)
3. Chainey, S., Ratcliffe, J.: GIS and crime mapping, vol. 6. John Wiley & Sons Inc. (2005)
4. Chainey, S., Tompson, L., Uhlig, S.: The utility of hotspot mapping for predicting spatial patterns of crime. Security Journal 21(1), 4–28 (2008)
5. Cohen, L.E., Felson, M.: Social change and crime rate trends: A routine activity approach. American Sociological Review, 588–608 (1979)
6. Cook, W., Ormerod, P., Cooper, E.: Scaling behaviour in the number of criminal acts committed by individuals. Journal of Statistical Mechanics: Theory and Experiment 2004, 07003 (2004)
7. Ding, W., Stepinski, T.F., Salazar, J.: Discovery of geospatial discriminating patterns from remote sensing datasets. In: Proceedings of SIAM (2009)
8. Dong, G., Li, J.: Efficient mining of emerging patterns: Discovering trends and differences. In: Proceedings of the 5th ACM SIGKDD, pp. 43–52. ACM (1999)
9. Eck, J.E., Chainey, S., Cameron, J.G., Leitner, M., Wilson, R.E.: Mapping crime: Understanding hot spots (2005)
10. Gladwell, M.: The tipping point: How little things can make a big difference. Little, Brown and Company (2000)
11. Harries, K.D.: Mapping crime: Principle and practice. US Dept. of Justice, Office of Justice Programs, Crime Mapping Research Center (1999)
12. Hirschfield, A.: Mapping and Analysing Crime Data: Lessons from research and practice. CRC (2001)
13. Jenks, G.F.: The data model concept in statistical mapping. International Yearbook of Cartography 7, 186–190 (1967)
14. Li, J., Liu, G., Wong, L.: Mining statistically important equivalence classes and delta-discriminative emerging patterns. In: Proceedings of the 13th ACM SIGKDD, pp. 430–439. ACM (2007)
15. Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Discovering Frequent Closed Itemsets for Association Rules. In: Beeri, C., Bruneman, P. (eds.) ICDT 1999. LNCS, vol. 1540, pp. 398–416. Springer, Heidelberg (1998)
16. Short, M.B., Bertozzi, A.L., Brantingham, P.J.: Nonlinear patterns in urban crime: Hotspots, bifurcations, and suppression. Journal on Applied Dynamical Systems 9, 462 (2010)
17. Skogan, W.G.: Disorder and decline: Crime and the spiral of decay in American neighborhoods. Univ. of California Pr. (1992)
18. Wand, M.P., Jones, M.C.: Kernel smoothing, vol. 60. Chapman & Hall/CRC (1995)

Spatio-temporal Reasoning with Qualitative and Quantitative Information about Constraints and Approximations

Anthony Y. Chang

Department of Information Technology, Overseas Chinese University,
Taichung, Taiwan, R.O.C.
achang@ocu.edu.tw

Abstract. This paper constructs a computational model for spatio-temporal constraint reasoning and approximate measuring by quantitative and qualitative methods. We develop a spatio-temporal algebra system as an efficient computational model to unify time-varying events and spatial-varying objects. Several computation algorithms and operations for dealing with spatio-temporal constraints are proposed. The combined temporal/spatial models are generalized by composing point and interval algebra with qualitative and quantitative functions. The time interval relations can be encoded and represented as a neat graphical representation. The interval relational-distances were derived for comparing similarity between two intervals relations. Base on the temporal models and the stage graph with relational-distances, the algorithms are extended to compute spatial relations between objects in space.[1]

Keywords: Temporal Data Model, Intelligent Systems, Temporal/Spatial Data Mining, Spatio-Temporal Constraints, Approximate Measuring.

1 Introduction

Representing spatio-temporal knowledge is an essential part of many computer applications. Researchers of artificial intelligence, linguistics, knowledge systems, and information science require a time model for knowledge representation and analysis. Allen [1] has proposed an interval framework and Vilain and Kautz [3] have proposed a point framework for representing indefinite qualitative temporal information. These frameworks are influential and have been applied in diverse areas such as multimedia database [4], composite services [5], multimodal analysis of human behavior [6], semantic web [7], and data mining [2]. In this paper, two fundamental temporal frameworks are integrated and extended for spatial representations.

Constraint satisfaction techniques play an important role in current computer science. Many difficult problems involving search from areas such as machine vision,

[1] This work is supported by the National Science Council of Taiwan, under Grant NSC-100-2221-E-240-005.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 563–573, 2012.

scheduling, graph algorithms, machine design, and manufacturing can be considered to be the cases of the constraint satisfaction problem. In our previous work [8], an Interval Transitive closure Table and constraint propagation algorithms are proposed for temporal reasoning. This point-interval algebra is also extended for spatial constraint reasoning.

2 Representation of Spatio-temporal Relations

To deal with qualitative temporal representations, we divide the real number in timeline into three intervals: [-∞, 0], [0, 0] and [0, +∞]. These three intervals were denoted by {<}, {=} and {>} for representing the relationship between two time instants. Based on the encoding of endpoint relations between intervals, the well-defined interval relations are summarized as qualitative variables. The time intervals are formally defined as follows:

***Definition 2.1*:** Well-defined Interval
If $\boldsymbol{P}$ is the set of points and ≤ is the usual ordering on $\boldsymbol{P}$, then [$\boldsymbol{P}$; ≤] is partially ordered. Let a, b ∈ $\boldsymbol{P}$ such that a ≤ b. The set {x | a ≤ x ≤ b} is called a *well-defined interval* of $\boldsymbol{P}$ and denoted as [a, b]. The set {x | a < x < b} is called a *pure interval*, which is the subset of well-defined interval. ■

***Definition 2.2*:** Endpoint Relations
If $\boldsymbol{P}$ is the set of points, a binary relation ✧ of two points based on point relations {<, =, >}. Let A:[a, b] and B:[c, d] are two intervals, where a, b, c, d ∈ $\boldsymbol{P}$. The binary relations a⊗b and c⊗d are the *duration* relations. And the binary relations a⊗c, a⊗d, b⊗c, and b⊗d are called *endpoint relations*. These relations are denoted as $A_s \otimes B_s$, $A_s \otimes B_e$, $A_e \otimes B_s$, and $A_e \otimes B_e$ respectively. ■

Based on qualitative point relations, we use an encoding method to generalize and prove the 13 interval exclusion relations. Suppose A_s and A_e are the starting and ending points of the line segment A. And, B_s and B_e are those of B. We define a binary relation, ⊗, (either <, =, or > for "A is before B", "A is the same as B", or "A is after B") of two points. The 13 *interval relations* introduced by Allen [1] make the binary relations hold in the first part of the following table:

Table 1. Relations between Endpoints of Two Intervals

$A_s \otimes B_s$	$A_s \otimes B_e$	$A_e \otimes B_s$	$A_e \otimes B_e$	ID	Point-Interval Relations
<	<	<	<	1	{ < }
>	>	>	>	2	{ > }
>	<	>	<	3	{ d }
<	<	>	>	4	{ di }
<	<	>	<	5	{ o }
>	<	>	>	6	{ oi }
<	<	=	<	7	{ m }
>	=	>	>	8	{ mi }
=	<	>	<	9	{ s }

Table 1. *(continued)*

=	<	>	>	10	{ si }
>	<	>	=	11	{ f }
<	<	>	=	12	{ fi }
=	<	>	=	13	{ e }
=	=	>	>	14	{los}
<	<	=	=	15	{loe}
=	<	=	<	16	{ols}
>	=	>	=	17	{ole}
=	=	=	=	18	{oo}

The second part of the table has five special cases for bounded relations. The extended interval relations allow start point and end point of one interval is equal. For instance, we use *A ols B* to represent *A* is a non-divided atom interval (i.e., the starting and the ending points are located at the same position) and *B* is a interval that could contain other sub-intervals. Where *A* and *B* meets at the starting point of *A* (see Figure 1). These five special cases were not considered in [1]. The situations of points of two intervals could have unto $3^4 = 81$ rows in the above table. However, except for the 18 cases illustrated in Figure 1, others are conflict situations (i.e., it is physically impossible for the situation to occur). For example, a relation ($A_s < B_s$, $A_s < B_e$, $A_e > B_s$, $A_e < B_e$) has conflict between four endpoint relations.

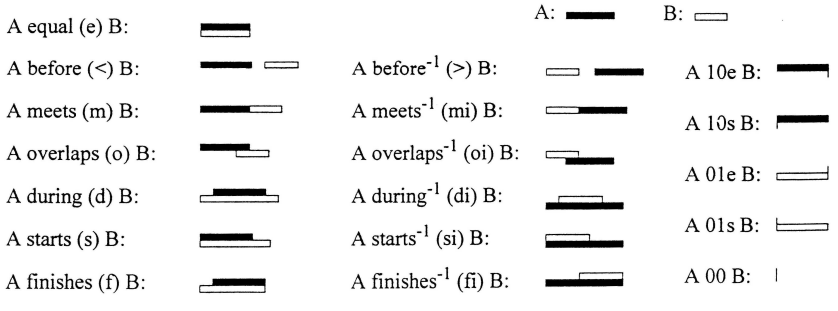


Fig. 1. The 18 Point-Interval Relations

For an arbitrary pair of points, *A* and *B*, located on a 1-dimensional line, there are three *point relations*: $A < B$, $A = B$, or $A > B$, for *A* is before *B, A* is at the same position as *B*, and *A* is after *B*, respectively. If these two points are located on a 2-dimensional plane, there exists nine (i.e., 3 * 3) cases. The X and the Y coordinates of these two points on the plane are independent. The possible relations between these two points on a plane can be denoted as *A* (<, <) *B*, *A* (<, =) *B*, *A* (<, >) *B*, *A* (=, <) *B*, *A* (=, =) *B*, *A* (=, >) *B*, *A* (>, <) *B*, *A* (>, =) *B*, and *A* (>, >) *B*, where the first element in the pair representing a point relation denotes the order on the X coordinate while the second is for the Y coordinate. Considering two line segments located on a 1-dimensional line, the situation becomes complicated. Since each line segment has a starting point and an ending point, we analyze the spatio-temporal relations of two

line segments according to these points. Allen's research is the special case of two line segments on the 1-dimensional line, with each line segment of length greater than zero.

Considering two line segments on a 2-dimensional plane, according to the above table and since the position of these two line segments are independent at the X and the Y coordinates, there exists $18^2 = 324$ possible relations between these two line segments on a plane. These relations, similar to those of two points on a plane, are denotes by pairs as: (< , <), (< , >), (< , *d*), (< , *di*), ..., (00 , 01e), and (00 , 00). We use these 324 binary relations to model spatial point-interval relations of two lines on a plane. Suppose X_A and X_B are the projection of two segments *A* and *B* (Figure 2). Y_A and Y_B are the projection on Y. X_A is {*start*} to X_B and Y_A is {*before*} to Y_B. We represent the spatial relation between A and B as (A, (s, <),B).

Relations of n-D objects can be used in object representation and recognition. A object in 3-D space can be projected onto *y*-*x*, *z*-*x*, and *y*-*z* planes. The projections correspond to surfaces generated from 3D objects. Similarly, a 2-D object is projected to *x* and *y* axes (Figure 5.5). If we look at two objects in the n-dimensional space, we can project the positional relation between these two objects from *n* directions to *n* 1-D space. The projections of 2-D object are x-interval and y-interval, but not the point. Thus, an n-dimensional relation can be formularized by a conjunction of *n* 1-D interval relations. A conjunction of two 1-D relations, which denotes a 2-D relation, has 13^2 variations, i.e. {(<, <), (<, >), (<, d),…, (=, fi), (=, =)} where the first element in the pair representing a interval relation denotes the order on the X coordinate while the second is for the Y coordinate. Similarly, there are 13^3 3-D relations.

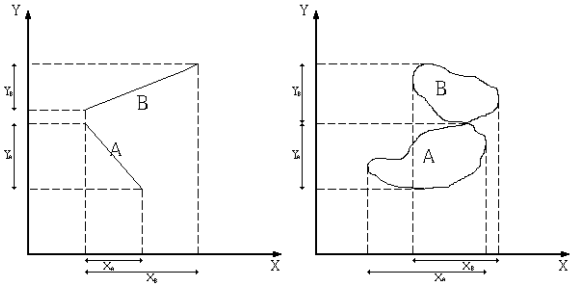


Fig. 2. Projection of two line segments and two 2-D objects a plane

3 Spatio-temporal Constraint Reasoning

We denote by *before*, *equal* and *after*(denoted as {<}, {=} and {>}) for representing relations between two points. Qualitative variables take these values only. The notation R_1 and R_2 denotes the point relations over three points *A*, *B*, and *C* which *A* R_1 *B* and *B* R_2 *C*. The meaning of some qualitative operators is defined as follows. Some values are uncertain and denoted by ⊤.

Table 2. Addition of qualitative point relations

R_1 \ R_2	<	=	>
<	<	<	T
=	<	=	>
>	T	>	>

In order to express more precise relations without losing qualitative information, the temporal relations extended with qualitative mechanisms for handling quantitative information. To give a concrete form to the topic of temporal representation, consider the following variables and equations with quantitative and qualitative information.

Since any interval relation between two intervals can be identified by four endpoint relations $A_s \otimes B_s$, $A_s \otimes B_e$, $A_e \otimes B_s$, and $A_e \otimes B_e$, the relations of temporal intervals can be identified by different endpoint space respectively. For example, if the endpoint relations based on the space {<, =, >} we can obtained the 18 well-defined interval relations after eliminating conflict situations. Similarly, if the endpoint relation space is {<, =, >, T}, we can derived the 29 transitive relations [8], and can convince the Interval Transitive Table which gives the transitive relations constraints is sound. Since at most 29 transitive relations out of the relevance of all the 2^{13} disjunction relations, we can maintain temporal knowledge efficiently when deciding an indefinite relation on a definite interval relation.

We develop an O(n)-time algorithm [8] for propagation temporal constraint between two time events. For solving point/interval algebra networks, we develop an algorithm for finding all pairs of feasible relations.

***Definition 3.1*: Formal Endpoint Relations with Quantity.**
A formal endpoint relation $Q_E = (E_R, V_E)$ is a quantitative-qualitative valuable.
where E_R, is an endpoint relation based on the point space {<, =, >}, and V_E is a quantity which expresses a quantitative value associated with E_R between two endpoints. ■

Let (R_1, v_1) and (R_2, v_2) denote two formal endpoint relations, R_1 and R_2 are two qualitative temporal variables, v_1 and v_2 are two quantity associated with quality. The meaning of some quantitative-qualitative calculus operators and equalities are defined as follows.

Table 3. Addition of qulitative-quantitative relations

(R_1,v_1) \ (R_2,v_2)	$<_{v2}$	=	$>_{v2}$
$<_{v1}$	$<_{v1+v2}$	$<_{v1}$	$<_{v1-v2}$, if (v1>v2) =, if (v1=v2) $>_{v2-v1}$, if (v1<v2)
=	$<_{v2}$	=	$>_{v2}$
$>_{v1}$	$>_{v1-v2}$, if (v1>v2) =, if(v1=v2) $<_{v2-v1}$, if(v1<v2)	$>_{v1}$	$>_{v1+v2}$

In addition, a quantitative-qualitative equation correctly expresses both qualitative equation and quantitative equations by formal endpoint variables and operators. We give a set of equations, for an example:

- Given $[d_A]$, $[d_B]$ and $[A_sB_s]$
  $[A_s\ B_e] = [A_s\ B_s] + [d_B]$
  $[A_e\ B_s] = -[d_A] + [A_sB_s]$
  $[A_e\ B_e] = -[d_A] + [A_sB_s] + [d_B]$

where $[d_A]$, $[d_B]$, $[A_s]$, $[B_s]$, $[A_e]$, and $[B_e]$ are expressing duration of *A*, duration of *B*, begin of *A*, begin of *B*, end of *A*, and end of *B* respectively. ■

Example 3.1: Considering three temporal interval A, B, and C, following requirements are just be known:

- The duration of *A* is 20-units length.
- The duration of *B* is 10-units length.
- The duration of *C* is 16-units length.
- Beginning of A is before beginning of *B* for 30 units.
- End of B is after beginning of *C* for 13 units.

In integrated temporal algebra, the information could be denoted as:$[d_A] = <_{20}$

$$[d_B] = <_{10}\ ,\ [d_C] = <_{16}\ ,\ [A_sB_s] = <_{30}\ ,\ [B_eC_s] = >_{13}$$

We could derive the complete temporal knowledge after following derivation.

Deriving formal endpoint relations:

$$[A_sB_e] = [A_sB_s] + [d_B] = <_{30} + <_{10} = <_{40}$$
$$[A_eB_s] = -[d_A] + [A_sB_s] = >_{20} + <_{30} = <_{10}$$
$$[A_eB_e] = -[d_A] + [A_sB_s] + [d_B] = >_{20} + <_{30} + <_{10} = <_{20}$$
$$[B_sC_s] = [d_B] + [B_eC_s] = <_{10} + >_{13} = <_{3}$$
$$[B_sC_e] = [d_B] + [B_eC_s] + [d_C] = <_{10} + >_{13} + <_{16} = <_{13}$$
$$[B_eC_e] = [B_eC_s] + [d_C] = >_{13} + <_{16} = <_{3}$$
$$[A_sC_s] = [A_sB_s] + [B_sC_s] = [A_sB_e] + [B_eC_s] = <_{27}$$
$$[A_sC_e] = [A_sB_s] + [B_sC_e] = [A_sB_e] + [B_eC_e] = <_{43}$$
$$[A_eC_s] = [A_eB_s] + [B_sC_s] = [A_eB_e] + [B_eC_s] = <_{7}$$
$$[A_eC_e] = [A_eB_s] + [B_sC_e] = [A_eB_e] + [B_eC_e] = <_{23}$$

Derivation of interval transitive relations:

Since each interval relations can be characterized by four endpoint relations, we can derive the qualitative interval relations using encoding technique enumerated.

- The quantity of endpoint relations (A_sB_s, A_sB_e, A_eB_s, A_eB_e) is (<, <, <, <) then we obtain (A, {<}, B).
- The quantity of endpoint relations (B_sC_s, B_sC_e, B_eC_s, B_eC_e) is (>, <, >, <) then we obtain (B, {d}, C).
- The quantity of endpoint relations (A_sC_s, A_sC_e, A_eC_s, A_eC_e) is (<, <, <, <) then we obtain (A, {<}, C).

Complete integrated temporal relations:

$$(A, \{<\}, B, <_{20}, <_{10}, <_{30}, <_{40}, <_{10}, <_{20}) \otimes (B, \{d\}, C, <_{10}, <_{16}, >_{3}, <_{13}, >_{13}, <_{3})$$
$$=(A, \{<\}, C, <_{27}, <_{43}, <_{7}, <_{23}, <_{7}, <_{23}) \quad \blacksquare$$

4 Spatio-temporal Approximate Measuring

If we consider two line segments or objects on a plane, based on the relative positions of two lines, it is feasible to construct a mechanism to compare the similarity between two polygons or two graphs with objects. Therefore, an evaluation mechanism is necessary to compute *relation similarity*.

Relations are similar to each other in certain degree. For example, "during" and "starts" are similar since the only difference is the starting points of the two intervals are different. However, "before" and the inverse of "meets" are not quite the same.

In Table 1, each of the 13 interval relations and 5 point-interval relations are defined by four "⊗" relations. These relations can be used as a base of our evaluation criterion. A *relational-distance* of two "⊗" relations belong to two different temporal relations occurs if those two temporal relations hold different relations in the same column of Table 1.

Definition 4.1: A *point relation distance* (PRD) defined with respect to a point relation r of index n have n incompatible differences from r. The following table gives a definition of point relation distance:

Table 4. Point Relation Distance (PRD)

PRD	>	=	<
>	0	1	2
=	1	0	1
<	2	1	0

Definition 4.2: An extended point-interval relation distance (EPIRD) defined with respect to a point-interval or interval relation r of index n have n incompatible differences from r. Let R and R′ are two interval relations or point-interval relations. The encoding point relation of R (see Table 1) is R_{AsBs}, R_{AsBe}, R_{AeBs}, R_{AeBe}, and the encoding point relation of R′ is R'_{AsBs}, R'_{AsBe}, R'_{AeBs}, R'_{AeBe}. We have a *EPIRD* formula:

$$EPIRD(R, R') = PRD(R_{As\otimes Bs}, R'_{As\otimes Bs}) + PRD(R_{As\otimes Be}, R'_{As\otimes Be}) + PRD(R_{Ae\otimes Bs}, R'_{Ae\otimes Bs}) + PRD(R_{Ae\otimes Be}, R'_{Ae\otimes Be})$$

The index of EPIRD with respect to each temporal relation in Table 1 can also be retrieved from the length of a shortest path in a *distance graph* (see Figure 3). For example, relations $\{d\}$ and $\{mi\}$ have an EPIRD index (i.e., the number of incompatible differences) of 3. Note that, in the distance graph, the shortest path between an arbitrary pair of nodes (i.e., relations) has a length between 1 and 8 since the distance graph is to represent the similarity of relations of two interval segments. Note that, in

Table 4. Extented Point-Interval Relation Distance between Two Interval Relations

EPIRD	<	>	d	di	o	oi	m	mi	s	si	f	fi	e	los	loe	ols	ole	oo
<	0	8	4	4	2	6	1	7	3	5	5	3	4	6	2	2	6	4
>	8	0	4	4	6	2	7	1	5	3	3	5	4	2	6	6	2	4
d	4	4	0	4	2	2	3	3	1	3	1	3	2	4	4	2	2	4
di	4	4	4	0	2	2	3	3	3	1	3	1	2	2	2	4	4	4
o	2	6	2	2	0	4	1	5	1	3	3	1	2	4	2	2	4	4
oi	6	2	2	2	4	0	5	1	3	1	1	3	2	2	4	4	2	4
m	1	7	3	3	1	5	0	6	2	4	4	2	3	5	1	1	5	3
mi	7	1	3	3	5	1	6	0	4	2	2	4	3	1	5	5	1	3
s	3	5	1	3	1	3	2	4	0	2	2	2	1	3	3	1	3	3
si	5	3	3	1	3	1	4	2	2	0	2	2	1	1	3	3	3	3
f	5	3	1	3	3	1	4	2	2	2	0	2	1	3	3	3	1	3
fi	3	5	3	1	1	3	2	4	2	2	2	0	1	3	1	3	3	3
e	4	4	2	2	2	2	3	3	1	1	1	1	0	2	2	2	2	2
los	6	2	4	2	4	2	5	1	3	1	3	3	2	0	4	4	2	2
loe	2	6	4	2	2	4	1	5	3	3	3	1	2	4	0	2	4	2
ols	2	6	2	4	2	4	1	5	1	3	3	3	2	4	2	0	4	2
ole	6	2	2	4	4	2	5	1	3	3	1	3	2	2	4	4	0	2
oo	4	4	4	4	4	4	3	3	3	3	3	3	2	2	2	2	2	0

the distance graph, there are five special edges illustrated in thick curved lines (i.e., from relation 00 to 10*e*, 01*s*, *e*, 10*s*, and 01*e*). This is due to the fact that, there exists no EPIRD element of index 1 of the 00 relation. A thick curved line has a weight of 2, instead of 1 as to the regular cases.

Also, let *EPIRD18*(r_i, r_j) be a EPIRD index function takes as input two relations, r_i, $r_j \in$ *18REL*, and returns a similarity index from 0 to 8. We have:

distance = *EPIRD18*(r_1, r_2)

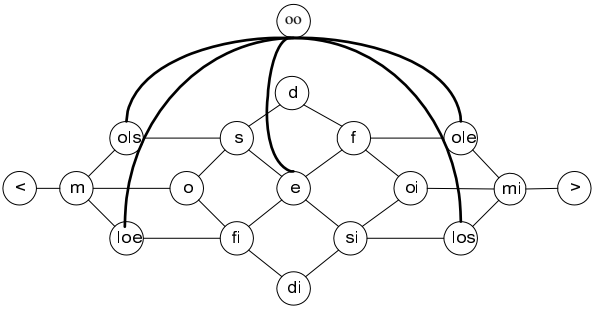


Fig. 3. Distance Graph of Well-defined Interval Relations

We have developed a fast computation mechanism to representation the interval relation distance. Using a bit-slicing representation of the relations, the index of image similarity can be computed in a few operations.

We use 10, 00, and 01 to represent the comparisons of end points of a line segment, >, =, and <, respectively. Eight bits are used to represent a relation of two line segments on a line. And 16 bits are used to represent a relation of two line segments on a plane. For instance, the two relations, *d* and *mi*, in the starting and ending point relation table in Table 1 can be represented as $(10\ 01\ 10\ 01)_2$ and $(10\ 00\ 10\ 10)_2$, respectively. An exclusive or logical operation is used to compute the *similarity syndrome*, which represents the difference with $(00\ 01\ 00\ 11)_2$.

An integer array, *A*, of 2^8 or 2^{16} elements (for either temporal or spatial relation, respectively) is used to simulate the EPIRD index function . The similarity syndromes (represented as unsigned integers) are indices to the array. For instance, *A*(00 01 00 11) = 3, which represents that the two relations, {*d*} and {*mi*}, have three incompatible error bits at the second and the fourth "⊗" relation. Therefore, one logical operator and one direct table lookup compute the relational-distance. The efficiency is very important in computing polygon similarity.

5 Applications for Shape Matching Using Approximations

2-D shape analysis is useful in a number of applications of machine vision, including aerial image analysis, medical image analysis and manufacturing.

In a spatial resource database, pictures are bitmapped images associated with *shape representation polygons* (SRPs). An image processing mechanism is used to compute the shapes of objects in a picture. Each picture has a set of SRPs. This SRP sets is that we based on to compute polygon similarity between a *query polygon* (QP).

If we consider two line segments on the plane, based on the relative positions of two lines, it is feasible to construct a mechanism to compare the similarity between two polygons since polygons are made of line segments. Therefore, an evaluation mechanism is necessary to compute *relation similarity*. Suppose the query polygon has *n* sides and a candidate shape representation polygon has *m* sides, where *n* and *m* are not necessarily equal. We have

Definition 5.1: Let
$SRP = \{(a_1\ S_1\ a_2), (a_2\ S_2\ a_3), ..., (a_{m-1}\ S_{m-1}\ a_m), \text{ and } (a_m\ S_m\ a_1)\}$ be a relational description with shape representation polygon SRS,

where a_i, $1 \le i \le n$, are the *n* sides of the query polygon, and $S_i \in$ *324REL* , $1 \le i \le n$, are relations of two line segments project on a plane. ■

Definition 5.2: Let
$QP = \{(b_1\ Q_1\ b_2), (b_2\ Q_2\ b_3), ..., (b_{m-1}\ Q_{m-1}\ b_m), \text{ and } (b_m\ Q_m\ b_1)\}$ be a relational description with query polygon QP,

where b_i, $1 \le i \le n$, are the *n* sides of the query polygon, and $Q_i \in$ *324REL* , $1 \le i \le n$, are relations of two line segments project on a plane. ■

Definition 5.3: Assume that A_s, A_e, B_s, and B_e are the starting and ending points of the two line segments on a plan, we want to define a *length ratio function*, $LR(r_{ix})$ and $LR(r_{iy})$:

$$LR(\mathrm{r_{ix}}) = (A_{\mathrm{e.x}} - A_{\mathrm{s.\,x}}) / (max(A_{\mathrm{e.x}}, B_{\mathrm{e.x}}) - min(A_{\mathrm{s.x}}, B_{\mathrm{s.x}}))$$
$$LR(\mathrm{r_{iy}}) = (A_{\mathrm{e.y}} - A_{\mathrm{s.y}}) / (max(A_{\mathrm{e.y}}, B_{\mathrm{e.y}}) - min(A_{\mathrm{s.y}}, B_{\mathrm{s.y}}))$$

where $A_{\mathrm{s.x}}$ and $A_{\mathrm{s.y}}$ are the X and the Y coordinates of the starting point of project line segment A. ■

Definition 5.4: Let function $sim(r_i, r_j)$ be a similarity function, which takes as input two relations, r_i, and r_j, and returns a similarity:

$$r_i = r_j \Rightarrow sim(r_i, r_j) = (LR(r_{ix})+LR(r_{iy})) - (LR(r_{jx})+LR(r_{jy})) \vee$$
$$r_i \neq r_j \Rightarrow sim(r_i, r_j) = EPIRD18(r_i, r_j) * (LR(r_{ix})+LR(r_{iy})) - (LR(r_{jx})+LR(r_{jy}))$$ ■

The similarity function, $sim(r_i, r_j)$, estimates the similarity between two 2-D project line segments or objects on a plan based on distance similarity index and the length ratio function. Based on the similarity function, we construct the similarity function for polygons and graphs with spatial relationships of objects.

Definition 5.5: The polygon and projection objects similarity function, *psim(QP, SRP)*, takes as input the query graph and the candidate representation graphs, and returns an integer:

$$psim(QP, SRP) = \sum_{\mathrm{i}=1}^{\mathrm{m}} sim(QP_{\mathrm{r.i}}, SRP_{\mathrm{r.i}})$$ ■

The system is able to handle 2-D shape analysis.

6 Conclusions

This paper constructs an algebra system of spatio-temporal interval relations and the set of enhanced mechanism for spatio-temporal relation composition. The temporal models are generalized by composing point temporal relations with qualitative and quantitative functions. Quantitative-qualitative physics is concerned with the dynamic behavior of the physical word. The spatio-temporal approximate measuring algorithm proposed in this paper can be used in other computer applications for comparing temporal relations and spatial objects. We hope that, the knowledge underlying spatio-temporal algebra can be used in many computer applications, especially in managing spatio-temporal knowledge and reasoning about time and space.

References

1. Allen, J.F.: Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26(11) (1983)
2. Patel, H., Hsu, W., Lee, M.L.: Mining Relationships among Interval-Based Events for Classification. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 393–404 (2008)
3. Vilain, M., Kautz, H.: Constraint Propagation Algorithms for Temporal Reasoning. In: Proceedings AAAI 1986, Philadelphia, pp. 377–382 (1986)
4. Wattamwar, S.S., Ghosh, H.: Spatio-Temporal Query for Multimedia Databases. In: Proceeding of the 2nd ACM Workshop on Multimedia Semantics 2008, pp. 48–55 (2008)

5. Ismail, A., Yan, J., Shen, J.: Verification of Composite Services with Temporal Consistency Checking and Temporal Satisfaction Estimation. In: Vossen, G., Long, D.D.E., Yu, J.X. (eds.) WISE 2009. LNCS, vol. 5802, pp. 343–350. Springer, Heidelberg (2009)
6. Miller, C., Quek, F.: Toward Multimodal Situated Analysis. In: Proceedings of the 13th International Conference on Multimodal Interfaces 2011, pp. 239–246. ACM Press (2011)
7. Hobbs, J.R., Pan, F.: An Ontology of Time for the Semantic Web. Transactions on Asian Language Information Processing (TALIP) 3(1), 66–85 (2004)
8. Shih, T.K., Chang, A.Y.: The Algebra of Spatio-Temporal Intervals. In: Proceedings of the 12th International Conference on Information Networking, Japan, January 21-23 (1998)

A Human-Like Agent Model for Attribution of Actions Using Ownership States and Inverse Mirroring

Jan Treur[1] and Muhammad Umair[1,2]

[1] VU University Amsterdam, Agent Systems Research Group
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
[2] COMSATS Institute of Information Technology, Lahore, Pakistan
{j.treur,m.umair}@vu.nl, mumair@ciitlahore.edu.pk

Abstract. This paper presents a neurologically inspired human-like agent model addressing attribution of actions to agents. It is not only capable of attribution of own actions to itself, but also to other agents, as for patients suffering from Schizophrenia. The mechanisms underlying the model involve ownership states and inverse mirroring to generate a mental image of the agent to which an action is attributed. The model is adaptive in that the inverse mirroring can develop based on Hebbian learning. The model provides a basis for applications to human-like virtual agents in the context of for example, training of therapists or agent-based generation of virtual stories.

Keywords: action attribution, cognitive agent model, ownership states, inverse mirroring, schizophrenia.

1 Introduction

To design human-like agent models, the fast growing amount of neurological literature is a useful source of information. For example, in this way virtual agents can be designed with a high extent of biological plausibility, which may show realistic shortcomings characteristic for humans. This paper contributes a human-like agent model for attribution of actions to agents. In the first place the modelled agent is able to attribute own actions to itself and other agents' actions to them. However, it is also possible for the agent model to display false attribution of own actions to other agents or other agents' actions to itself, as sometimes occurs in human agents, for example, in those who have symptoms of Schizophrenia; e.g., [6, 8, 9, 10, 21, 24]. Due to such variation possibilities, the model covers large parts of the variety in types of behaviour as occurring naturally in the overall human population.

In the neurological literature used as inspiration, two aspects are put forward as playing an important role in attribution of actions: (1) prediction of action effects, and (2) mirroring of actions of other agents. Concerning (1) it has been found that action effect prediction capabilities relate to proper attribution of own actions to oneself (e.g., [3], [7], [9], [10], [18] [21], [24]). Concerning (2), note that not attributing a self-generated action to oneself is not the same as attributing such an action to another agent. Actions may simply be not attributed to any agent (e.g., the wind may

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 574–585, 2012.

have caused it). To attribute an action to another agent, a mental image of somebody else performing the action has to be generated. When an action of another agent is observed, such a mental image is formed based on the incoming sensory information. However, when an own action is falsely attributed to another agent who may even be not present, forming this mental image requires a shift from a representation of an action from a first-person to a representation from a third-person perspective (mental rotation; e.g., [17]). This is the inverse operation of what happens in mirroring where a shift is made from a representation from a third-person to a representation from a first-person perspective; cf. [5], [14], [15], [16], [17], [20].

The human-like agent model presented in this paper is based on the perspective discussed above in relation to both (1) and (2). For (1) elements of the agent model for ownership introduced in [22] were adopted, and for (2) elements from the agent model for inverse mirroring introduced in [23]. A further question is how such a reverse mental rotation mapping can develop. This is modelled assuming a *Hebbian learning* principle: connected neurons that are frequently activated simultaneously strengthen their connecting synapse. In the cognitive agent model described below this principle has been adopted to realise an inverse mirroring connection from preparation of an action to sensory representation of a similar observed action. In the paper, in Section 2 the agent model is introduced. Section 3 presents some simulation results. Finally, Section 4 is a discussion.

2 The Cognitive Agent Model for Attribution of Actions

In this section the design of the cognitive agent model is presented. First an example scenario used is described, then modelling format used is introduced, and finally the agent model is addressed in detail.

Example Scenario. The designed agent model will be illustrated for the following scenario. Any sensed stimulus s leads to a sensory representation SR(s) of this stimulus, which in turn triggers the preparation state PA(a) of an action a as a response of the agent; see the causal chain from SR(s) to PA(a) in Fig. 1. The stimulus s can be any stimulus s1 from the world, but also a stimulus s2 which is the observation that another agent performs action a. In the former case, the arrow from SR(s1) to PA(a) models a reactive response of the agent triggered by stimulus s1. In the latter case the sensory representation SR(s2) indicates the mental image of another person performing the action a, and the arrow from SR(s2) to PA(a) models the agent's mirroring capability for action a; e.g., [14], [15], [20]. When this latter chain of events happens (i.e., whenever mirroring takes place), it is assumed that by Hebbian learning this will strengthen the reverse connection from preparation PA(a) to sensory representation SR(s2) (mental image of the observed action), thus developing inverse mirroring capabilities (the dotted arrow).When such a learning process has achieved substantial connection strength, the agent's response on stimulus s1 may have changed. When s1 is sensed (in the absence of s2), not only will the agent trigger preparation (and execution) of action a as before, but in addition it will generate a mental image of another agent performing action a (the sensory representation SR(s2)), thus creating a third person perspective on the action.

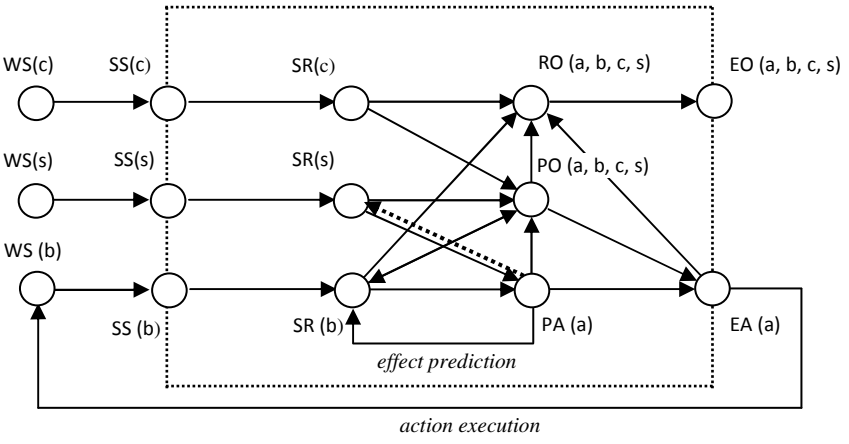


Fig. 1. Overview of the cognitive agent model

In the model s denotes a stimulus, c a context, a an action, and b a world state affected by the action. Examples of contexts are another agent B, or the agent self. The effect state b is considered to be positive for the agent (e.g., in accordance with a goal). The state properties used in the model are summarised in Table 1. The cognitive agent model distinguishes prior and retrospective ownership states for actions, indicated by PO(a, b, c, s) and RO(a, b, c, s), respectively (see Fig. 1). These states are taken specific for a given action a, effect b, context c, and stimulus s (triggering preparation of a). When the context c is self, an ownership state for c indicates self-ownership attribution, whereas for context c another agent B, it indicates ownership attributed to B. Note that the stimulus s triggering preparation of action a can be of any type; for social scenarios, it can be taken as a body state (e.g., face expression) of the other agent B. An action effect state b can be any state of the world (possibly including body states).

Table 1. State properties used

Notation	Description
WS(W)	world state W (W is a context c, stimulus s, or effect b)
SS(W)	sensor state for W
SR(W)	sensory representation of W
PA(a)	preparation for action a
EA(a)	execution of action a
PO(a, b, c, s)	prior ownership state for action a with b, c, and s
RO(a, b, c, s)	retrospective ownership state for a with b, c, and s
EO(a, b, c, s)	communication of ownership of a with b, c, and s
cs(ω)	strength of connection ω (from preparation of a to sensory representation of s2)

The prior ownership state PO(a, b, c, s) is affected by the preparation state PA(a) for the action a, the sensory representation SR(b) of the (predicted) effect b, the sensory representation SR(s) of the stimulus s, and the sensory representation SR(c) of the context c; see the four arrows to PO(a, b, c, s) in Figure 1. Similarly, the retrospective

ownership state RO(a, b, c, s) is affected by the sensory representation SR(c) of the context c, the sensory representation SR(b) of the effect b of the action, the prior ownership state PO(a, b, c, s), and the execution EA(a) of the action a; see the arrows to RO(a, b, c, s) in Fig. 1. Action prediction is modelled by the connection from the action preparation PA(a) to the sensory representation SR(b) of the effect b. Suppression of the sensory representation of the effect is modelled by the (inhibiting) connection from the prior ownership state PO(a, b, c, s) to sensory representation SR(b). The control exerted by the prior ownership state (similar to a super miror neuron function; e.g., [13], [15]) is modelled by the connection from PO(a, b, c, s) to EA(a). Finally, acknowledging of ownership is modelled by the connection from the retrospective ownership state RO(a, b, c, s) to the communication effector state EO(a, b, c, s).

Connections between state properties (the arrows in Fig. 1) have weights ω_k, as indicated in Table 2. Here LP refers to the (temporally) Local Properties LP1 to LP10 presented below. A connection weight ω_k has a value between *-1* and *1* and may depend on the specific context c, stimulus s, action a and/or effect state b involved. By varying these connection strengths, different possibilities for the repertoire offered by the model can be realised. Note that usually connection weights are assumed non-negative, except for the inhibiting connections, such as ω_{2o} which models suppression of the sensory representation of effect b (so that you cannot tickle yourself; cf. [3]).

Table 2. Overview of the connections and their weights

from states	to state	weights	process	LP
SS(W)	SR(W)	ω_1	representing world state W: stimulus s1 or context c	LP1
SS(s2), PS(b)	SR(s2)	ω_1, ω	representing observed action / inverse mirroring	LP2
PA(a), PO(a, b, self, s), SS(b)	SR(b)	ω_2, ω_{2o}, ω_3	representing effect state e	LP3
SR(s), SR(b)	PA(a)	ω_4, ω_5	action preparation/mirroring	LP3
SR(c), SR(s), SR(b), PA(a)	PO(a, b, c, s)	ω_6, ω_7, ω_8, ω_9	prior ownership	LP4
PO(a, b, self, s), PA(a)	EA(a)	ω_{10}, ω_{11}	action execution	LP5
EA(a)	WS(b)	ω_{12}	action effect	LP6
WS(W)	SS(W)	ω_{13}	sensing world state	LP7
SR(c), SR(b), PO(a, b, c, s), EA(a)	RO(a, b, c, s)	ω_{14}, ω_{15}, ω_{16}, ω_{17}	retrospective ownership	LP8
RO(a, b, c, s)	EO(a, b, c, s)	ω_{18}	expressed ownership	LP9
SR(s2), PA(a)	cs(ω)	η, ζ	learning inverse mirroring	LP10

Below, the dynamics are described in more detail, following the connections between the states in Fig. 1. This is done for each state by a dynamic property specifying how the activation value for this state is updated based on the activation values of the states connected to it (the incoming arrows in Fig. 1). The cognitive agent model has been computationally formalised in this way using the hybrid modeling language LEADSTO; cf. [4]. During processing, each state property has a strength represented by a real number between *0* and *1*; variables *V* (possibly with

subscripts) run over these values. In dynamic property specifications, this is added as a last argument to the state property expressions (an alternative notation activation(p, V) with p a state property has not been used for the sake of notational simplicity).

Below, f is a function for which different choices can be made, for example, the identity function $f(W) = W$ or a combination function based on a continuous logistic threshold function of the form

$$th(\sigma, \tau, X) = \left(\frac{1}{1+e^{-\sigma(X-\tau)}} - \frac{1}{1+e^{\sigma\tau}}\right)(1 + e^{-\sigma\tau})$$

with σ a steepness and τ a threshold value. Note that for higher values of $\sigma\tau$ (e.g., σ higher than $20/\tau$) this threshold function can be approximated by the expression $th(\sigma, \tau, X) = \frac{1}{1+e^{-\sigma(X-\tau)}}$. In the example simulations, for the states that are affected by only one state (i.e., in LP1, LP7, LP8, LP10), f is taken the identity function $f(W) = W$, and for the other states f is a combination function based on the logistic threshold function: $f(X_1, X_2) = th(\sigma, \tau, X_1+X_2)$, and similarly for more arguments. Other types of combination functions might be used as well. The first property LP1 describes how sensory representations are generated for context c and stimulus s1 (together indicated by variable W), and for stimulus s2, which is the action a performed by another agent.

LP1 Sensory representation for world state W: stimulus s_1 or context c

If the sensor state for W has level V_1
and the sensory representation of W has level V_2
then after duration Δt the sensory representation of W will have
level $V_2 + \gamma [f(\omega_1 V_1) - V_2] \Delta t$.

$SS(W, V_1)$ & $SR(W, V_2) \rightarrow SR(W, V_2 + \gamma [f(\omega_1 V_1) - V_2] \Delta t$

LP2 Sensory representation for stimulus s_2 indicating another agent's action a

If the sensor state for s_2 has level V_1
and preparation of a has level V_2
and the sensory representation of s_2 has level V_3
then after duration Δt the sensory representation of s_2 will have
level $V_3 + \gamma(f(\omega_1 V_1, \omega V_2) - V_3) \Delta t$.

$SS(s2,V_1)$ & $PA(a,V_2)$ & $SR(s2, V_3) \rightarrow SR(s2, V_3 + \gamma (f(\omega_1 V_1, \omega V_2) - V_3) \Delta t)$

The sensory representation of an effect state b as described by property LP2 is not only affected by a corresponding sensor state for b (which in turn is affected by the world state), as in LP1, but also by two action-related states:

- via the predictive loop by a preparation state, to predict the effect b of a prepared action a
- by an inhibiting connection from the prior self-ownership state, to suppress the sensory representation of the effect b of the action a, once it is initiated (e.g., [3], [8])

This is expressed in dynamic property LP3. Note that for this suppressing effect the connection weight ω_{2o} from prior ownership state for action a to sensory representation for effect b is taken negative, for example $\omega_{2o} = -1$.

LP3 Sensory representation for an effect state

If the preparation state for action a has level V_1
and the prior self-ownership of action a for b, self, and s has level V_2
and the sensor state for state b has level V_3
and the sensory representation of state b has level V_4
then after duration Δt the sensory representation of state b will have level $V_4 + \gamma[\, f(\omega_2 V_1, \omega_{2o} V_2, \omega_3 V_3) - V_4 \,]\, \Delta t$.

$PA(a, V_1)$ & $PO(a, b, self, s, V_2)$ & $SS(b, V_3)$ & $SR(b, V_4)$
$\rightarrow SR(b, V_4 + \gamma [\, f(\omega_2 V_1, \omega_{2o} V_2, \omega_3 V_3) - V_4 \,] \Delta t)$

Preparation for action a is affected by a sensory representation of stimulus s (triggering the action), and also strengthened by predicted effect b of the action:

LP4 Preparing and mirroring for an action

If sensory representation of s has level V_1 and sensory representation of b has level V_2
and the preparation for action a has level V_3
then after duration Δt the preparation state for action a will have level $V_3 + \gamma[\, f(\omega_4 V_1, \omega_5 V_2) - V_3 \,]\, \Delta t$.

$SR(s, V_1)$ & $SR(b, V_2)$ & $PA(a, V_3)$
$\rightarrow PA(a, V_3 + \gamma [\, f(\omega_4 V_1, \omega_5 V_2) - V_3 \,] \Delta t)$

Prior ownership of an action a is generated by LP5.

LP5 Generating a prior ownership state

If the sensory representation of context c has level V_1
and the sensory representation of s has level V_2
and sensory representation of b has level V_3
and the preparation for action a has level V_4 and prior ownership of a for b, c, and s has level V_5
then after duration Δt prior ownership of a for c, s, and b will have level $V_5 + \gamma\, [\, f(\omega_6 V_1, \omega_7 V_2, \omega_8 V_3, \omega_9 V_4) - V_5 \,]\, \Delta t$.

$SR(c, V_1)$ & $SR(s, V_2)$ & $SR(b, V_3)$ & $PA(a, V_4)$ & $PO(a, b, c, s, V_5)$
$\rightarrow PO(a, b, c, s, V_3 + \gamma [\, f(\omega_6 V_1, \omega_7 V_2, \omega_8 V_3, \omega_9 V_4) - V_3 \,]\, \Delta t)$

In case the context c is self, the prior ownership state strengthens the initiative to perform a as a self-generated action: executing a prepared action depends on whether a prior self-ownership state (for the agent self) is available for this action. This models control over the execution of the action (go/no-go decision) and can, for example, be used to veto the action in a late stage of preparation. This is modelled by LP6.

LP6 Action execution

If prior ownership of a for b, self, and s has level V_1
and preparation for action a has level V_2
and the action execution state for a has level V_3
then after Δt the action execution state for a will have level $V_3 + \gamma\, [\, f(\omega_{10} V_1, \omega_{11} V_2) - V_3 \,]\, \Delta t$.

$PO(a, b, self, s, V_1)$ & $PA(a, V_2)$ & $EA(a, V_3)$
$\rightarrow EA(a, V_3 + \gamma [\, f(\omega_{10} V_1, \omega_{11} V_2) - V_3 \,]\, \Delta t)$

Property LP7 describes in a straightforward manner how execution of action a affects the world state b.

LP7 From action execution to effect state

If the execution state for action a has level V_1 and
world state b has level V_2
then after Δt world state b will have
level $V_2 + \gamma [f(\omega_{12}V_1) - V_2] \Delta t$.

$EA(a, V_1)$ & $WS(b, V_2) \rightarrow\!\!\!\!\rightarrow WS(b, V_2 + \gamma [f(\omega_{12}V_1) - V_2] \Delta t)$

The following property models how sensor states are updated. It applies to stimulus s1, s2 effect b, and context c (indicated by variable W).

LP8 Generating a sensor state for a world state

If world state W has level V_1 and
the sensor state for W has level V_2
then after Δt the sensor state for W will have level $V_2 + \gamma [f(\omega_{13}V_1) - V_2] \Delta t$.

$WS(W, V_1)$ & $SS(W, V_2) \rightarrow\!\!\!\!\rightarrow SS(W, V_2 + \gamma [f(\omega_{13}V_1) - V_2] \Delta t)$

A retrospective ownership state takes into account the prior ownership, the execution of the action, the context, and the sensory representation of the action's effect:

LP9 Generating a retrospective ownership state

If the sensory representation of context c has level V_1,
and the sensory representation of effect state b has level V_2
and prior ownership of a for b, c, and s has level V_3
and the execution state for action a has level V_4
and retrospective ownership of a for b, c, and s has level V_5
then after Δt retrospective ownership of a for b, c, and s will have
level $V_5 + \gamma [f(\omega_{14}V_1, \omega_{15}V_2, \omega_{16}V_3, \omega_{17}V_4) - V_5] \Delta t$.

$SR(c,V_1)$ & $SR(b,V_2)$ & $PO(a, b, c, s, V_3)$ & $EA(a, V_4)$ & $RO(a, b, c, s, V_5)$
$\rightarrow\!\!\!\!\rightarrow RO(a, b, c, s, V_5 + \gamma [f(\omega_{14}V_1, \omega_{15}V_2, \omega_{16}V_3, \omega_{17}V_4) - V_5] \Delta t))$

Note that LP9 applies to context c that can be self as context, but also another agent B. For another agent as context the connection strength ω_{17} in LP9 is assumed *0* or negative; in the simulated scenarios discussed in Section 3 it was taken $\omega_{17} = -1$. The communication to attribute authorship (to any context c) depends on the retrospective ownership state as specified in LP10.

LP10 Communication of ownership awareness

If retrospective ownership of a for b, c, and s has level V_1,
and communication of a for b, c, and s has level V_2
then after duration Δt communication of a for b, c, and s will have
level $V_2 + \gamma [f(\omega_{18}V_1) - V_2] \Delta t$.

$RO(a, b, c, s, V_1)$ & $EO(a, b, c, s, V_2)$
$\rightarrow\!\!\!\!\rightarrow EO(a, b, c, s, V_2 + \gamma [f(\omega_{18}V_1) - V_2] \Delta t)$

Finally, it is shown in LP11 how the Hebbian learning process of the connection from preparation state for b to sensory representation s2 of an observed action was modelled. This takes place using the following *Hebbian learning rule*, with maximal connection strength *1*, a *learning rate* η, and *extinction rate* ζ (usually taken small):

$$\Delta\omega = \gamma[\ \eta V_1 V_2\ (1 - \omega) - \zeta\omega\ ]\ \Delta t$$

Here V_1 and V_2 are (time-dependent) activation levels of the connected nodes, and γ is an adaptation speed factor. In differential equation format it can be written as

$$\frac{d\omega}{dt} = \gamma[\ \eta V_1 V_2\ (1 - \omega) - \zeta\omega\ ] = \gamma[\ \eta V_1 V_2\ - (\eta V_1 V_2 + \zeta)\ \omega\ ]$$

A similar Hebbian learning rule can be found in [11], p. 406. By the factor $(1 - \omega)$ the learning rule keeps the level of ω bounded by *1*. When the extinction rate is relatively low, the upward changes during learning are proportional to both V_1 and V_2 and maximal learning takes place when both are *1*. Whenever one of them is close to *0*, extinction takes over, and ω slowly decreases. This is specified as follows:

LP11 Learning for inverse mirroring

If the sensory representation of stimulus s2 has level V_1,
and the preparation for a has level V_2,
and the connection weight from preparation for a to sensory representation of s2 has level W,
then after duration Δt the connection weight from preparation for b to sensory representation of s2 will have level $W + \gamma[\ \eta V_1 V_2\ (1 - W) - \zeta W\ ]\ \Delta t$.

$$SR(s2, V_1)\ \&\ PA(a, V_2)\ \&\ cs(\omega, W) \rightarrow cs(\omega, W + \gamma[\ \eta V_1 V_2\ (1 - W) - \zeta W\ ]\ \Delta t)$$

3 Simulation Results

This section presents some simulation results for the model described in Section 2. A number of simulations have been performed with the focus of simulating *normal functioning* and *deviant functioning* of the model. Moreover its effect with the case of an agent having a *poor action prediction capability* and *satisfactory prediction capability* is modeled and results are presented here (see also [22]), relating to *deviant functioning* and *normal functioning*, respectively. For the simulation results shown in Figures 2 and further, time is on the horizontal axis and the activation level of the state properties on the vertical axis. The initialized connection strengths between different states for *normal functioning* are shown in Table 3 below.

Table 3. Overview of the connections and their weights

Connection	ω_1	ω_2	ω_{2_0}	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	ω_9
Self	1	0.8	-0.6	0.5	0.8	0.8	1	1	1	1
Other	1	-	-	-	-	-	1	1	1	1
	ω_{10}	ω_{11}	ω_{12}	ω_{13}	ω_{14}	ω_{15}	ω_{16}	ω_{17}	ω_{18}	ω_{19}
Self	1	1	1	1	1	1	1	1	1	1
Other	-	-	-	1	-	-	-	-0.4	-	-

These values are kept fixed throughout the simulation, except the connection strength ω which is initialized with *0* and is adapted over time by the Hebbian learning rule given in LP10 in Section 2. Other parameters are set as $\Delta t = 0.1$, learning

rate $\eta = 0.3$, extinction rate $\zeta = 0.05$, speed factor $\lambda = 0.001$. A relatively slow value *0.3* for the update speed parameter γ was applied for external processes (action execution, effect generation and effect sensing) modelled by LP6, LP7, and LP8, and a fast value *0.6* for γ for the internal processes modelled by the other LP's. Threshold and steepness values for different states are given in Table 4.

Table 4. Overview Steepness and Threshold values

State	SR(b)	PA(a)	PO (other)	PO (self)	RO (other)	RO (self)	EA(a)	EO (other)	EO (self)
Steepness (σ)	4	4	8	8	20	20	20	40	40
Threshold (τ)	0.1	0.8	4	3	0.87	3.6	1.5	0.6	0.8

For the initial duration of 50 time units the stimulus s2 for the observed action occurs three times for 250 time units alternatively, i.e., for the first 50 time units the world state for s2 has value *1* and for the next 250 time units value *0*, and so on (see Fig. 2) to generate the similar scenario as described for inverse mirroring case in Section 3. During these 900 time units the world state for context self was kept *0* (see Fig. 2). This represents the situation in which a person observes somebody else performing some action (or bodily change) and the mirroring function of the preparation state makes the person prepare for this action for him or herself.

The fluctuation in the activation level of the sensor state repeats the same pattern between *0.1* to *0.9* as it only depends (via LP8) upon the world state for observed action, which also is repetitive. Due to space limitation those graphs are not included here but Fig. 3 shows how the sensory representation for the observed action reacts to the situation described above. For this particular case , as the stimulus for observed action remain present for a very short time i.e 50 time points, after attaining higher activation level, SR(s2) start declining immediately and as WS(self) is *0* during this time hence SR(self) remains *0*. Consequently prior other-ownership i.e PO(a, b, other, s) did not attain reasonably higher activation level whereas retrospective other-ownership i.e RO(a, b, other, s) keep on fluctuating similar to SRS(other). Moreover prior self-ownership i.e PO(a, b, self, s) and retrospective self-ownership i.e RO(a, b, self, s) remain almost *0* during this time (see Fig. 4 and Fig. 5). Due to quite low activation of both *self* and *other* prior-ownership (Fig. 4), one does not observe any action execution for *a* during this time.

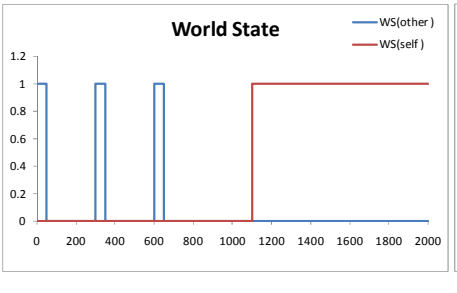


Fig. 2. World States

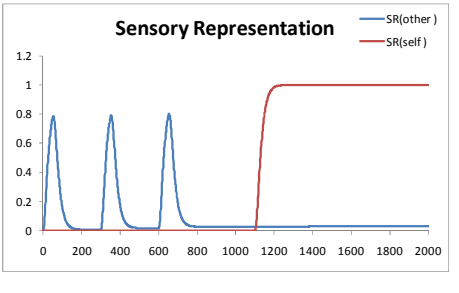


Fig. 3. Sensory Representation

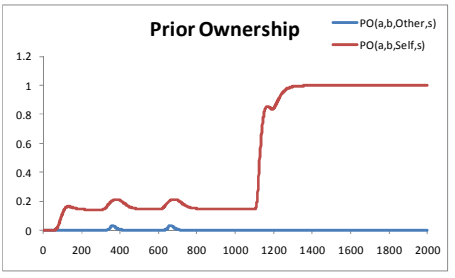


Fig. 4. Prior Ownership

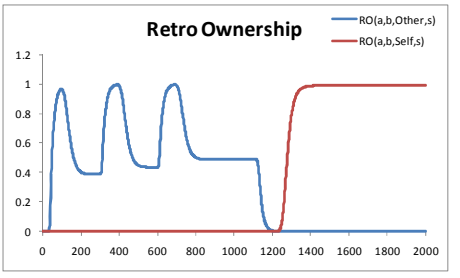


Fig. 5. Retrospective Ownership

As communication of ownership state is directly related to the retrospective ownership, it shows similar fluctuating activation level as of RO(a, b, c, s) . Furthermore the inverse link from preparation state for action a to the sensory representation for the observed action i.e. SR(s2) is not strengthened during this phase which reflects the *normal functioning* of the agent (see Fig.6). Thus in the absence of WS(other), SR(other) also remain zero (see Fig. 6)

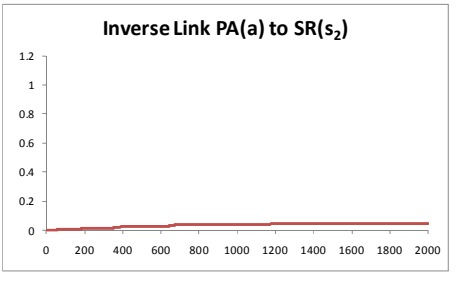


Fig. 6. Normal Functioning

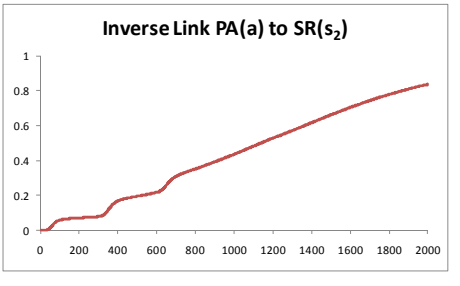


Fig. 7. Deviant Functioning

After that, for 200 time units both the world states stimulus for *self* and *other* are kept *0* from time point 900 to 1100, so that the effect of any stimulus on different states is neutralized as shown in Fig. 2. Then after 200 time points the world state for *self* WS(self) is set to 1 while keeping WS(other) at *0*. As from now on WS(self) remains *1*, high activation levels for sensory representation for *self*, prior self-ownership and retrospective self-ownership occur. This in turns produces high activation levels of action execution for a, i.e., EA(a). Similar behavior can be observed for communication of ownership.

Now to simulate the deviant behavior, again all parameters were initialized with the same values as used to simulate normal behavior shown in Table 3 and Table 4 earlier except the extinction and learning rate: $\eta = 2$, $\zeta = 0.01$ respectively. In contrast to the previous results, by using these parameter settings the inverse link from preparation state for action a to sensory representation of observed action is learnt substantially during first 900 time units as shown in Fig. 7. Due to this higher connection strength of the inverse link, SR(s2) also gains a higher activation level, even in the absence of WS(s2) from 900 onwards (see Fig. 8).

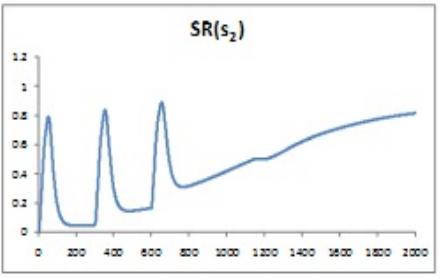


Fig. 8. Sensory Representation for Observed Action

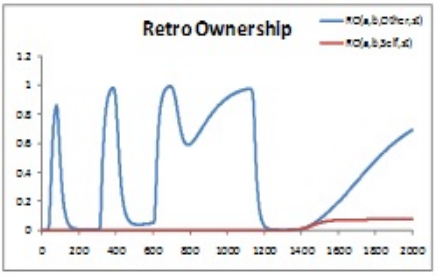


Fig. 9. Retrospective Ownership

Thus, it results in increasing the value of the retrospective other-ownership, i.e. RO(a, b, other, s) to *0.69* (see Fig. 9). Hence an agent develops a mental image of somebody else performing action a and the same is communicated based on the high activation level of the retrospective ownership state.

4 Discussion

The human-like agent model presented in this paper incorporates two mechanisms that play an important role in attributing actions to agents. In the first place it exploits prior and retrospective ownership states for an action based on principles from recent neurological theories; this was adopted from [22]. In the second place the agent model incorporates an adaptive inverse mirroring mechanism (adopted from [23]) to generate mental images of an agent to whom an action is attributed.

The obtained human-like agent model can be used as a basis for the design of virtual agents in simulation-based training or in gaming. For example, a virtual patient model can be developed based on the presented model so that, for example, a psychiatrist or psycho-therapist (e.g., during his or her education) can gain insight in the processes in certain types of patients, or it can be used by a therapist to analyse how a certain form of therapy can have its effect on these processes.

References

1. Barringer, H., Fisher, M., Gabbay, D., Owens, R., Reynolds, M.: The Imperative Future: Principles of Executable Temporal Logic. John Wiley & Sons (1996)
2. Bi, G.Q., Poo, M.M.: Synaptic Modifications by Correlated Activity: Hebb's Postulate Revisited. Ann. Rev. Neurosci. 24, 139–166 (2001)
3. Blakemore, S.-J., Wolpert, D.M., Frith, C.D.: Why can't you tickle yourself? Neuroreport 11, 11–16 (2000)
4. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by Simulation. Intern. J. of AI Tools 16, 435–464 (2007)
5. Damasio, A., Meyer, K.: Behind the looking-glass. Nature 454, 167–168 (2008)

6. Farrer, C., Franck, N.: Self-monitoring in schizophrenia. Curr. Psychiatry Rev. 3, 243–251 (2007)
7. Feinberg, I.: Efference copy and corollary discharge: Implications for thinking and its disorders. Schizophrenia Bulletin 4, 636–640 (1978)
8. Fourneret, P., de Vignemont, F., Franck, N., Slachevsky, A., Dubois, B., Jeannerod, M.: Perception of self-generated action in schizophrenia. Cogn. Neur. 7, 139–156 (2002)
9. Frith, C.D., Blakemore, S., Wolpert, D.: Explaining the symptoms of schizophrenia: Abnormalities in the awareness of action. Brain Research Rev. 31, 357–363 (2000)
10. Frith, C.D.: The cognitive neuropsychology of schizophrenia. Lawrence Erlbaum Associates Ltd., Hove (1992)
11. Gerstner, W., Kistler, W.M.: Mathematical formulations of Hebbian learning. Biol. Cybern. 87, 404–415 (2002)
12. Hebb, D.O.: The Organization of Behaviour. John Wiley & Sons, New York (1949)
13. Hendriks, M., Treur, J.: Modeling Super Mirroring Functionality in Action Execution, Imagination, Mirroring, and Imitation. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part I. LNCS (LNAI), vol. 6421, pp. 330–342. Springer, Heidelberg (2010)
14. Hurley, S.: The shared circuits model: How control, mirroring, and simulation can enable imitation, deliberation, and mind reading. Behavioral and Brain Sciences 31, 1–22 (2008)
15. Iacoboni, M.: Mirroring People: the New Science of How We Connect with Others. Farrar, Straus & Giroux (2008)
16. Jeannerod, M.: The sense of agency and its disturbances in schizophrenia: a reappraisal. Exp. Brain Res. 192, 527–532 (2009)
17. Jeannerod, M., Anquetil, T.: Putting oneself in the perspective of the other: A framework for self-other differentiation. Social Neuroscience 3, 356–367 (2008)
18. Moore, J., Haggard, P.: Awareness of action: Inference and prediction. Consciousness and Cognition 17, 136–144 (2008)
19. Port, R.F., van Gelder, T. (eds.): Mind as Motion: Explorations in the Dynamics of Cognition. MIT Press (1995)
20. Rizzolatti, G., Sinigaglia, C.: Mirrors in the Brain: How Our Minds Share Actions and Emotions. Oxford Univ. Press (2008)
21. Synofzik, M., Thier, P., Leube, D.T., Schlotterbeck, P., Lindner, A.: Misattributions of agency in schizophrenia are based on imprecise predictions about the sensory consequences of one's actions. Brain 133, 262–271 (2010)
22. Treur, J.: A Cognitive Agent Model Incorporating Prior and Retrospective Ownership States for Actions. In: Walsh, T. (ed.), Proc. IJCAI 2011, pp. 1743-1749 (2011)
23. Treur, J., Umair, M.: A Cognitive Agent Model Using Inverse Mirroring for False Attribution of Own Actions to Other Agents. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 109–119. Springer, Heidelberg (2011)
24. Voss, M., Moore, J., Hauser, M., Gallinat, J., Heinz, A., Haggard, P.: Altered awareness of action in schizophrenia: a specific deficit in predicting action consequences. Brain 133, 3104–3112 (2010)

Fault Resolution Support Based on Activated Knowledge and Information

Kazuto Sasai, Gen Kitagata, and Tetsuo Kinoshita

Research Institute of Electrical Communication, Tohoku University
Kitahira 2-1-1, Aoba-ku, Sendai, Miyagi, Japan
kazuto@riec.tohoku.ac.jp

Abstract. An intelligent network management system consists of multiple distributed agent is important issue to maintain recent complex network infrastructures. However, to realize the efficient support of the management task of the administrators, various knowledge and information with respect to the managed network have to be utilized and integrated in the system. In this paper, we propose the practical design method and implementation of the Active Information Resource based Network Management System (AIR-NMS) which consists of activated information resources in a distributed network environment. The AIR-NMS realize not only the autonomic management facilities but also the effective support for administrator. The effectiveness of an implemented prototypical AIR-NMS is evaluated by experiments conducted on an experimental network environment. The information resource oriented design is shown to reduce the burden for administrators by the support to utilize and manage various information and knowledge in an autonomic way.

Keywords: AIR-NMS, active information resource, network management system, autonomic computing, multiagent system.

1 Introduction

Intelligent applications supporting network / service management are required to maintain various IT services and their infrastructures because the structures of systems grow complexity. A distributed agents based network management system [2] is a basic concept to build the solutions of network management systems (NMSs). Currently, the autonomic computing paradigm inspired from biological systems have been proposed[3]. Although there have been many studies to improve the autonomic features of NMS [4], it still remains a difficult problem how to manage the systems. As our body sometimes needs a doctor, the network systems need the human administrators, in order to develop the security policies and to enhance the performance. Since the administrators should understand the network status and the control method to manage the network systems, the NMSs must have the functions to efficiently deal with the status information and to manage the knowledge about control method. However, the agents do not have

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 586–595, 2012.

framework to deal with information and knowledge because the concept focuses on the autonomic behavior [1]. Thus, new techniques for building /developing an autonomic NMSs to efficiently utilize the knowledge and information.

Hence, we have studied and proposed a new concept of distributed information resources called Active Information Resource (AIR) [5], and proposed the AIR-based NMS (AIR-NMS) [7] to overcome the above mentioned problem. The AIR-NMS consists of two types of AIRs, I-AIR and K-AIR, where the former manages status information of various network elements, and the latter manages network management heuristics of human administrators. In this paper, we propose and discuss the practical design and implementation of an AIR-NMS focusing on the network fault resolution task, which is an important problem to overcome for human administrators. In Section 2, the essential concept of designing the AIR-NMS is presented. Section 3 introduces the practical design of the AIR-NMS focusing on the fault resolution tasks. The implementation of a prototype system and evaluation experiments are demonstrated in Section 4. Finally, Section 5 concludes this paper.

2 Design Concept of AIR-NMS

An Active Information Resource (AIR) [5] is an extended entity of a distributed electric information resources, based on the active feature supported by the Knowledge of Utilization Support (KUS) and the Function of Utilization Support (FUS). The KUS consists of meta-level knowledge, i.e., knowledge for handling information resources and cooperation knowledge with other AIRs. The active function of an AIR is supported by FUS, which consists of various functions to process its information resources and for communication between AIRs. By applying the AIR concept to distributed academic information materials, the realized AIRs provide intelligent retrieval and shared functions for academic users [6].

Generally, the administrators have to deal with many kinds of distributed information allocated in various distributed elements of the network. Using the AIR concept, a distributed information resource can be extended to an AIR by attaching KUS and FUS, which have been designed with respect to the distributed information of the network. The realized AIRs for network management tasks can support a part of the management tasks of human administrators, and as a result, the burden of the administrators can be reduced in a systematic way [7].

The essential functions of an AIR-NMS are as follows:

(1) Functions to accumulate, manage and utilize distributed status information of managed networks, such as fault detection, monitoring and traffic measurement.
(2) Functions to deal with various problem solving in network management tasks, such as network fault resolution and management operation.

To realize functions (1) and (2), we have designed and implemented a network status Information AIR (I-AIR) and a network management Knowledge AIR

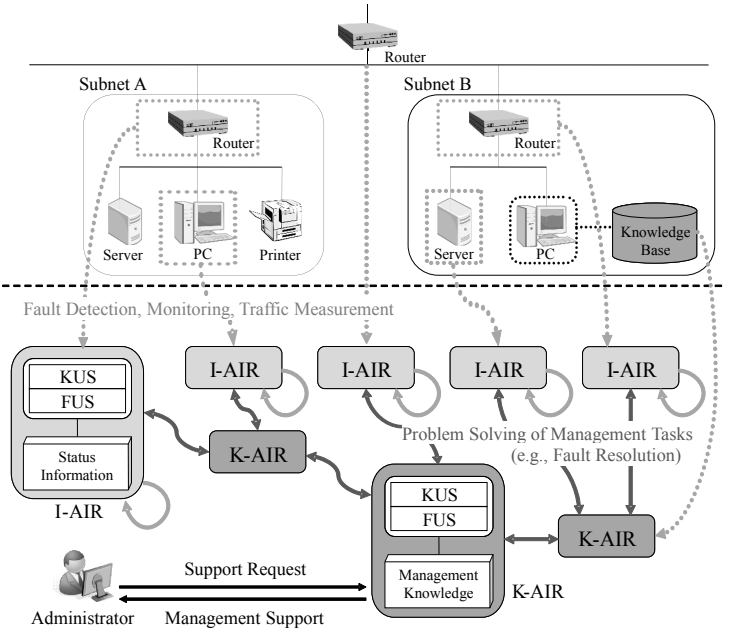


Fig. 1. Schematic diagram of AIR-NMS

(K-AIR), respectively. Hence, an AIR-NMS is designed and implemented using I-AIRs and K-AIRs as shown in Fig. 1 [9].

These AIRs deal with the network management task together with an administrator in two ways. The administrator can send a support request to a K-AIR, and receive a response of support (Request-based driven mode), or a K-AIR can receive information of the detected event from an I-AIR, and then cooperate with other AIRs to solve the problem with respect to the event (Alarm-based driven mode). To realize effective cooperation among AIRs based on the above two operation modes, we have to provide a practical design method of the AIR-NMS for the administrators. In the next section, we describe the design and implementation of a prototype of an AIR-NMS for a network fault resolution task to demonstrate a practical design method of AIR-NMSs.

3 Building an AIR-NMS for Network Fault Resolution

The design process of the AIR-NMS is defined as follows:

1. Definition of knowledge and functions of the required AIRs with respect to the management tasks.
2. Design of knowledge of AIRs.
3. Design of functions of AIRs.

```
<sc symptom="unable to send mail">
  <cause>unable to resolve name</cause>
  <cause>network connection failure</cause>
  <cause>over sendable size limit</cause>
</sc>
```

(a). K_{SC}: Cause assumption.

```
<cd cause="over sendable size limit">
  <dm>
    <p>request #//sent_mail_size# from #source#</p>
    <p>request #//client/mta/servername# from #source#</p>
    <p>request #//sendable_size_limit# from #//client/mta/servername#</p>
    <p>true(#//sendable_size_limit# -lt #//sent_mail_size#)<p>
  </dm>
  <dr>
    Sent mail size, #//sent_mail_size#KB, is over sendable size limit,
    #//sendable_size_limit#KB, which is set to SMTP server
    #//server/mta/name#.
  </dr>
</cd>
```

(b). K_{CD}: Cause diagnosis.

```
<cm cause="over sendable size limit">
  <m>
    Setting at SMTP server #//client/mta/servername# can be changed,
    because its OS is
    #//host/os/name@//client/mta/servername#=(CentOS,Fedora)
    and its MTA is #//server/mta/name@//client/mta/servername#=(Postfix, Sendamail, Qmail)
          ,
    by following operations ...
  </m>
</cm>
```

(c). K_{CM}: Means planning.

Fig. 2. Examples of described network management knowledge of the K-AIRs

4. Implementation of agents corresponding to AIRs.
5. Integration of AIRs to build an AIR-NMS.

According to the agent-oriented design of AIRs, a prototypical AIR-NMS is designed in this section, focusing on the network fault resolution task.

3.1 Design of K-AIRs for Network Fault Resolution Tasks

The management knowledge used in these subtasks are defined as the combination of the following knowledge description elements, *Symptom*, *Cause*, *Diagnosis method* and *Means*. Here, we design the following three types of K-AIRs for the fault resolution task.

A K_{SC}-AIR deals with management knowledge K_{SC} to assume the conceivable causes from an observed symptom. A K_{SC} consists of a *Symptom* and the conceivable *Causes* assumed to be the origin of the symptom. Fig. 2(a) shows an example description of K_{SC} for the case of the symptom "unable to send mail," which specifies the faulty situation that a user cannot send mail. A K_{SC}

is structured by the tag <sc> which has an attribute "symptom" indicating a subject of a symptom, i.e., "unable to send mail" is set to "symptom". Assumed conceivable causes is designated by the tags <cause>, and in this example, three conceivable causes are specified for the symptom "unable to send mail", i.e., "unable to resolve name", which specifies the cause with respect to the error of DNS, "network connection failure," which specifies the cause with respect to a network connection error, and "over sendable size limit", which specifies the size of the sent mail exceeding the mail size limit. The FUS of the K_{SC}-AIR provides the functions to send the message to other K_{SC}-AIRs and K_{CD}-AIRs where the observed symptom can be handled.

A K_{CD}-AIR deals with management knowledge K_{CD} to diagnose and verify a cause assumed by a K_{SC}-AIR. A K_{CD} consists of three descriptions, a *Cause*, a *Diagnosis Method* to verify the cause and a *Diagnosis Report* to notify the result of the diagnosis to the administrator. Fig. 2(b) shows an example description of a $\mathbf{K}_{CD}$ for the cause "over sendable size limit," which specifies the particular situation that the size of a mail sent by a user exceeds the configuration of mail size limit. A K_{CD} is structured by the tag <cd> which has an attribute "cause" indicating a subject of a cause, i.e., "over sendable size limit" is set to "cause." The tag <cd> has the child tags <dm> and <dr>, that specify a *Diagnosis Method* and a *Diagnosis Report*, respectively. The child tag <p> of the tag <dm> indicates the diagnosis processes that are sequentially executed in order to verify the condition of the cause. The FUS of the K_{CD}-AIR provides the functions to correct information from the I-AIRs. It sends messages to the K_{CM}-AIRs, and presents a diagnosis report to an administrator when assumed cause can be handled.

Finally, a K_{CM}-AIR deals with management knowledge K_{CM} to recommend the *Means* for a detected cause to the administrators. A K_{CM} consists of a *Cause* and the *Means* for the cause. Fig. 2(c) shows an example description of a K_{CM} for the cause "over sendable size limit." A K_{CM} is structured by the tag <cm> which has an attribute "cause" indicating a subject of a cause, i.e., "over sendable size limit" is set to "cause." The FUS of the K_{CM}-AIR provides th efunctions to plan the means for a detected cause, and presents the mwans to an administrator when a detected cause can be handled.

Here, according to the above difinition of AIRs, the following three kinds of messages, Msg-S, Msg-C and Msg-I, are defined as the message exchange scheme between AIRs. The Msg-S is a support message of a fault resolution task that an administrator asks the AIR-NMS to get. The Msg-C is used in the following two situations. The first is to represent a diagnosis request from a K_{SC}-AIR to the K_{SC}-AIRs and K_{CD}-AIRs, and the second is to represent a means planning request from a K_{CD}-AIR to the K_{CM}-AIRs. Note that Msg-C is sent not only to the K_{CD}-AIRs but also to the K_{SC}-AIRs. This is useful to elaborate the problem. The Msg-I is used to send an information acquisition request from the K-AIRs to the I-AIRs.

```
<si>
  <host>
    <name>srvA</name>
    <subnet>subnetA</subnet>
    <domain>example.com</domain>
    <fqdn>srvA.subnetA.example.com</fqdn>
    <nic name="eth0">
      <ipadress>172.16.0.2</ipaddress>
      <netmask>255.255.255.0</netmask>
      <status>up</status>
    </nic>
    <link destination="172.16.0.1" ping_status="reachable"/>
  </host>
  ...
</si>
```

(a). Status information of I_H-AIR.

```
<host>
  <name>WMI</name>
  <subnet>Command_Windows</subnet>
       <Link>Command_Windows</Link>
  ...
</host>

<Command_Windows>
  <key>/link</key>
  <methodname>ping</methodname>
  <ok_checkString>0% loss</ok_checkString>
  <false_checkString>100% loss</false_checkString>
  <checkinfo>
    <check>exist</check>
    <ok_exist>reachable</ok_exist>
    <false_exist>unreachable</false_exist>
  </checkinfo>
  ...
</Command_Windows>
```

(b). Utilization support knowledge KUS of I_H-AIR.

Fig. 3. Examples of the description of I_H-AIR

3.2 Design of I-AIRs to Cooperate with K-AIRs

The fault diagnosis and resolution functions of the AIR-NMS are realized based on the flexible provision of the I-AIRs that handle various information of network equipment. To realize the flexible provision of the I-AIRs, we have designed an I-AIR, called I_H-AIR which can be instantiated onto various type of hosts and respond to various type of requests from the K-AIRs. In this section, we explain the design of KUS and FUS of the I_H-AIR to cooperate with the K-AIRs in a fault resolution task.

Here, we design a set of I_H-AIRs which deals with status information of hosts, servers and clients. Fig. 3(a) shows an example of status information of a host, which is specified by attributes, such as hostname, subnet, OS, NIC, etc. In a network fault resolution tasks, we often need temporal status information such as an occurrence time of the fault. The KUS in an I_H-AIR should include knowledge to acquire, analyze and integrate temporal information. Although the original behavior of an I-AIR discussed in [8] is more complex, in this paper, it is designed

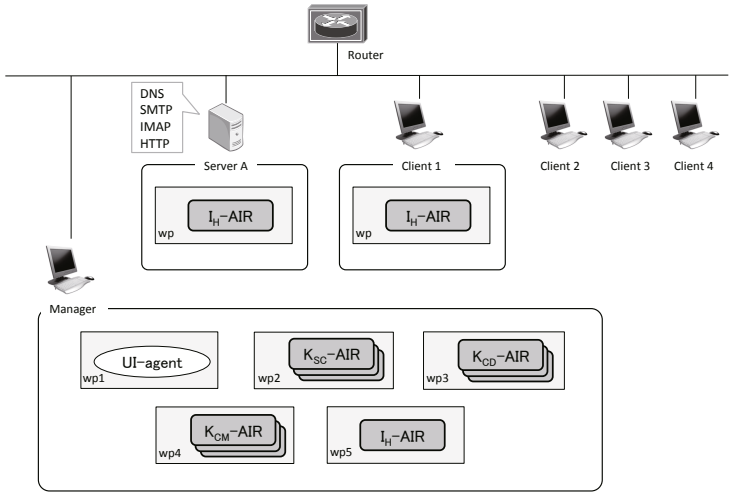


Fig. 4. Configuration diagram of prototypical system

focusing on the functions related to cooperation with the K-AIRs. The FUS of the I_H-AIR consists of two kinds of functions, to communicate the K-AIRs and to acquire the status information required by the K-AIRs. The AIR-NMS is realized by allocating the designed AIRs to network environment. In the next section, we explain the implemented prototype system.

4 Experiments and Evaluation

We implement a prototype system of an AIR-NMS for network fault resolution using the repository-based multiagent framework ADIPS/DASH [10], as explained in Section 3. The AIRs of the AIR-NMS are designed as various agents stored in the agent repository. The ADIPS/DASH framework provides a runtime environment for agents, called the workplace, which operates on a distributed platform such as a PC allocated over the networked environment. The agents are instantiated from the agent repository onto the workplaces based on the requests of users, in order to execute the distributed problem solving tasks.

Fig. 4 shows the configuration of the prototypical AIR-NMS over the experimental network environment, which consists of four client PCs, one server PC, and one administrator PC, that are depicted as Client1-4, ServerA and Manager, respectively. ServerA runs CentOS 5 with BIND 9.3.6 for DNS, Postfix 2.3.3 for SMTP, Dovecot 1.0.7 for IMAP, and Apache 2.2.3 for HTTP server. Client1-4 and Manager runs Windows XP OS.

At runtime of the AIR-NMS, the agents of both the I-AIRs and the K-AIRs are instantiated onto the distributed workplace of various PCs of the experimental network. Although the K-AIRs can operate on distributed workplaces in the prototype system, all K-AIRs are instantiated onto a workplace of the administrator PC.

To test the capabilities of the prototypical AIR-NMS, we conduct a fault resolution experiment with actual human users. The experimental network is the same as the previous experiment, with user playing the role of an administrator to resolve the given faulty situation. Thereafter, we observe and compare the results of the operations by human users, called "manual-based operations," with those supported by AIR-NMS, called "AIR-NMS-based operations."

We record the behavior of the user by two cameras and capture the screens of the PCs to movies. In this experiment, ServerA is implemented as a virtual machine on the Manager for convenience. The user playing the role as an administrator can operate any PC in the experimental environment.

The experimental procedure of the manual-based operation is described as follows:

Step1. An experimenter set up a faulty situation in the experimental network.
Step2. The experimenter asks the user to resolve the given faulty situation by telling a symptom and a site of fault occurence.
Step3. The user can operate all of the network's management functions in the experimental environment, e.g., internet search, windows utilities, Linux/Unix utilities, etc.
Step4. The consumption time of the experiment is defined as the duration until the user recovers the given symptom.

In the AIR-NMS-based operation, the experimenter is allowed to teach the user how to use the AIR-NMS in Step2.

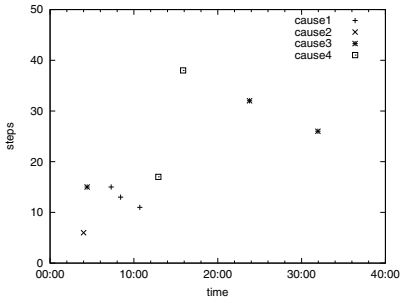
Five students of Graduate School of Information Science, Tohoku University, participate in the experiment as the users. All of the students have basic knowledge of network technologies, however, only two of them are familiar with network management tasks, and the others are not.

The following four cases are selected as the faulty situations:

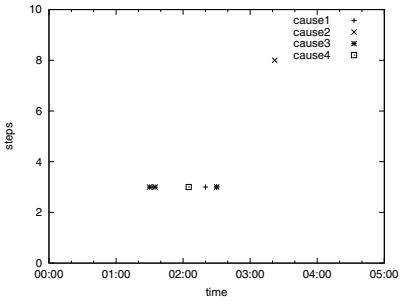
Cause1. HTTP server process down.
Cause2. Link missing of target Web page.
Cause3. DNS server process down.
Cause4. HTTP port is closed.

Cause1 implies a situation where a HTTP process of ServerA is down, Cause2 implies that the link of a Web page on the ServerA is missing because the target Web page is removed, Cause3 implies that a DNS server process on ServerA is down, and Cause4 implies that a HTTP port of ServerA is closed. We select several cases for respective users, and the users resolve the faults in two ways, manual-based and AIR-NMS-based operation.

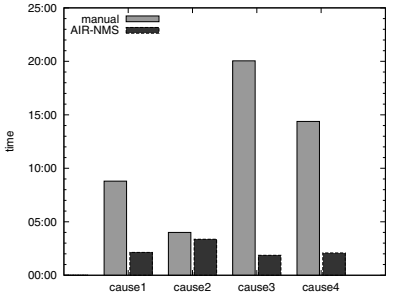
We summarize the results of the experiment in Fig. 5. Fig. 5(a) and Fig. 5(b) show the distributions of sample data of manual-based and AIR-NMS-based operation, respectively. The consumption time and the operation steps of the fault resolution tasks are depicted on horizontal and vertical axis, respectively. We can see that the time and steps depend on the types of conceivable causes, and the users can resolve the faults in pretty much the same time and steps



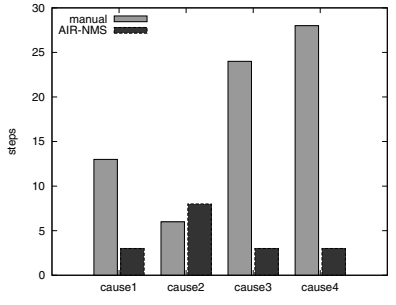
(a) Manual-based operation.



(b) AIR-NMS-based operation.



(c) Consumption time.



(d) Operation steps.

Fig. 5. Results of experiment with human administrators

by the AIR-NMS-based operation. It implies that an AIR-NMS can effectively support the users by guiding them in the fault resolution process. Note that the two figures are depicted with different scale. Fig. 5(c) and Fig. 5(d) show the comparison between manual and AIR-NMS-based operation with the time and steps, respectively. The AIR-NMS shows its effectiveness in most of the cases, but the result of Cause2 is almost the same because its conceivable cause "missing link" is comparatively easier than the other.

In this section, we conducted the fault resolution experiment with the prototypical system to validate the effectiveness of the proposed practical design of the AIR-NMS. From the experiment, it is confirmed that the problem solving capabilities of the AIR-NMS for the fault resolution task can be realized by the cooperative problem solving behavior of the AIRs. The results of the time and steps for the fault resolution tasks imply that the AIR-NMS can efficiently guide the administrators to the means of the fault resolution. For unskilled users, the AIR-NMS dramatically reduce consumption time and workload. Actually, it shows an average 80% reduction of time and steps for the case of an unskilled user. On the other hand, for skilled users, an average 70% reduction of time and steps is realized with only one exception. It implies that the AIR-NMS effectively supports the fault resolution task not only for unskilled users but also for skilled users to greater and lesser degrees. For the distributed large scale network systems, software multi agent nature of the AIR design is effective because the

load of the individual AIR does not increase dramatically compared with other existing centralized NMSs. As described in the previous sections, the effective support function is realized by autonomous cooperation between K-AIR and I-AIR that are agent-based designs of KUS and FUS for management knowledge (K-AIR) and status information (I-AIR). Hence, the proposed practical design of the AIR-NMS is useful and effective to realize the AIR-NMS, which can deal with a part of network management tasks to reduce the burden of administrators.

5 Conclusion

Focussing on the fault resolution task of network administrators, the practical design and implementation of an AIR-based NMS (AIR-NMS) is proposed and evaluated in this paper. Through the experiment, using a prototypical AIR-NMS, the AIR-NMS shows the capabilities of resolving faulty situations based on the cooperation of AIRs to support human administrators. It remains as future work to extend the capabilities of the AIR-NMS to deal with many kinds of faults and anomalies of network system by introducing various AIRs.

References

1. Nwana, H.S.: Software Agents: An overview. Knowledge Engineering Review 11(3), 205–244 (1996)
2. Bieszczad, A., Pegurek, B., White, T.: Mobile Agents for Network Management. IEEE Communication Surveys 1(1), 2–2 (1998)
3. Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing. IEEE Computer 36(1), 41–50 (2003)
4. Samaan, N., Karmouch, A.: Towards Autonomic Network Management: An Analysis of Current and Future Research Directions. IEEE Communications Surveys & Tutorials 11(3), 22–36 (2009)
5. Li, B., Abe, T., Sugawara, K., Kinoshita, T.: Active Information Resource: Design Concept and Example. In: AINA 2003, pp. 274–277 (2003)
6. Li, B., Kinoshita, T.: Active Support for Using Academic Information Resources in Distributed Environment. International Journal of Computar Science and Network Security 7(6), 69–73 (2007)
7. Konno, S., Iwaya, Y., Abe, T., Kinoshita, T.: Design of Network Management Support System based on Active Information Resource. In: AINA 2004, pp. 102–106 (2004)
8. Konno, S., Abar, S., Iwaya, Y., Kinoshita, T.: Effectiveness of Autonomous Network Monitoring based on Intelligent-Agent-Mediated Status Information. In: Okuno, H.G., Ali, M. (eds.) IEA/AIE 2007. LNCS (LNAI), vol. 4570, pp. 1078–1087. Springer, Heidelberg (2007)
9. Takahashi, Y., Misugi, D., Sakatoku, A., Satoh, A., Takahashi, A., Sasai, K., Kitagata, G., Abe, T., Kinoshita, T.: Knowledge Oriented Network Fault Resolution Method Based on Active Information Resource. In: Proceedings of WI/IAT 2010, pp. 361–364 (2010)
10. Kinoshita, T., Sugawara, K.: ADIPS Framework for Flexible Distributed Systems. In: Ishida, T. (ed.) PRIMA 1998. LNCS (LNAI), vol. 1599, pp. 18–32. Springer, Heidelberg (1999)

Creating User's Knowledge Space from Various Information Usages to Support Human Recollection

Harumi Murakami[1], Kenta Mitsuhashi[2], and Keita Senba[1]

[1] Graduate School for Creative Cities, Osaka City University,
3-3-138, Sugimoto, Sumiyoshi, Osaka 558-8585 Japan
harumi@media.osaka-cu.ac.jp
http://murakami.media.osaka-cu.ac.jp/
[2] OGIS-RI Co.,Ltd.,
3-Minami 2-37, Chiyozaki, Nishi, Osaka 550-0023 Japan

Abstract. To support human recollection, we present a data integration method using a simple information structure called a history structure, which is constructed from time, keywords, and URI sets. We also present algorithms that generate history structures from such information usages as Web searches, twitter, e-mails, calendars, and book purchases and create a user knowledge space. Based on our approach, we developed a system called a knowledge-space browser and evaluated whether it can help users recall a particular day by summarizing that day's history structure. Experimental results reveal the usefulness of our approach and our implemented system.

Keywords: knowledge space, history structure, human recollection, web search, twitter, e-mail, calendar.

1 Introduction

Memory is crucial for various activities. We think and feel using memory. We need to recall past memories of particular periods. For example, we may have to write progress reports about what we have done on particular days or weeks. We may have to plan anniversaries and recall what we did last year or in previous years. Or we might simply want to reminisce about the day when we saw our spouse for the first time. We aim to help users recall the past from particular periods.

We propose an approach that gathers pieces of the past memory of a day and visualizes them as a knowledge space to help recollection. We present a method of data integration using a simple information structure called a history structure, which is constructed from time, keywords, and URI sets. A history structure is simply generated from existing information sources. In this paper, we present a method that generates history structures from such information usages as web searches, twitter, e-mails, calendars, and book purchases, and create a user knowledge space from them.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 596–605, 2012.

Below, we explain our proposed algorithms in Sections 2 and 3. The implementation and examples of the knowledge-space browser are described in Section 3. Our experiments are described in Section 4. We discuss the significance of our research in Section 5.

2 Generating History Structure

2.1 Gathering Information from Various Information Usages

We gather texts that have time information and express user thoughts or activities to help them recall their memories.

Web Search. We believe that Web search histories often express user interests and are thus related to their thoughts or activities. We use Web search histories (Google queries and search results) to gather information that expresses user interests. For queries, we extract a time of Web search, query, and query URI by dividing the query using spaces and generate keywords. For search results, we extract a time of browsing, browsed page's title, and page's URI and generate keywords from the title of the browsed page using the algorithm described in Section 2.2.

Twitter. Since tweets generally express the user's thoughts or activities, we use all of them except for those starting with @ because they are mainly discourse and official RTs (Retweets) because they are mainly other's opinions. We extract a tweet time, tweet, and its URI and generate keywords from the tweet using the algorithm described in Section 2.2.

E-mail. E-mails sometimes express the user's thoughts or activities. We use the receivers and the subjects of the sent messages because sent messages often express thoughts or activities. We do not use received messages, because most are direct mails that are not related to the user activities. We extract the time e-mail was sent, its receivers and subject, and its URI. For receivers, we simply extract names and addresses. For subjects, we generate keywords using the algorithm described in Section 2.2.

Calendar. Calendars or diaries are obviously useful sources for user activities. We extract a "start time of event, event title, and event URI by simply dividing the event title into keywords using spaces and adding the original event title as keywords. We believe that the original event title is meaningful for users.

Book Purchases. Product purchases sometimes help users remember thoughts or activities. In this research, we focus on book purchases because book contents are probably related to user knowledge. We extract a time of order/purchase, title, and ISBN of the book. We generate keywords from the book title using the algorithm described in Section 2.2.

2.2 Generating Keyword Algorithm

We developed a generating keyword algorithm that creates a set of keywords from such texts as the titles of browsed Web pages, tweets, the subjects of sent e-mails, event titles in calendars, and the titles of books purchased.

Our algorithm extracts noun phrases, adjectives, verbs, and non-Japanese terms with MeCab, a Japanese morphological analysis tool [1], which outputs non-Japanese terms as nouns. See Fig. 1.

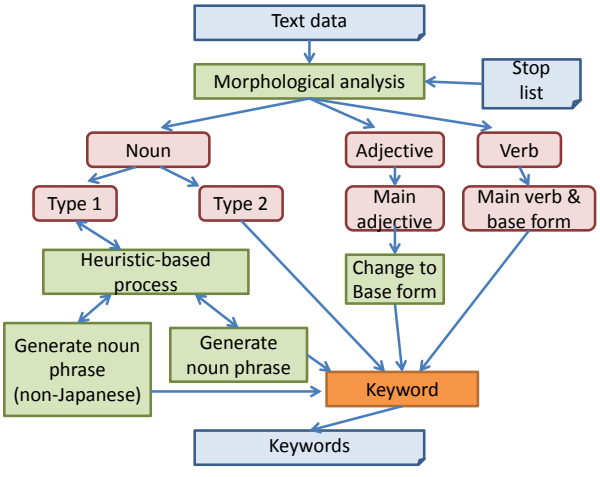


Fig. 1. Algorithm of generating keywords

When a term is a noun, a common noun, a proper noun, a noun verbal, a noun suffix, or a noun number (type 1), it is repeatedly concatenated with previous terms as a non-Japanese keyword or as a Japanese keyword using heuristics. When the noun is a noun adverbial or a noun adjective base (type 2), it directly becomes a keyword. When a term is an adjective and its type is a main adjective, the base form becomes a keyword. When a term is a verb, its type is a main verb, and it is equal to the base form, the base form of the term becomes a keyword. The detailed algorithm is described in [2].

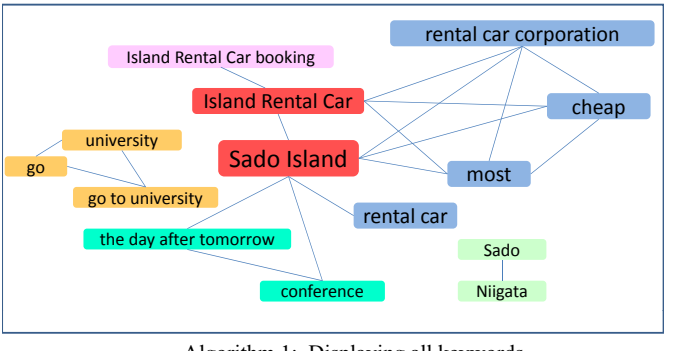
3 Generating Knowledge Space

3.1 Generating Knowledge-Space Algorithm

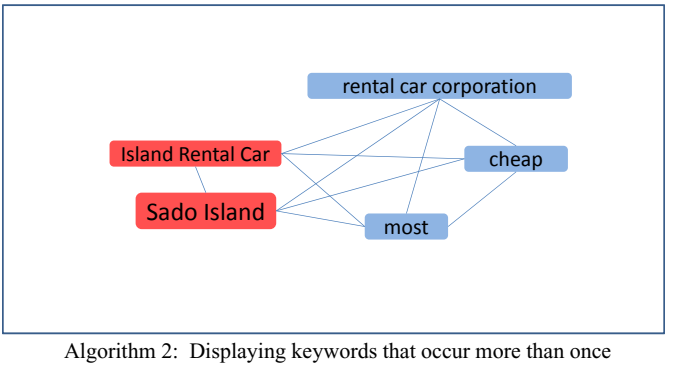
The idea of knowledge space is based on semantic networks that represent semantic memory in cognitive psychology [4]. We believe that displaying a user's knowledge space like semantic networks helps user recollection.

History Structure

Time	Keywords	URI	Note
Mon 13 00:00:00 JST 2010	Go to university, go, university	http://www.google.com/calendar/...	Calendar
Mon 13 00:28:00 JST 2010	Niigata, Sado	ISBN-10:4398153187	Book purchase
Mon 13 15:19:14 JST 2010	Sado island, rental car	http://www.google.com/search?q...	Web browsing (query)
Mon 13 16:06:18 JST 2010	Island Rental Car, Sado island, most, cheap, rental car corporation	http://park2wakwak.com/...	Web browsing (title of browsed page)
Mon 13 17:14:33 JST 2010	The day after tommorow, conference, Sado island	http;//twitter.com/…	Twitter
Mon 13 19:07:55 JST 2010	Island Rental Car, Sado island, most, cheap, rental car corporation	http://park2.wakwak.com/...	Web browsing (title of browsed page)
Mon 13 19:09:35 JST 2010	Island Rental Car, Island Rental Car booking	https://mail.google.com/mail...	Mail



Algorithm 1: Displaying all keywords



Algorithm 2: Displaying keywords that occur more than once and have one or more relations.

Fig. 2. Algorithm of generating knowledge space

The basic algorithms for generating knowledge space connect the keywords that co-occurred in the history structures. Clusters generated by natural connections help users recall their past.

The basic algorithm is as follows:

- Connect keywords that co-occurred in history structures.
- Keywords that occurred frequently in the history structure are displayed as larger.
- Different colors correspond to different information sources: light blue for Web searches, aquamarine for twitter, pink for e-mails, yellow for calendars, and purple for book purchases. These five colors for each information source were designed through a preliminary investigation.
- Keywords that occurred in multiple information sources are emphasized in red.

We designed and implemented two algorithms: displaying all keywords (visualization algorithm 1) and displaying keywords that occur more than once in history structures and have one or more relations (visualization algorithm 2). Fig. 2 shows an overview of the two algorithms. This example is part of the next section.

3.2 Knowledge-Space Browser

We implemented a knowledge-space browser that is comprised of four parts: (a) a knowledge space display that shows the user's knowledge space, (b) a history structure display that lists the user's history structures, (c) an operation display on which the user manages the knowledge space, and (d) a selected keyword display that lists the history structures on which the selected keyword is included. For the knowledge space, users can change either visualization algorithm 1 or 2 at anytime.

Fig. 3 is an example screen of a user's knowledge space generated by visualization algorithm 1 on 13/Sep/2010. He prepared to go to a conference that will be held on 15/Sep/2010 on Sado Island. See Fig. 2. His calendar shows his plan to go to the university to prepare his presentation. He bought a guide book of Sado Island in Niigata prefecture. He searched "Sado Island, rental car" and browsed a page of a "rental car corporation named Island Rental Car" that is denoted as the cheapest on the island. He tweeted "I will go to Sado Island for a conference the day after tomorrow." He booked a rental car at Island Rental Car by e-mail. The biggest cluster in the knowledge space contains *Sado Island*, *Island Rental Car*, *Sado*, *conference*, in red. This cluster shows the main activities of the user for that day. The biggest (i.e. the most frequent) keyword is *trial version* in a different cluster. Such keywords as *Photoshop* and *download* are included in this cluster, which shows that the user was preparing a presentation for the conference using photoshop.

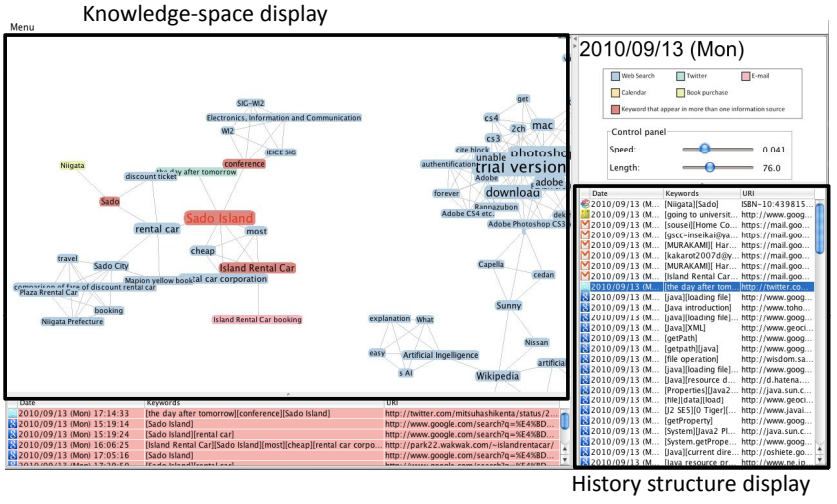


Fig. 3. Example of knowledge-space browser

4 Experiment

4.1 Experiment 1

We evaluated the usefulness of our algorithm that extracts keywords.

Method. The subjects were 10 male, computer and information science graduate students aged 22-24.

We generated history structures from five information usages (Web searches, twitter, e-mails, calendars, and book purchases) using four methods. The comparative method (a) extracts only nouns, (b) extracts nouns and concatenates them using TermExtract [3], which is a Japanese noun phrase extraction tool that orders the extracted Japanese phrases in importance, or (c) only extracts noun phrases by our proposed method. Our method (d) extracts concatenated noun phrases, adjectives, and verbs.

The subjects evaluated whether the generated keywords are appropriate among five information usages and four methods by five values (5: very appropriate; 4; appropriate; 3: OK; 2: not very appropriate; 1: inappropriate). Each subject evaluated 10 history structures * 5 information usages * 4 methods: 200 history structures.

Results and Discussion. Table 1 shows the average values in Experiment 1. Our method was the best among all methods. These results suggest the usefulness of our algorithm for generating keywords.

Except for calendars, there were significant differences among the four methods ($p < 0.01$) by a one-way anova. There also were significant differences between (a) and (c), (a) and (d), and (b) and (d) (each $p < 0.01$).

From the comparison of (a) and (c), we found that the noun phrases extracted by our method are better than nouns; from the comparison of (b) and (c), our method is better than TermExtract for extracting noun phrases from our data; from the comparison of (c) and (d), adding adjectives and verbs is better than only using noun phrases. We found that noun phrases, adjectives, and verbs must be extracted as keywords rather than only nouns or noun phrases, and appearance is better than importance as the order of keywords.

Table 1. Result of Experiment 1

	(a)	(b)	(c)	(d) our method
Web searches	2.71	2.81	4.02	**4.05**
Twitter	2.56	2.43	3.56	**3.91**
E-Mails	2.85	3.26	4.10	**4.15**
Calendars	3.50	3.54	3.89	**4.00**
Book purchases	2.78	2.69	3.88	**4.18**

4.2 Experiment 2

We investigated the features of three visualization methods: a list of history structures, visualization algorithm 1, and visualization algorithm 2 when the users can recall memories of a day last week.

Method. The subjects were six male, computer and information science graduate students aged 22-24.

We gathered and generated history structures for each subject from five information usages: Web searches, twitter, e-mails, calendars, and book purchases. Three sets of history structures of three days from about one week before were assigned to three visualization methods.

We asked the subjects to use the history structures display (without showing other display parts) and explain what they recalled about the day in interviews. Next, we repeated the above processes using a knowledge-space display (visualization 1) and a knowledge-space display (visualization 2). After each experiment, we asked subjects to answer the pros and cons for each visualization method. Finally, the subjects ranked their answers for the three methods to these two questions: "Was this method useful to recall the past?" (Q1) and "Was this method useful to summarize the day?" (Q2).

Results and Discussion. For question 1, all subjects answered that using history structures was the most useful way to recall the past. Visualizations 1 and 2 were equal; three subjects ranked visualization 1 and three others ranked visualization 2 second. For question 2, three subjects answered that visualization 2 was the most useful, two answered visualization 1, and one answered the history structures. We think that a list of history structures is useful to recall the past and that knowledge space is useful to summarize one day. All subjects recommended displaying the history structures and the knowledge space together.

We combined user comments for the pros and cons that express the same meanings. For the list of history structures, "The keywords are ordered chronologically" (4: pros) and "Viewing was difficult" (2: cons). For visualization 1, "I can get a day's overview by looking at the clusters" (6: pros), "One cluster corresponds to a topic" (3: pros), and "I can grasp a day intuitively" (3: pros) and "Similar clusters need to be combined" (3: cons). For a visualization 2, "The information is well-organized, and there are many keywords that help recall" (4: pros), "Visualization 2 is better arranged than visualization 1" (2: pros), and "This method doesn't provide enough information" (2: cons).

We think that a list of history structure is beneficial for sequentially viewing chunks of memory, and a knowledge space is good to see grasp their overview.

4.3 Experiment 3

We evaluated the usefulness of our system to support human recollection when the users can recall the memories of a day of the previous week.

Method. The subjects were the same six male students from Experiment 2, which was conducted immediately before Experiment 3.

For Experiment 3, one set of the history structures of five information usages of one day gathered in Experiment 2 was used. The subjects used the knowledge-space browser and explained what they recalled about the day in interviews.

The subjects also evaluated the usefulness of the system by answering six questions on a five-point scale (5: I completely agree, 4: I agree, 3: OK, 2: I don't agree, 1: I completely disagree).

Table 2. Results of Experiment 3

	Question	Mean
Q1	Could you recall your past faster using the system?	4.8
Q2	Could you recall your past easier using the system?	5.0
Q3	Did the system help you recall something that you had forgotten?	4.7
Q4	Do you think that the system is useful?	4.2
Q5	Do you want to use the system again?	4.5
Q6	Do you think that you can recall your past easier using the system?	4.7

Results and Discussion. The evaluation results are shown in Table 2. The subjects thought the system was useful to support human recollection. In a knowledge-space browser, a user can choose the visualization algorithm. In this experiment, three subjects selected visualization 1 and three others selected visualization 2. Two of the latter subjects changed their choices from visualization 2 to 1 because they thought visualization 2 lacked sufficient information.

The overall results suggest that the combination of display of history structure and knowledge space is useful to support human recollection.

5 Related Work and Discussion

This research is a part of a system called *Memory-Organizer* that helps users construct "externalized memory" [5][6]. We previously proposed knowledge space to help users explore their knowledge spaces created from Web browsing history [7]. We extended this idea and proposed history structures to integrate various information sources [8]. The differences between the previous research and this paper are that we presented new algorithms for generating history structures for five information usages and two visualization algorithms for generating knowledge space. We also developed a new version of the knowledge-space browser and conducted experiments to determine what users can recall on a particular day. We found that the overall usefulness of our approach was good and that the system helped user recollection.

Much research has presented ideas for integrating such information in the light of Personal Information Management (PIM) [9], to overcome information overload [10][11]. History structure is simply generated from existing information sources. Our approach resembles tagging; however, the manual tagging of personal information is time-consuming. We aim to automatically generate history structure. Our research and PIM are related to a research topic called lifelogs that capture a person's entire lives. Mylifebits [12] is well-known not only as PIM but also as lifelog research. In typical lifelog research, lifeloggers wear computers to capture what they have seen and heard. Much research has been conducted in the context of multimedia or image processing. Our research resembles text-based lifelog research. Our approach is simple and does not need special devices to capture information from the real world. Although much research extracts information from each information source such as web searches, twitter, e-mails, and calendars, little research extracts and integrates information from those sources.

The following are our paper's main contributions. First, we presented a data integration method using a simple information structure called a history structure that is constructed from time, keywords, and URI sets. We also developed algorithms for generating history structures from various information sources and visualizing user knowledge spaces from them. Second, we developed a knowledge-space browser based on this approach and combined history structure and knowledge-space displays to help human recollection. Third, we evaluated whether our system helps users recall a particular day by summarizing its history structures. The experimental results revealed the usefulness of our approach and the implemented system.

Future work is listed below. First, although we analyzed the difference between visualization algorithms 1 and 2, we did not find out which is better in which user situation. Second, we need to improve our algorithms for generating keywords and visualization. Third, we need to examine the system for different periods, such as a day or a week within the past month.

6 Conclusions

We presented a data integration method using a simple information structure called history structure that is constructed from time, keywords, and URI sets and developed the following: (a) heuristic based keyword generation algorithms that extract noun phrases, adjectives, and verbs from various information sources; and (b) two visualization algorithms that create user knowledge spaces from history structures. We developed a system based on our approach to support human recollection and evaluated whether it can help users recall a particular day by summarizing that day's history structure. The experimental results revealed the usefulness of our approach and the implemented system.

References

1. MeCab: Yet Another Part-of-Speech and Morphological Analyzier, `http://mecab.sourceforge.net/`
2. Mitsuhashi, K.: A Human Recollection Support System by Integrating Diverse Information and Creating Knowledge Space, Master's Thesis, Graduate School for Creative Cities, Osaka City University (2011) (in Japanese)
3. `http://gensen.dl.itc.u-tokyo.ac.jp/termextract.html`
4. Collins, A.M., Loftus, E.F.: A Spreading-Activation Theory of Semantic Processing. Psychological Review 82(6), 407–428 (1975)
5. Murakami, H., Hirata, T.: Information Acquisition and Reorganization from the WWW by using Memory-Organizer. Bulletin of Osaka City University Media Center 3, 9–14 (2002)
6. Murakami, H., Hirata, T.: A System for Generating User's Chronological Interest Space from Web Browsing History. International Journal of Knowledge-Based and Intelligent Engineering Systems 8(3), 149–160 (2004)
7. Murakami, H., Hirata, T.: Recall of Web Browsing History by Generating User's Knowledge Space. Bulletin of Osaka City University Media Center 5, 21–26 (2004)
8. Murakami, H.: History Structure for Exploring Desktop Data. In: Proceedings of the SIGIR 2010 Workshop on Desktop Search (Understanding, Supporting and Evaluating Personal Data Search), pp. 25–26 (2010)
9. Jones, W.: Personal Information Management. ARIST 41, 453–504 (2007)
10. Dumais, S., Cutrell, D., Cadiz, J.J., Jancke, G., Sarin, R., Robins, D.C.: Stuff I've Seen: A System for Personal Information Retrieval and Re-Use. In: Proceedings of SIGIR 2003, pp. 72–79 (2003)
11. Cutrell, E., Dumais, S., Teevan, J.: Searching to Eliminate Personal Information Management. CACM 49(1), 58–64 (2006)
12. Gemmel, J., Bell, G., Lueder, R., Drucker, S.: Mylifebits: Fullfilling the Memex Vision. In: Proceedings of Tenth ACM Multimedia, pp. 235–238 (2002)

A Robust Approach to Digit Recognition in Noisy Environments

O. Matei[1], P.C. Pop[2], and H. Vălean[3]

[1] Dept. of Electrical Engineering, North University of Baia Mare,
Str. V. Babes, 430083, Baia Mare, Romania
oliviu.matei@holisun.com
[2] Dept. of Mathematics and Computer Science, North University of Baia Mare,
Str. Victoriei, 430122, Baia Mare, Romania
petrica.pop@ubm.ro
[3] Dept. of Automation, Technical University of Cluj-Napoca,
Str. G. Baritiu nr. 26-28, Cluj-Napoca, Romania
honoriu.valean@aut.utcluj.ro

Abstract. The article presents an original approach to optical character recognition (OCR) used in real environments, such as gas- and electricity-meters, where the quantity of noise is sometimes as large as the quantity of good signal. This approach uses two algorithms for better results. These are a neural network on one hand, respectively the k-nearest neighbor as the confirmation algorithm. Unlike other OCR systems, this one is based on the angles of the digits, rather than on pixels. This makes it insensitive to the possible rotations of the digits, respectively to the quantity of noise that may appear in an image. We will prove that the approach has several advantages, such as: insensitivity to the possible rotations of the digits, the possibility to work in different light and exposure conditions, the ability to deduct and use heuristics for character recognition.

Keywords: optical character recognition, gas-meter, electricity-meter, k-nearest neighbor, neural networks.

1 Introduction

Optical Character Recognition (OCR) is a method to locate and recognize text stored in an image, such as a jpeg or a gif image, and convert the text into a computer recognized form such as ASCII or unicode. OCR converts the pixel representation of a letter into its equivalent character representation. The benefits are straightforward. Many companies have large collections of paper forms and documents. Another application field is in automation.

Therefore a large amount of research has been carried along time for finding and improving OCR algorithms. Mori et al. present a very good overview of the existing OCR systems in [12] to which Blue et al. add a very pertinent and valuable evaluation in [4]. Regarding the digit recognition, LeCun et al. [10] give an overview of the existing algorithms. LeCun et al. [5] uses neural networks

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 606–615, 2012.

Fig. 1. Several images to be processed by the OCR system

for handwritten digit recognition. Rabiner et al. [13] use Markov models for the same purpose and Lee [11] use a combination of k-nearest-neighbor, radial-basis function, and neural networks. Deformable templates have been used by Jain et al. [9].

1.1 Requirements Specification

This article is based on our experience with a real-life application researched and developed for several European companies specialized in automation for energy providers.

Such companies need OCR systems which can work in harsh and real environments, such as gas-meters and electricity-meters. There are several problems with them: they are not uniformly illuminated along the day, they are not uniformly exposed within the same shot (some parts are in shade, while some are over-exposed), the quantity of noise is very high, there is no possibility to fully train a system for all types of gas-meters etc.

In such conditions, a robust system is highly needed, especially when the application is real-life and used by large companies with thousands of subscribers.

The size and zoom of the images may vary as well as the font and the color of the digits. Figure 1 shows several such images.

To summarize, the requirement is to recognize the digits from gas- and electricity-meters obtained from web cameras in real environments. If the digits are visible less than 50% of their height, the result will be discarded. The region for each digit is selected manually.

2 General Algorithm

The flow of the algorithm is depicted in figure 2.

The image is acquired using web cameras mounted in front of the gas- or electricity-meters and is fed as an input to the OCR system, which enhances it, processes it and returns the values of two digits: the last one before the decimal point, respectively the first one afterwards.

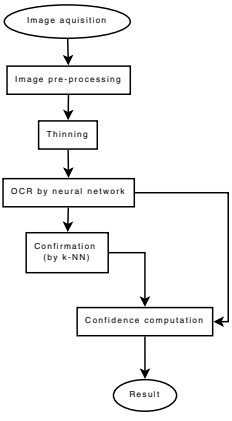


Fig. 2. The general process of character recognition

The process of digit recognition consists of several stages:

image pre-processing which converts the color image to a monochrome one, prepared for further analysis;

image thinning and segmenting for converting the digit (with a width) to a segmented line. The angles between subsequent segments will be the input for both the neural network as well as for the confirmation algorithm;

digit recognition using a neural network returns the most probable value of the digit with a confidence. We cannot rely solely on this results, therefore the next step is required;

confirmation of the result using k-nearest neighbor is the stage in which the value of the digit is computed with a completely different algorithm.

computation of the confidence: In most cases, the values of the two approached should coincide. However, in practice they are not always the same. Therefore a principle for accepting either result is required.

2.1 Image Pre-processing

The images display a wide range of colors. Therefore the first step is to convert them to a monochrome images (containing various intensities of gray). After the conversion, the general contrast is improved by histogram equalization (for more details see [14]). Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast without affecting the global contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. A related technique is the enhancement of the high frequency details (sharpening). And finally, an adaptive thresholding (see [15]) brings up the digit (along with some noise).

Further the image is sharpened. What it actually does, however, is to emphasize edges in the image and make them easier for the eye to pick out – while

Fig. 3. A quite common case when the noise cannot be set aside the digit itself

the visual effect is to make the image seem sharper, no new details are actually created.

The final pre-processing step is adaptive thresholding, used to segment an image by setting all pixels whose intensity values are above a threshold to a foreground value and all the remaining pixels to a background value.

After this stage, the image is ready for further specific processing.

2.2 Thinning and Segmenting

Quite often, due to the difference in light exposure, the noise is attached to the digit itself and cannot be removed, like in figure 3.

In this case the quantity of noise is almost equal to the quantity of the digit. This is the number of pixels with noise is almost equal to the number of digits representing the digit. Feeding any recognition system with such a distorted signal would fail priori. Therefore an approach to reduce the noise as much as possible was necessary, and this is thinning, which is the process of reducing the digit to a line and subsequently segmenting that line segments of equal size. The main characteristic of the thinning process is the preservation of the length.

We have come up with an algorithm based on the center of mass of each region of the digit. This assures a thinning in only one step, preserving the ratio between the length and the width of the image. The algorithm is as follows:

1. Choose a bordering pixel, preferably starting with the left-most and top-most one, but not necessarily.
2. Build the smallest square neighboring matrix (NM), containing the selected pixel and not overlapping and existing neighboring matrix. Each cornering pixel of the NM must have at most one neighboring pixel (on X and Y directions, not on diagonal) not belonging to the NM.
3. The mass center is computed for the NM. If the NM is 2x2 and fully white, any of the pixels may be the mass center.
4. Replace NM with the pixel in the mass center.
5. Repeat from step 1 until there are no more available pixels.
6. Connect the mass centers by lines, resulting a thinned image.

The results of the proposed thinning and segmenting stage is depicted in figure 5.

Afterwards, the angles of the digit are computed. Those are the angles between each two adjacent segments.

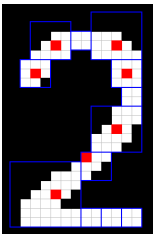


Fig. 4. The processing of thinning a digit. The blue lines represent the NM's. The red pixels are the mass centers.

Fig. 5. The source and the result of thinning and segmenting process for two digits: 2 and 3

Given the consecutive segments u and v, the angle made by them is computed as the angle between the vectors $\overrightarrow{u}$ and $\overrightarrow{v}$:

$$\widehat{u,v} = \widehat{\overrightarrow{u},\overrightarrow{v}} \tag{1}$$

The angle between two 2D vectors is:

$$\widehat{\overrightarrow{u},\overrightarrow{v}} = arccos\left(\frac{u \cdot v}{\|u\|\|v\|}\right) \tag{2}$$

where $\|u\|$ is the norm of the vector $\overrightarrow{u}$, $\cdot$ is the cross product and *arccos*, the inverse or *cosine* function.

The first segment starts always from the top-most extreme point. That is the top-most point with only one neighbor. If there are n segments, then there will be $n-1$ angles. The last segment may be or may not be adjacent to the first one. In other words, the first and the last point may coincide, like in the case of 0 and 8.

The algorithm has been developed in the following manner:

1. Choose the first point as the top-most extreme point.
2. Look at the adjacent point in the clockwise direction on the curve at a distance of k pixels. (In our project, $k = 5$).
3. If the line has already been passed, look further clockwise (in other words "search on another line"). If there are other points, go to step 2.
4. If there are no neighboring points, then stop.
5. repeat from step 2.

Depending on the digit, the number of segments may vary significantly. The shortest digits are 1 and 7, the longest one is 8. However, there is no limit imposed to the number of segments.

2.3 OCR by Neural Network

An artificial neural network involves a network of simple processing elements (artificial neurons) which can exhibit complex global behavior, determined by the connections between the processing elements and element parameters. Artificial neurons were first proposed in 1943 by Warren McCulloch, a neurophysiologist, and Walter Pitts, an MIT logician.

The input of the neural network consists of the angles between subsequent segments as computed in equation 2. This approach has several advantages over the algorithms using the value of each pixel: the input is significantly smaller, the weight of the noise is reduced and the rotation of the digits does not affect the neural network architecture and its inputs.

The neural network is a multi-layer perceptron (MLP) with only one hidden layer. The input layer consists of 35 rows, which is the highest number of angles. The hidden layer consists of 15 units. The output layer consists of 10 units, one for each digit. The transfer function for both layers is hyperbolic tangent. The output layer, with 10 elements, uses $softMax$ function, defined as:

$$f(x_i, w_i) = \frac{e^{x_i^{lin}}}{\sum_j x_i^{lin}} \tag{3}$$

where $x_i^{lin} = \beta x_i$ is scaled and offset activity inherited from the linear function $f(x_i, w_i) = \beta x_i + w_i$.

The learning rule is Fahlman's quickprop algorithm (see [7]), which is a gradient search procedure that has been shown to be very fast in a multitude of problems. It basically uses information about the second order derivative of the performance surface to accelerate the search. The number of epochs was set to 500. Over this number, the neural network becomes overtrained, which is it classifies the digits in the training set very well, but performs bad on new inputs. The training set consists of the segmented images from 2 gas- and electricity-meters taken with web cameras in real conditions.

The tests have been carried out on the images from other 3 gas- and electricity-meters. The confusion matrix based on the classification of the neural network is shown in the table 1.

The table 1 states that 0 was classified correctly in 90% of the trials and as 8 in 10% of the cases. The least correctly classified digit was 3, which was mistaken for 0 in 20% of the cases, for 4 in 5% and for 8 in 10% of the trials. Digit 6 was classified correctly for all the experiments. This table gives a clear idea about the similarities between various digits, such as 0, 3,8 and 9 on one hand, respectively 1 and 7 on the other hand. These results concorde with the ones presented by Abdezaleem in [1].

Table 1. The confusion matrix after the neural network classification

	0	1	2	3	4	5	6	7	8	9
0	0.9	0	0	0	0	0	0	0	0.1	0
1	0	0.85	0	0	0	0	0	0.1	0	0.05
2	0.1	0	0.85	0	0	0	0	0.05	0	0
3	0.2	0	0	0.65	0.05	0	0	0	0.1	0
4	0	0.1	0	0	0.75	0	0.15	0	0	0
5	0	0	0	0	0.05	0.95	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0
7	0.1	0	0	0	0	0	0	0.9	0	0
8	0	0	0	0	0	0	0	0	0.8	0.2
9	0.1	0	0	0	0	0	0	0	0.1	0.8

2.4 Confirmation by k-NN

For confirmation of the result of the neural network, the k-nearest neighbor (k-NN) algorithm has been used.

The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors.

The neighbors are taken from a set of objects for which the correct classification (or, in the case of regression, the value of the property) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. In order to identify neighbors, the objects are represented by position vectors in a multidimensional feature space. It is usual to use the Euclidean distance. However, as the objects are made of angles, we simply used the distance between angles, that is the absolute value of their difference. The k-nearest neighbor algorithm is sensitive to the local structure of the data.

An object (digit) is a vector consisting of the angles computed as in section 2.2 along with their number:

$$o_k = (n_k, \widehat{a_1^k}, \widehat{a_2^k}, ..., \widehat{a_{n_k}^k}) \tag{4}$$

The distance between two objects:

$$o_i = (n_i, \widehat{a_1^i}, \widehat{a_2^i}, ..., \widehat{a_{n_i}^i}) \tag{5}$$

$$o_i = (n_j, \widehat{a_1^j}, \widehat{a_2^j}, ..., \widehat{a_{n_j}^j}) \tag{6}$$

is

$$d(i,j) = max(|n_i - n_j|, |\widehat{a_1^i} - \widehat{a_1^j}|, |\widehat{a_2^i} - \widehat{a_2^j}|, ..., |\widehat{a_{n_i}^i} - \widehat{a_{n_j}^j}|) \tag{7}$$

where $|a|$ is the absolute value of a.

If two objects are of different length, the shorter one is filled with null values until the lengths of the two objects are equal.

Table 2. The confusion matrix after the confirmation step

	0	1	2	3	4	5	6	7	8	9
0	1	0	0	0	0	0	0	0	0	0
1	0	0.99	0	0	0	0	0	0.01	0	0.05
2	0	0	1	0	0	0	0	0	0	0
3	0.02	0	0	0.98	0	0	0	0	0	0
4	0	0.01	0	0	0.99	0	0.15	0	0	0
5	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	0	0.98	0.02
9	0	0	0	0	0	0	0	0	0.01	0.99

After the confirmation step using the kNN, the confusion matrix looks like in table 2.

From table 2 we can see that digit 1 was still mistaken for 7 and 9, digit 4 for 1 and 6, and digit 8 for 9 and viceversa. However, the percentages of misclassified trials is significantly low that after the first step, using the neural network (see section 2.3).

2.5 Confidence Computation

Both algorithms return a result with some confidence, which is different than 100% in most cases. When the confidence factors are high or when the results of the 2 approaches are the same, the result is straightforward and requires no further discussions.

The problems occur when the confidence factors are low and the results differ. In this case a decision tree is used based mostly on heuristics, such as:

- the length of a digit:
  short digits are those digits whose length is less than 20 segments. These are 1 and 7.
  medium-sized digits are the ones consisting of more than 20 segments, but less than 35. These are 2, 3, 4, 5.
  long digits are the ones longer than 35 segments, such as: 0, 6, 8 and 9.
- the position of the last segment relatively to the first one
  adjacent segments for digits 0 and 8;
  non-adjacent segments for all the other digits.
- characteristics of the curve (open of closed, intersecting or free etc.)
  open curve is characteristic to the digits 1, 2, 3, 5, 7.
  completely closed curve is the curve for each the first and the last pixel coincide and it is specific to 0 and 8.
  partially closed curve is specific to 6 and 8. Digits 4 may have or not a closed curve, depending on the font.

The first important observation is that the total "length" of a digit gives a significant clue about its value. The "length" of a digit is the number of segments which cover it entirely.

Such a classification is important for several reasons: it brings another glim to digits analysis and therefore: can be the starting point of an OCR based on decision trees, and can also be the starting point of an OCR knowledge-based.

Even if the results are similar to others, reported by Impedovo et al. [8], Bay [3] and Alpadyn [2], the proposed algorithm is very important for its applications in real life, harsh environments.

The robustness of the algorithm is given by the fact that each recognition is done by two different algorithms - a neural network approach and a k-NN algorithm. If the recognized character is the same in both cases, that the confidence of the result is high. Othewise, some heuristics are used within the step `confidence computation`. The former two algorithms are very mathematical and objective, whereas the latter one is rather subjective. It applies a completely different approach to make a clear distinction between the characters returned by the previous algorithms if they have different results. The heuristics used refer to features of digits, such as length, shapes, curves etc.

3 Conclusions

The algorithm was tested on real-life pictures highly distorted from gas- and electricity-meters. No noise was added manually or for the sake of experiments. A complete training set was not available since the range of tools used for various energy measurements is very wide, with no standards regarding the fonts, colors and sizes.

Another remark is that any recognition system needs a confirmation stage. In this case, two complementary algorithms have been used: a neural network and the kNN algorithm.

We proposed a novel approach to OCR based on the angles of the digit, rather than on the values of the pixels in the image and therefore it is insensitive to the digit rotation. Although it has been applied specifically to digit recognition, it can be extended for any character recognition. Its main strength is the robustness to noise, quite common in real-life applications, where the distortion can be up to 50%.

In conclusion, the approach has several advantages, such as: insensitivity to the possible rotations of the digits, the possibility to work in different light and exposure conditions, the ability to deduct and use heuristics for character recognition. These make it very good for real life applications, such as OCR of the gas- and electricity-meters.

Further, combining pixel-based approaches with angle-based method proposed in this article may lead to improved results.

Acknowledgment. This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS - UEFISCDI, project number PN-II-RU-TE-2011-3-0113.

References

1. Abdelazeem, S.: Comparing arabic and latin handwritten digits recognition problems. World Academy of Science, Engineering and Technology 54 (2009)
2. Alpaydin, E.: Combined 5 x 2 cv f test for comparing supervised classification learning algorithms. Neural Computation (11) (1999)
3. Bay, S.D.: Nearest neighbor classification from multiple feature subsets. Intelligent Data Analysis (3) (1999)
4. Blue, J.L., Candela, G.T., Grother, P.J., Chellappa, R., Wilson, C.L.: Evaluation of pattern classifiers for fingerprint and ocr applications. Pattern Recognition 27(4) (2004)
5. Le Cun, Y., Boser, B., Denker, J.S., Howard, R.E., Habbard, W., Jackel, L.D., Henderson, D.: Handwritten digit recognition with a back-propagation network. pp. 396–404 (1990)
6. Gu, X., Yub, D., Zhanga, L.: Image thinning using pulse coupled neural network. Pattern Recognition Letters 25(9) (2004)
7. Hoehfeld, M., Fahlman, S.E.: Learning with limited numerical precision using the cascade-correlation algorithm. IEEE Transactions on Neural Networks 3(4), 602–611 (1995)
8. Impedovo, S., Ottaviano, L., Occhinegro, S.: Optical character recognition–a survey. Internationl Journal of Pattern Recognition 5(1-2) (May 1991)
9. Jain, A.K., Zongker, D.: Representation and recognition of handwritten digits using deformable templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(12) (1997)
10. LeCun, Y., Jackel, L., Bottou, L., Brunot, A., Cortes, C., Denker, J., Drucker, H., Guyon, I., Muller, U., Säckinger, E., Simard, P., Vapnik, V.: Comparison of learning algorithms for handwritten digit recognition. In: International Conference on Artificial Neural Networks, pp. 53–60 (2004)
11. Lee, Y.: Handwritten digit recognition using k nearest-neighbor, radial-basis function, and backpropagation neural networks. Neural Computation 3(3), 440–449 (1991)
12. Mori, S., Suen, C.Y., Yamamoto, K.: Historical review of ocr research and development. Pattern Recognition 80(7) (2002)
13. Rabiner, L.R., Wilpon, J.G., Soong, F.K.: High performance connected digit recognition using hidden markov models. IEEE Transactions on Acoustics, Speech and Signal Processing 37(8) (1999)
14. Russ, A.: The Image Processing Handbook. CRC (2002)
15. Shapiro, L.G., Stockman, G.C.: Computer Vision. Prentice Hall (2002)

A One-Phase Method for Mining High Utility Mobile Sequential Patterns in Mobile Commerce Environments

Bai-En Shie, Ji-Hong Cheng, Kun-Ta Chuang, and Vincent S. Tseng

Department of Computer Science and Information Engineering,
National Cheng Kung University, Taiwan, ROC
brianshie@gmail.com, jihong@live.com,
{ktchuang,tsengsm}@mail.ncku.edu.tw

Abstract. Mobile sequential pattern mining is an emerging topic in data mining fields with wide applications, such as planning mobile commerce environments and managing online shopping websites. However, an important factor, i.e., actual utilities (i.e., profit here) of items, is not considered and thus some valuable patterns cannot be found. Therefore, previous researches [8, 9] addressed the problem of mining *high utility mobile sequential patterns* (abbreviated as *UMSPs*). Nevertheless the tree-based algorithms may not perform efficiently since *mobile transaction sequences* are often too complex to form compress tree structures. A novel algorithm, namely *UM-Span* (*high Utility Mobile Sequential Pattern mining*), is proposed for efficiently mining UMSPs in this work. UM-Span finds UMSPs by a projected database based framework. It does not need additional database scans to find actual UMSPs, which is the bottleneck of utility mining. Experimental results show that UM-Span outperforms the state-of-the-art UMSP mining algorithms under various conditions.

Keywords: Mobile sequential pattern, mobility pattern mining, utility mining, mobile commerce environment.

1 Introduction

Nowadays mobile devices and wireless applications become indispensable parts of our life. Analyzing user behavior patterns in mobile commerce environments becomes an important and emerging topic in the data mining field [5, 6, 8, 9, 11]. Extended from traditional sequential pattern mining [1, 7], Yun et al. [11] first proposed a framework which combines users' moving paths and purchased services to find *mobile sequential patterns* in mobile transaction sequences. For example, a mobile sequential pattern <(*A*{*clothes*})(*B*)(*C*{*lipsticks*}> means that *the customers often moved through the path <ABC> and bought clothes and lipsticks in locations A and C, respectively*. Mobile sequential pattern mining has many applications, such as planning mobile commerce environments and managing online shopping websites.

Nevertheless, the profits of items are not considered in mobile sequential pattern mining. Utility mining [2, 3, 4, 10] is proposed to address this problem in traditional transactional data mining framework. Pushing utility mining into mobile sequential pattern mining is crucial in many domains, such as mobile commerce environments,

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 616–626, 2012.

metropolitan planning and online shopping websites which sell a wide selection of merchandise in different web pages. In the point of view, Shie et al. addressed the problem of finding *high utility mobile sequential patterns* in mobile commerce environments by level-wise [9] and tree-based algorithms [8]. Although tree-based algorithm $UMSP_{BFG}$ [8] is the currently best algorithm, it still has the following drawbacks. First, data compression of MTS-Tree may not be good for sequence data. Since the combination of locations and items in mobile transaction sequences are complicated and mostly long, they may not be merged into nodes well. Even more, to evaluate the badly compressed tree structures is a hard task. Second, $UMSP_{BFG}$ needs additional scans of databases to get real utility of patterns, which is the bottleneck of utility mining [2, 3, 4, 10]. By the above discussions, we can realize that the tree structure is not the best solution for mining high utility mobile sequential patterns.

In view of this, we propose a novel method named *UM-Span* (*high Utility Mobile Sequential Pattern mining*). Major contributions of this work are described as follows. First, we propose two efficient strategies in UM-Span to catch actual utilities for high utility mobile sequential patterns. Thus UM-Span does not need additional scans of databases to check actual utilities of patterns. Second, UM-Span generates high utility mobile sequential patterns with an efficient projected database based method. Complete information of sequences can be recorded in projected databases. Thus the strategies can be applied and the performance can be effectively improved. The experimental results show that UM-Span outperforms the state-of-the-art mobile sequential pattern mining algorithms [8, 11] in different conditions.

The organization of this paper is as follows. Problem definitions are defined in Section 2. In Section 3, the proposed algorithm UM-Span is described in detail. Experimental evaluation is shown in Section 4. Conclusions are given in Section 5.

2 Preliminary and Problem Definitions

Let $L = \{l_1, l_2, \ldots, l_p\}$ be a set of *locations* in the mobile commerce environment and $I = \{i_1, i_2, \ldots, i_g\}$ be a set of *items* sold in the locations. An *itemset* is denoted as $\{i_1, i_2, \ldots, i_k\}$, where each item $i_v \in I$, $1 \le v \le k$ and $1 \le k \le g$. Given a *mobile transaction sequence database D*, a *mobile transaction sequence* $S = \langle T_1\ T_2\ \ldots\ T_n \rangle$ is a set of transactions ordered by time. A *transaction* T_j is a triple consisting of a location, a set of items and a set of quantities, denoted as $(l_j, \{i_{1^{(j)}}, i_{2^{(j)}}, \ldots, i_{h^{(j)}}\}, \{q_{1^{(j)}}, q_{2^{(j)}}, \ldots, q_{h^{(j)}}\})$. It represents that a user made T_j in l_j, where $1 \le j \le n$. There are h items with their quantity values in T_j. The elements in the set of items and the set of quantities are one-to-one, that is, the *purchased quantity* of $i_{p^{(j)}}$ is $q_{p^{(j)}}$ in T_j, where $1 \le p \le h$. For the sets that have only one element, the brackets could be omitted. Similarly, for the transactions without purchased item, we only record the locations that users passed by. A *path* is denoted as $\langle l_1\ l_2 \ldots\ l_r \rangle$, where $l_j \subseteq L$ and $1 \le j \le r$. For example, in Table 1, the third transaction in S_2, i.e., (C, $\{i_2, i_3\}$, {2, 5}), stands for the user bought two i_2 and five i_3 in location C. The path of S_6 is <CDEF>. We use a utility table to keep the unit utility for each item. Table 2 is an example utility table. For each item i_j in a utility table, $w(i_j)$ is *unit utility* of i_j.

Definition 1. (Loc-item) A *loc-item*, denoted as $l_{loc}\{i_j\}$, stands for the item i_j is purchased in the location l_{loc}, where $l_{loc} \in L$ and $i_j \in I$. The *utility* of $l_{loc}\{i_j\}$ in a mobile transaction sequence S_j is defined as $u(l_{loc}\{i_j\}, S_j) = q_j \times w(i_j)$.

Definition 2. (Loc-itemset) A *loc-itemset*, denoted as $(l_{loc}\{i_{1^{(j)}}, i_{2^{(j)}}, ..., i_{g^{(j)}}\})$, stands for the itemset $\{i_{1^{(j)}}, i_{2^{(j)}}, ..., i_{g^{(j)}}\}$ is purchased in l_{loc}, where $l_{loc} \in L$ and $\{i_{1^{(j)}}, i_{2^{(j)}}, ..., i_{g^{(j)}}\} \in I$. The utility of a loc-itemset $Y = (l_{loc}\{i_{1^{(j)}}, i_{2^{(j)}}, ..., i_{g^{(j)}}\})$ in S_j is defined as $u(Y, S_j) = \sum_{k=1}^{g} u(l_{loc}\{i_{k^{(j)}}\}, S_j)$. Moreover, the utility of Y in a mobile transaction sequence database D is defined as $u(Y) = \sum_{(Y \subseteq S_j) \wedge (S_j \in D)} u(Y, S_j)$.

Definition 3. (Moving pattern) A *moving pattern* P is a sequence of loc-itemsets, recorded as $P = < (l_m\{i_{1^{(m)}}, i_{2^{(m)}}, ..., i_{g^{(m)}}\})(l_n\{i_{1^{(n)}}, i_{2^{(n)}}, ..., i_{g^{(n)}}\})...(l_r\{i_{1^{(r)}}, i_{2^{(r)}}, ..., i_{g^{(r)}}\}) >$. The utility of P in S_j is defined as $u(P, S_j) = \sum_{\forall Y \in X} u(Y, S_j)$. The utility of P in D is defined as $u(P) = \sum_{(P \subseteq S_j) \wedge (S_j \in D)} u(P, S_j)$.

Definition 4. (Support) The *support* of an element e, such as a location, item, loc-item, loc-itemset and moving pattern, is denoted as $sup(e)$ and defined as the number of mobile transaction sequences that contain e in D.

Take Table 1 and Table 2 for example, $u(A\{i_1\}, S_1) = 2 \times 1 = 2$. The utility of the loc-itemset $C\{i_2, i_3\}$ in S_2 is calculated as $u(C\{i_2, i_3\}, S_2) = u(C\{i_2\}, S_2) + u(C\{i_3\}, S_2) = 2 \times 5 + 5 \times 3 = 25$. The utility of $C\{i_2, i_3\}$ in the database *DB* is calculated as $u(C\{i_2, i_3\}) = u(C\{i_2, i_3\}, S_1) + u(C\{i_2, i_3\}, S_4) = 25+20 = 45$. The utility of the loc-pattern $P_1 = <(A\{i_1\})(B)(C\{i_2, i_3\})>$ in S_2 is calculated as $u(P_1, S_2) = u(A\{i_1\}, S_2) + u(C\{i_2, i_3\}, S_2) = 3 + 25 = 28$. The utility of P_1 in *DB* is calculated as $u(P_1) = u(P_1, S_2) + u(P_1, S_3) = 28 + 23 = 51$. The support of P_1 is 2.

Definition 5. (Mobile sequential pattern) Given a user-specified *minimum support threshold* δ, a moving pattern P is called a *mobile sequential pattern* if $\sup(P) \geq \delta$. Length of a pattern is the number of loc-itemsets in this pattern. A pattern with length k is denoted as k-pattern.

In [11], mobile sequential patterns have two implicit properties as follows.

Property 1. (Path connectivity) A mobile transaction sequence S is said to *contain* a mobile sequential pattern P_{MSP} iff the path of P_{MSP} is an *uninterrupted* subsequence of the path of S.

Table 1. Mobile transaction sequence database *DB*

SID	Mobile transaction sequence	SU
S_1	< (A, i_1, 2) (B) (C, i_2, 1) (D, i_4, 1) (E) (F, i_5, 2) >	54
S_2	< (A, i_1, 3) (B) (C, {i_2, i_3}, {2, 5}) (K) (E, i_6, 10) (F, i_5, 4) (G, i_8, 2) (L) (H, i_7, 2) >	132
S_3	< (A, i_1, 3) (B) (C, {i_2, i_3}, {1, 5}) (D, i_4, 2) (E) (F, {i_5, i_6}, {1, 2}) (G) (H, i_7, 1) >	72
S_4	< (A, i_1, 1) (W) (C, i_3, 10) (E) (F, i_5, 1) (G, i_8, 2) (L) (H, i_7, 1) (E, i_9, 1) >	59
S_5	< (A, i_1, 4) (B) (C, i_3, 10) (D, i_4, 1) (E) (F, i_5, 1) (G) (H, i_7, 2) >	73
S_6	< (C, i_2, 2) (D) (E, i_9, 1) (F, i_5, 1) >	31

Table 2. Utility table

Item	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
Utility	1	5	3	11	18	2	5	1	3

Property 1 avoids generating the mobile sequential patterns with invalid routes since the road map is not used in this work. For example, assume there are two mobile sequential patterns P_2 = <A{i_1}BC{i_2}> and P_3 = <A{i_1}C{i_2}>. P_2 is contained in S_1 of Table 1, but P_3 is not. P_3 could not be a mobile sequential pattern since there may be no path between A and C directly.

Property 2. (Path validity). A mobile sequential pattern P_{MSP} is *valid* if the set of items in the first and the last loc-itemsets of P_{MSP} are not empty.

Definition 6. (High utility mobile sequential pattern, UMSP) Given a user-specified minimum utility threshold ε, a mobile sequential pattern P_{MSP} is called a *high utility mobile sequential pattern*, abbreviated as *UMSP*, if $u(P_{\text{MSP}}) \geq \varepsilon$. In other words, a moving pattern P is a UMSP if $u(P) \geq \varepsilon$ and $\sup(P) \geq \delta$.

For example, if δ = 2 and ε = 50, the moving pattern <(A{i_1})(B)(C{i_2, i_3})>, whose support is 2 and utility is 51, is a 3-high utility mobile sequential pattern (3-UMSP). Generally speaking, if ε is set to 0, this problem equals to the problem of finding traditional mobile sequential patterns.

Downward closure property [1] of a pattern P is that if P satisfies some constraints, the sub-patterns of P must also satisfy the constraints. By the property, the search space can be effectively reduced. However utility patterns have no downward closure property [2, 3, 4, 10]. A naïve method for finding UMSP is to discover MSP [11] first and then find UMSP within the set of MSP. Nevertheless this method is not efficient. Thus we introduce the *sequence weighted downward closure* (abbreviated as SWDC) property in mobile transaction sequence databases, which are proposed in [8, 9]. By SWDC, downward closure can be applied in the process of mining UMSP.

Definition 7. (Sequence utility) *Sequence utility* of mobile transaction sequence S_j, which is the sum of utilities of all loc-items in S_j, is defined as $SU(S_j) = \sum_{l_{loc}\{i_j\} \subseteq S_j} u(l_{loc}\{i_j\}, S_j)$.

Definition 8. (Sequence weighted utilization) *Sequence weighted utilization*, abbreviated as SWU, of a moving pattern P is defined as $SWU(P) = \sum_{(P \subseteq S_j) \wedge (S_j \in D)} SU(S_j)$.

Definition 9. (High sequence weighted utilization mobile sequential pattern, WUMSP) A moving pattern P is called a *high sequence weighted utilization mobile sequential pattern* if $sup(P) \geq \delta$ and $SWU(P) \geq \varepsilon$.

For example in Table 1, the sequence utility of the mobile transaction sequence S_6 is computed as $SU(S_6) = u(C\{i_2\},S_6) + u(E\{i_9\},S_6) + u(F\{i_5\},S_6)$ = 10+3+18 = 31. SWU of the moving pattern P_1 = <(A{i_1})(B)(C{i_2, i_3})> is computed as $SWU(P_1) = SU(S_2) + SU(S_3)$ = 132+72 = 204. If δ = 2 and ε = 50, P_1 is a 3-WUMSP.

Property 3. (Sequence weighted downward closure, SWDC). For any pattern P, if P is not a WUMSP, any superset of P is not a WUMSP.

***Proof*:** By the definition about SWU, for a moving pattern P, $SWU(P)$ is larger than or equal to $SWU(P')$, where P' is a superset of P. If $SWU(P)$ is less than ε, $SWU(P')$ is also less than ε. Similarly, by the definition about support, $sup(P)$ is larger than or equal to $sup(P')$. If $sup(P)$ is less than δ, $sup(P')$ is also less than δ. By the above two conditions, if $SWU(P)$ is less than ε or $sup(P)$ is less than δ, $SWU(P')$ or $sup(P')$ must be less than ε or δ, respectively. ■

Property 4. Assume the complete set of UMSPs and WUMSPs are called SET_{UMSP} and SET_{WUMSP}, respectively. $SET_{UMSP} \subseteq SET_{WUMSP}$.

***Proof*:** For a moving pattern P in a sequence S_j, $u(P, S_j) \leq SU(S_j)$. Thus, $u(P) = \sum_{(P \subseteq S_j) \wedge (S_j \in D)} u(P, S_j) \leq \sum_{(P \subseteq S_j) \wedge (S_j \in D)} SU(S_j) = SWU(P)$. Assume P is a UMSP, we get $SWU(P) \geq u(P) \geq \varepsilon$, P must also be a WUMSP. Therefore, the complete set of UMSPs is contained in the set of WUMSPs, in other words, $SET_{UMSP} \subseteq SET_{WUMSP}$. ■

Problem Statement. Given a mobile transaction sequence database D, a pre-defined utility table, a user-specified minimum support threshold δ and a user-specified minimum utility threshold ε, the problem of mining high utility mobile sequential patterns from D is to discover the complete set of mobile sequential patterns that satisfy δ and ε.

3 Proposed Method: UM-Span

Algorithm UM-Span consists of four main steps as shown in Fig.1. In Step 1, the original database is scanned once to find 1-WUMSPs. Then in Step 2, projected databases are built with the *utility shrinking strategy*. At the same time, the loc-items that are impossible to be parts of UMSPs are removed from the database. Subsequently, the projected databases are scanned to find local high utility loc-items in Step 3. Finally in Step 4, the *utility accumulation strategy* is applied to find UMSPs. In the following paragraphs, we use an example to describe the process of UM-Span in details. Take the mobile transaction sequence database *DB* and the utility table in Table 1 and Table 2 for example. Assume the minimum support threshold δ is 2 and the minimum utility threshold ε is 100.

In Step 1 of UM-Span, the database *DB* is scanned once. SU of each mobile transaction sequence, as shown in the last column of Table 1, as well as 1-WUMSPs, their support, SWU and utility, as shown in Table 3, are accumulated. At the same time, we also find global frequent locations as follows: {A, B, C, D, E, F, G, H, L}. In Table 3, there is only one UMSP: F{i_5}.

In Step 2, the projected databases of 1-WUMSPs are built in the same way as [7]. Moreover, the loc-items and locations that are not WUMSPs and frequent locations are removed. If the path of a sequence is broken since some loc-items or locations are removed, the loc-items and locations are called *cut-off point*. Only the part of a sequence from the prefix to a cut-off point is kept by Property 1. Take <C{i_2}> in Table 1 for example, <C{i_2}>'s projected database (abbreviated as <C{i_2}>-PDB) is built as follows. For S_1, C{i_2} is in the third transaction, we record the remaining sequence <(D,i_4,1)(E)(F,i_5,2)> into <C{i_2}>-PDB. C{i_2} is also in the third transaction of S_2. When we take away C{i_2} and its previous transactions from S_2, there is still an item {i_3} bought in C. Thus we record (_,{i_3}) for noticing that the location of {i_3} is the same as the last loc-itemset of its prefix. In S_2, since K is not a frequent location, K is

removed from S_2 and form a cut-off point in S_2. Thus the second sequence recorded in <C{i_2}>-PDB is only <(_, i_3, 5)>. When S_3 is dealt with, loc-item F{i_6} should be removed since it is not a 1-WUMSP; However, since F is a frequent location, we only remove {i_6} and its purchased quantity. Thus < (_, i_3, 5) (D, i_4, 2) (E) (F, i_5, 1) (G) (H, i_7, 1) > is recorded as the third sequence in <C{i_2}>-PDB. S_4 and S_5 do not contain C{i_2}. In S_6, item {i_9} is removed because E{i_9} is not a 1-WUMSP but E is a frequent location. <(D)(E)(F,i_5,1)> is recorded as the last sequence in <C{i_2}>-PDB. Projected databases of A{i_1} and C{i_2} are recorded in Fig. 2.

```
Algorithm: UM-Span
Input: A database DB, a minimum support threshold and a minimum utility
    threshold
Output: UMSPs

/* STEP 1: Scan database */
1. Scan DB once. Get the set of 1-WUMSPs WUMSP, the set of frequent
   locations FL and 1-UMSPs. Output 1-UMSPs.
2. For each 1-WUMSP P_1-WUMSP in WUMSP, call GEN_PROJDB(P_1-WUMSP, DB, WUMSP, FL)

/* STEP 2: Generate projected database */
GEN_PROJDB(a prefix P_prefix, a database DB', a list of WUMSPs WUMSP, a list
of frequent locations FL)
1. For each sequence S in DB'
2.   SU(S, P_prefix-PDB)=SU(S, DB'), PV(P_prefix, S)=PV(DB', S).
3.   For each loc-item LI in S
4.     If LI is not in WUMSP or FL
5.       Prune LI. Subtract u(LI) from SU(S, P_prefix-PDB).
6.       If LI is a cut-off point
7.         Prune the remaining loc-items in S.
8.         Subtract utilities of the pruned loc-items from
   SU(S, P_prefix-PDB).
9.     Else if LI is P_prefix
10.      Set a flag to start recording the following loc-items into
         P_prefix-PDB.
11.      PV(P_prefix, S)+=u(LI).
12.    Else if the flag is set
13.      Record LI to S in P_prefix-PDB.
14. Call SCAN_PROJDB(P_prefix-PDB)

/* STEP 3: Scan projected database */
SCAN_PROJDB(a database DB')
1. Scan DB' once. Get the set of local 1-WUMSPs WUMSP_L, the set of local
   frequent locations FL_L and the set of following elements SET_E.
   /* STEP 4: Generate UMSP */
   Calculate actual utility for each local loc-item by Strategy 2.
2. For each local 1-WUMSPs LI_WUMSP
3.   New WUMSP P_WUMSP= P_prefix+LI_WUMSP.
4.   If u(P_WUMSP)≧ , output P_WUMSP as a new UMSP.
5. For each following elements E in SET_E
6.   New prefix P_New= P_prefix+E, Call GEN_PROJDB(P_New, DB', WUMSP_L, FL_L)
```

Fig. 1. The framework of the proposed algorithm UM-Span

Table 3. Length 1 WUMSPs

Loc-item	A{i_1}	C{i_2}	C{i_3}	D{i_4}	F{i_5}	G{i_8}	H{i_7}
Support	5	4	4	3	6	2	4
SWU	390	289	336	199	421	191	336
Utility	13	30	90	44	180	4	30

For calculating the actual utilities of moving patterns, we record a *prefix value*, abbreviated as *PV*, in the sequences of the projected databases. The definition of PV is described as follows.

Definition 10. (Prefix value, PV) Prefix value of a prefix P_{pre} in a sequence *S'* in P_{pre}*-PDB*, denoted as $PV(P_{pre}, S')$, is the utility of P_{pre} in the sequence *S* where *S'* is derived from *S*.

For example, $PV(A\{i_1\},S_1) = 2\times1 = 2$ and $PV(D\{i_4\},S_3) = 2\times11 = 22$.

SID	<A{i_1}>-PDB	PV	SU
S_1	< (B) (C, i_2, 1) (D, i_4, 1) (E) (F, i_5, 2) >	2	54
S_2	< (B) (C, {i_2, i_3}, {2, 5}) >	3	28
S_3	< (B) (C, {i_2, i_3}, {1, 5}) (D, i_4, 2) (E) (F, i_5, 1) (G) (H, i_7, 1) >	3	68
S_5	< (B) (C, i_3, 10) (D, i_4, 1) (E) (F, i_5, 1) (G) (H, i_7, 2) >	4	73

(a) <A{i_1}>-PDB

SID	<C{i_2}>-PDB	PV	SU
S_1	< (D, i_4, 1) (E) (F, i_5, 2) >	5	52
S_2	< (_, i_3, 5) >	10	25
S_3	< (_, i_3, 5) (D, i_4, 2) (E) (F, i_5, 1) (G) (H, i_7, 1) >	5	64
S_6	< (D) (E) (F, i_5, 1) >	10	28

(b) <C{i_2}>-PDB

Fig. 2. Projected databases of 1-WUMSP A{i_1} and C{i_2}

Table 4. Local frequent loc-items in <A{i_1}>-PDB

Loc-item	C{i_2}	C{i_3}	D{i_4}	F{i_5}	H{i_7}
Support	3	3	3	3	2
SWU	150	169	195	195	141
Utility	28	70	53	81	22

Moreover, since the loc-items that are not 1-WUMSPs are pruned in the projected databases, SUs for the sequences in the projected databases should be recalculated to get the new SUs after pruning the loc-items. Thus we propose a *utility shrinking strategy* which can make the overestimated utilities, i.e., SWU, be effectively shrunk. The utility shrinking strategy can just record the sequence utility after pruning irrelevant items in each sequence. The strategy is described as follows.

Strategy 1. (Utility shrinking) Assume that a sequence *S'* in a projected database P_{pre}*-PDB* is derived from a sequence *S* by deleting the set of irrelevant loc-items SET_{ir}. Sequence utility of *S'* in P_{pre}*-PDB* is recalculated as Eq. 1.

$$SU(S', P_{pre}\text{-}PDB) = SU(S) - \sum_{\forall x \in SET_{ir}} u(x) \cdot \tag{1}$$

Take S_2 in <A{i_1}>-PDB for example, since the loc-items after K in the original sequence, i.e., S_2 in *DB*, are all pruned, $SU(S_2$, <A{i_1}>-PDB) is calculated as $132 - (10\times2 + 4\times18 + 2\times1 + 2\times5) = 28$. It can be observed that this value is just the exact utility of sequence <(A, i_1, 3) (B) (C, {i_2, i_3}, {2, 5})>, that is the possible maximal pattern derived from S_2 in <A{i_1}>-PDB.

In Step 3, projected databases are scanned sequentially and local 1-WUMSPs are generated from the projected databases recursively. In this example, <A{i_1}>-PDB is scanned once and local frequent loc-items, as shown in Table 4, are acquired. For keeping path connectivity, a list for the loc-items or locations that *exactly follow* the prefix in each sequence is kept during the first scan of a projected database. We call the

loc-items or locations *following elements*. For example, B is a following element in <A{i_1}>-PDB since only B exactly follows <A{i_1}> in the sequences. Thus we can only generate one moving pattern <A{i_1}B> which satisfies property 1 in <A{i_1}>-PDB. However it is not a valid mobile sequential pattern by property 2, thus it is not a WUMSP. Next, <A{i_1}B>-PDB, as shown in Fig. 3 (a), is generated. Then <A{i_1}B>-PDB is scanned once and the local frequent loc-items are acquired (In this example, they are just the same as those in Table 4). The following elements in <A{i_1}B>-PDB are C, C{i_2} and C{i_3}, thus two valid WUMSPs, i.e, <A{i_1}BC{i_2}> and <A{i_1}BC{i_3}>, are generated.

SID	<A{i_1}B>-PDB	PV	SU
S_1	< (C, i_2, 1) (D, i_4, 1) (E) (F, i_5, 2) >	2	54
S_2	< (C, {i_2, i_3}, {2, 5}) >	3	28
S_3	< (C, {i_2, i_3}, {1, 5}) (D, i_4, 2) (E) (F, i_5, 1) (G) (H, i_7, 1) >	3	68
S_5	< (C, i_3, 10) (D, i_4, 1) (E) (F, i_5, 1) (G) (H, i_7, 2) >	4	73

(a) <A{i_1}B>-PDB

SID	<A{i_1}BC{i_2}D{i_4}E>-PDB	PV	SU
S_1	< (F, i_5, 2) >	18	54
S_3	< (F, i_5, 1) (G) (H, i_7, 1) >	30	53

(b) <A{i_1}BC{i_2}D{i_4}E>-PDB

Fig. 3. Projected databases of <A{i_1}B> and <A{i_1}BC{i_2}D{i_4}E>

In Step 4, actual utilities of WUMSPs are calculated. To get actual utilities of WUMSPs, previous researches [8, 9] have developed two-phase based methods for mining UMSP. However, the methods need to perform additional scans of databases to find UMSPs from WUMSPs in the second phase, which is the bottleneck of utility mining. To conquer this problem, we develop a *utility accumulation strategy* for effectively calculating actual utilities of WUMSPs in UM-Span.

Strategy 2. (Utility accumulation) Actual utility of a moving pattern P_{MP}, which is the combination of a prefix P_{pre} and a local loc-item LI_{local}, in the projected database P_{pre}*-PDB* is calculated as Eq. 2.

$$u(P_{MP}) = \sum_{\forall S_j \supseteq LI_{local} \wedge S_j \in P_{pre}-PDB} PV(P_{pre}, S_j) + u(LI_{local}, S_j) \quad (2)$$

By Strategy 2, actual utilities of WUMSPs can be calculated during the first time scan of projected databases without additional scans of databases. For example, actual utilities of <A{i_1}BC{i_2}> and <A{i_1}BC{i_3}> are calculated as u(<A{i_1}BC{i_2}>) = PV(A{i_1}B, S_1) + u(C{i_2}, S_1) + PV(A{i_1}B, S_2) + u(C{i_2}, S_2) + PV(A{i_1}B, S_3) + u(C{i_2}, S_3) = 2+5+3+10+3+5 = 28 and u(<A{i_1}BC{i_3}>) = PV(A{i_1}B, S_2) + u(C{i_3}, S_2) + PV(A{i_1}B, S_3) + u(C{i_3}, S_3) + PV(A{i_1}B, S_5) + u(C{i_3}, S_5) = 3+15+3+15+4+30 = 70. The utilities of the two WUMSPs are less than ε, thus no UMSP are generated.

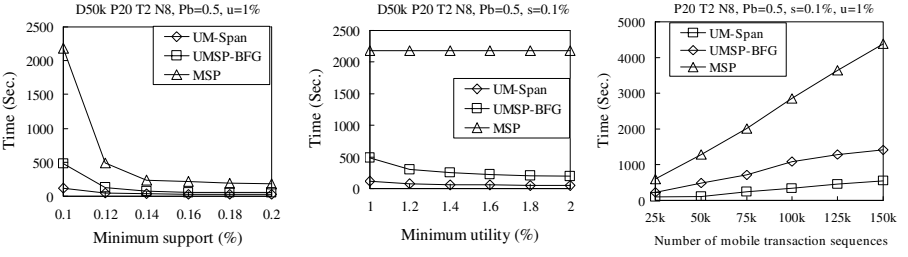
The process is recursively performed. When <A{i_1}BC{i_2}D{i_4}E>-PDB (as shown in Fig. 3 (b)) is dealt with, a following element F{i_5} is obtained and its SWU is 107. Thus a WUMSP <A{i_1}BC{i_2}D{i_4}EF{i_5}> are generated. Its actual utility is calculated as u(<A{i_1}BC{i_2}D{i_4}EF{i_5}>) = PV(<A{i_1}BC{i_2}D{i_4}E>, S_1) + u(F{i_5}, S_1) + PV(<A{i_1}BC{i_2}D{i_4}E>, S_3) + u(F{i_5}, S_3) = 18+36+30+18 = 102. Thus a UMSP <A{i_1}BC{i_2}D{i_4}EF{i_5}> is identified. In this way, all UMSPs are acquired by recursively mining projected databases.

4 Experimental Evaluations

In this section, we evaluate the performance of the proposed algorithms. The experiments were performed on a 2.8 GHz Processor with 3 gigabyte memory, and the operating system is Microsoft Windows 7 64bit. The algorithms are implemented in Java language. The settings of parameters related to mobile commerce environment and utility mining are similar to [11] and [4, 10], respectively. Default settings and the descriptions of parameters are listed in Table 5.

Table 5. Parameter settings

Parameter Descriptions	Default
D: Number of mobile transaction sequences	50k
P: Average length of mobile transaction sequences	20
T: Average number of items per transaction	2
N: Size of mesh network	8
$\mathbf{n_I}$: The range of the number of items sold in each location	200
$\mathbf{P_b}$: The probability that user makes the transaction in the location	0.5
w: Unit profit of each item	1~1000
q: Number of purchased items in transactions	1~5



(a) Varied min_sup (b) Varied min_util (c) Varied DB size

Fig. 4. Experimental results

The first part of the experiments is the performance under various minimum support thresholds. Minimum utility threshold is set to 1%. The results are shown in Fig. 4 (a). With the decrease of minimum supports, the runtime of compared methods increases since the number of UMSPs increases. In this experiment, UM-Span outperforms $UMSP_{BFG}$ [8] and MSP [11] due to the following reasons. First, the SUs for the sequences are decreased during the mining process of UM-Span by Strategy 1, thus fewer WUMSPs are needed to be checked. Second, by Strategy 2, UM-Span does not need to spend time for additional scans of databases to get UMSPs. The two effective strategies make UM-Span more efficient than other methods.

The second part of the experiments is the performance under various minimum utility thresholds. Minimum support threshold is set to 0.1%. The results are shown in Fig. 4 (b). It can be seen that UM-Span also outperforms the two compared algorithms. Besides, since MSP does not consider utility in the mining process, its execution time remains the same.

The last part of the experiments is the scalability of the compared algorithms. The results are shown in Fig. 4 (c). We can see that when the number of mobile transaction sequences increase, the execution time of the compared algorithms increase linearly. Thus the compared algorithms have good scalability on runtime. Moreover, UM-Span is still the best performer in this experiment.

By the above experiments, the proposed algorithms are shown to outperform the state-of-the-art algorithms $UMSP_{BFG}$ and MSP. In addition, we do not show the results about memory usage of the compared methods due to the page limit. The memory usage of UM-Span is less than the two compared algorithms since it needs to check fewer candidates and then generates fewer sub data structures than $UMSP_{BFG}$.

5 Conclusions

In this research, we proposed a novel algorithm, named UM-Span, for mining high utility mobile sequential patterns in mobile commerce environments. Main advantages of UM-Span are listed as follows. First, UM-Span applies a projected database based framework. By the first proposed strategy, estimated utilities and the number of candidate patterns are effectively reduced from projected databases. Second, by the second proposed strategy, UM-Span does not need additional scans of database to check the actual utilities of patterns. The experimental results show that UM-Span outperforms the state-of-the-art high utility mobile sequential pattern mining algorithms in different conditions. For future work, we will consider some constraints in this scenario and develop efficient and effective algorithms for finding the interesting patterns that fit to users' demand.

Acknowledgements. This research was supported by National Science Council, Taiwan, R.O.C. under grants no. NSC 100-2218-E-006-039 and NSC 100-2631-H-006-002.

References

1. Agrawal, R., Srikant, R.: Mining Sequential Patterns. In: Proc. of 11th Int'l Conf. on Data Mining, pp. 3–14 (1995)
2. Ahmed, C.F., Tanbeer, S.K., Jeong, B.-S., Lee, Y.-K.: Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases. IEEE Transaction on Knowledge and Data Engineering 21(12), 1708–1721 (2009)
3. Li, Y.-C., Yeh, J.-S., Chang, C.-C.: Isolated items discarding strategy for discovering high utility itemsets. Data & Knowledge Engineering 64(1), 198–217 (2008)
4. Liu, Y., Liao, W.-K., Choudhary, A.: A fast high utility itemsets mining algorithm. In: Proc. of Utility-Based Data Mining (2005)
5. Lu, E.H.-C., Tseng, V.S.: Mining cluster-based mobile sequential patterns in location-based service environments. In: Proc. of IEEE MDM, pp. 273–278 (2009)
6. Lu, E.H.-C., Tseng, V.S., Yu, P.S.: Mining Cluster-Based Temporal Mobile Sequential Patterns in Location-Based Service Environments. IEEE Transactions on Knowledge and Data Engineering 23(6), 914–927 (2011)

7. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C.: Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. IEEE Transactions on Knowledge and Data Engineering 16(10) (2004)
8. Shie, B.-E., Hsiao, H.-F., Tseng, V.S., Yu, P.S.: Mining High Utility Mobile Sequential Patterns in Mobile Commerce Environments. In: Yu, J.X., Kim, M.H., Unland, R. (eds.) DASFAA 2011, Part I. LNCS, vol. 6587, pp. 224–238. Springer, Heidelberg (2011)
9. Shie, B.-E., Hsiao, H.-F., Yu, P.S., Tseng, V.S.: Discovering Valuable User Behavior Patterns in Mobile Commerce Environments. In: Cao, L., Huang, J.Z., Bailey, J., Koh, Y.S., Luo, J. (eds.) PAKDD Workshops 2011. LNCS, vol. 7104, pp. 77–88. Springer, Heidelberg (2012)
10. Tseng, V.S., Wu, C.-W., Shie, B.-E., Yu, P.S.: UP-Growth: An Efficient Algorithm for High Utility Itemsets Mining. In: Proc. of the 16th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD 2010), pp. 253–262 (2010)
11. Yun, C.-H., Chen, M.-S.: Mining Mobile Sequential Patterns in a Mobile Commerce Environment. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews 37(2) (2007)

The Use of Shapley Value to Power Allocation Games in Cognitive Radio Networks

Jerzy Martyna

Jagiellonian University, Institute of Computer Science
Faculty of Mathematics and Computer Science
ul. Prof. S. Lojasiewicza 6, 30-348 Cracow, Poland

Abstract. In this paper, the Shapley value method is introduced to power allocation in cognitive radio (CR) networks. This concept is intended to maximize the information throughput sustained by each link over the network. In the traditional approach, game theory is used as a tool to model the interaction between several players and predict the outcome of the power allocation game. Our solution to this problem determines the participation of the players in the formed coalition of secondary users (SUs) in cognitive radio networks by using the Shapley value. The simulation results show the effectiveness of the proposed power control policy.

Keywords: game theory, cognitive radio networks, Shapley value.

1 Introduction

Cognitive radio (CR) networks are highly agile wireless platforms capable of autonomously choosing device parameters based on current conditions [1, 8, 11, 18]. For instance, during any natural disaster in wireless communication, centralized wireless network may not be available due to the overloaded and/or damaged access points or base stations. In the CR networks wireless access can be established in areas of the accessible wireless infrastructure.

We use Fig. 1 to illustrate the CR network, where multiple primary users (PUs) or wireless providers compete for a shared pool of secondary users. All the secondary users (SUs) are equipped with cognitive radio technologies. They are usually static or mobile. The primary users are the infrastructure of wireless operators and then are responsible to throw unused frequency to the secondary users for a monetary payoff.

Cognitive radio networks provide the capacity to share a wireless channel with the licensed users in an opportunistic way. The CR networks are envisioned to be able to provide a high bandwidth to mobile users via heterogeneous wireless architectures and a dynamic spectrum access (DSA) technology. The users of the CR networks can either communicate with each other in a multi-hop manner or access the base station. Therefore, we may categorise the CR application of the spectrum into three possible scenarios, (a) a CR network on a licensed band, (b) a CR network on an unlicensed band, (c) a CR network on both a licensed band and an unlicensed band.

Applying game theory to address various problems in CR networks is currently a hot area [10]. Most games considered in wireless communication systems to data have been

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 627–636, 2012.

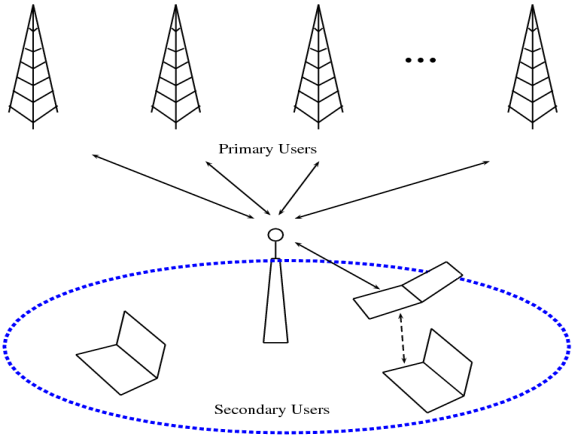


Fig. 1. Spectrum structure of the cognitive radio

applied to address the resource allocation problems in CR systems [5, 12]. Furthermore, multiuser power control problems in these systems are formulated under game theoretic framework and various of iterative water-filling algorithms are used to obtain the Nash equilibrium [13, 17]. However, to the best of our knowledge, none of the existing algorithms to calculate the Nash equilibrium have been addressed the hierarchical games, in which radios interact to maximize their respective payoff following a leader-follower approach.

In this paper, we examine the problem of power allocation (PA) in the CR networks on a licensed and unlicensed band with the main emphasis on the concept of hierarchy of the existing between radios. This problem arises in the following situations: (a) when the primary and secondary systems share the spectrum, (b) when users have access to the medium in an asynchronous manner, (c) when operators deploy their networks at different times, (d) when some nodes have more power than others, such as the base station. One of the most popular model of the hierarchical spectrum of sharing is the Stackelberg equilibrium (SE) [6], [16]. This approach was motivated by the fact that the noncooperative Nash equilibrium (NE) is generally inefficient and nonoptimal. The Stackelberg equilibrium provides better outcomes as compared to the noncooperative approach. However, the mathematical framework of the Stackelberg equilibrium is not suitable for practical use. Therefore, we propose a new scheme of the power allocation (PA) problem in the CR networks which is based on Shapley's value vector.

The main goal of this paper is to show that the Shapley values as one of the game's solutions of power allocation problem in the CR networks. Following the Shapley value model, we will discuss the issues of maximizing the effective throughput of the secondary users (SUs) in a licensed band subject to the constraints of the transmission power and average interference power.

The rest of this paper is outlined as follows. Section 2 concerns the application of the Shapley value of games to power allocation for the SUs in the CR networks. In section 3, we introduce the optimal power policy to maximize the effective capacity of the SUs subject to the constraints of the transmission power and the average interference power.

The simulation results are illustrated in section 4. Concluding remarks are drawn in section 5.

2 Application of the Shapley Value of Games to Power Allocation in Cognitive Radio Networks

In this section we investigate applications of the Shapley value of games to power allocation in the CR network.

Our approach is based on two important multiuser channel models, namely: the multiple access channel (MAC) [3] and the interference channel (IFC).

In the first one, we assume use of the uplink channel in a single-cell multi-carrier cellular system in which each multiple access channel (MAC) consists of K transmitters aiming to communicate with a single receiver using a common channel. There exist N independent or parallel MACs. None of the transmitters in different MACs interferes with each other. The channel gain from transmitter i to the receiver over channel n is denoted by h_i^n. Let the channel realizations during the transmission of M consecutive symbols be constant. All the channel realizations i, $i \in \{1, \ldots, k\}$ and $n \in \{1, \ldots, N\}$ are drawn from a Gaussian distribution with a zero mean and a unit variance. Thus, the power allocated by transmitter i to channel n is denoted by p_i^n. We can formulate the following condition for the transmitter i, namely

$$\sum_{n=1}^{N} p_i^n \leq P_i^{max} \ \ \forall i = \{1, \ldots, K\} \tag{1}$$

We assume that the noise at the receiver is described by w_i^n. It corresponds to the additive white Gaussian noise (AWGN) process with a zero mean and variance σ^2. The received signal can be written as

$$y^n = \sum_{i=1}^{K} h_i^n x_i^n + w_i^n \quad \forall n = \{1, \ldots, N\} \tag{2}$$

where x_i^n and h_i^n are the transmitted symbols and the channel realization of the transmitted symbols, respectively.

Assuming a single-user decoding (SUD) on channel n for transmitter i, the received signal to interference plus the noise ratio (SINR) is expressed as

$$SINR_i^n = \frac{p_i^n \mid h_i^n \mid^2}{\sum_{j \neq i}^{K} p_j^n \mid h_j^n \mid^2 + \sigma^2} \quad \forall i = \{1, \ldots, K\}, \ \ \forall n = \{1, \ldots, N\} \tag{3}$$

The interference channel model (IFC) described by T. S. Chan *et al.* [7] and by Etkin *et al.* [4] consists of a set of K point-to-point links sufficient to produce mutual interference due to their co-existence on the same channel. Assuming that $N \geq 1$ channels are available, in the IFC model N independent or parallel channels exist, where transmitters in different IFCs do not interfere with each other. In essence, the IFC model corresponds to the transmission in pairs between nodes over a set of sub-carriers. We assume that

the channel realization from transmitter i to receiver j on channel n is denoted by b_i^n, where $n = \{1, \ldots, N\}$, and $(i, j) \in \{1, \ldots, K\}^2$. Thus, the received signal at receiver i is given by

$$r_i^n = \sum_{j=1}^{K} h_{ji}^n x_i^n + w_i^n \tag{4}$$

where $\mathbf{w}_i = \{w_i^1, \ldots, w_i^n\}$ is the noise at receiver i over channel n.

Assuming a single-user decoding (SUD) on channel n for transmitter i, the received SINR can be expressed as

$$SINR_i^n = \frac{p_i^n \mid h_{ii}^n \mid^2}{\sum_{j \neq i}^{K} p_j^n \mid h_{ji}^n \mid^2 + \sigma^2} \;\; \forall i = \{1, \ldots, K\}, \;\; \forall n = \{1, \ldots, N\} \tag{5}$$

The difference between Eqs. (3) and (5) is that each transmitter knows the channel realization h_i^n for all $\forall i = \{1, \ldots, K\}$ in the MAC model and h_{ij}^n for all $(i, j) \in \{1, \ldots, K\}^2$ in the IFC model.

In the PA game, the set of players includes transmitters, base stations, and mobile stations. In general, a game is presented in a normal form as follows:

Definition 1 (Normal form). *A game in a normal form is given by* $\{\mathcal{K}, \mathcal{S}, \{u_k\}_{k \in \mathcal{K}}\}$ *and is composed of three elements:*

- *a set of players:* $\mathcal{K} = \{1, \ldots, K\}$,
- *a set of strategy profiles:* $\mathcal{S} = S_1 \times \ldots \times S_k$, *where* S_k *is the strategy set of player* k,
- *a set of utility functions: the* k*-th player's utility function is* $u_k : \; S_k \rightarrow \mathcal{R}_+$ *and is denoted by* $u_k(s_k, s_{-k})$ *where* $s_k \in S_k$ *and* $s_{-k} = (s_1, \ldots, s_{k-1}, s_{k+1}, \ldots, s_k) \in S_1 \times \ldots \times S_{k-1} \times S_{k+1} \times \ldots S_k$.

Considering that players are willing to cooperate to achieve a fair allocation of resources, we impose a condition that the utility function must account for both the interference perceived by the current players, and the interference that particular player is causing to neighboring players sharing the same channel.

The utility function is defined as follows:

Definition 2 (Utility function). *The utility function is given by*

$$u_k(s_k, s_{-k}) = - \sum_{j \neq k, j=1}^{N} p_j(s_j) G_{kj} f(s_j, s_k)$$
$$- \sum_{j \neq k, j=1}^{N} p_k G_{jk} f(s_k, s_j) \;\; \forall k = 1, \ldots N \tag{6}$$

where G_{kj} *is the link gain between transmitter* j *and receiver* k, $f(s_k, s_j)$ *is an interference function given by*

$$f(s_k, s_j) = \begin{cases} 1, \text{ if } s_j = s_k, \text{ transmitter } j \text{ and } k \text{ choose} \\ \quad \text{the same strategy (same channel)} \\ 0, \text{ otherwise.} \end{cases} \tag{7}$$

The above utility function accounts for both the interference measured at the current user's receiver and the interference created by the user to others.

As a solution of the coalition for the game a method introduced by L. S. Shapley is used [14], [15]. The main idea of Shapley's method lies in the defintion of player usefulness for the coalition and rewards assignment which is proportional to their potential contributions.

We introduce the Shapley value:

Definition 3 (The Shapley value of a game in a normal form). *Let v be a game given by $\{\mathcal{K}, S, \{u_k\}_{\forall k}\}$. The Shapley value of v, $\Phi(v) = (\phi_1(v), \ldots, \phi_k(v)) \in \mathcal{K}$ is defined by*

$$\phi_i(v) = (K-1)!\frac{(a-1)!}{K!}\sigma(C,i) \tag{8}$$

for each player i, $1 \leq i \leq K$ attached to coalition C couting $(a-1)$ players as the a-th player, and $\sigma(C,i)$ is the usefulness of player i for the coalition C and is given by

$$\sigma(C,i) = \mu(C) - \mu(C\backslash\{i\}) \tag{9}$$

where $\mu(C\backslash\{i\})$ is the reward for coalition C without the i-th player and $\mu(C)$ is the reward for coalition C.

Each coalition can be assigned the usefulness function of all players for the formed coalition. We assume that the usefulness function of a dummy player is equal to 0.

A formal definition of the influence of the outgoing player for the coalition is given as follows:

Definition 4 (Influence of the player's going out into the coalition). *For the sake of the best possible coalition we have observed that the reward of coalition C changes its value from 1 to 0 after player i is going out of the coalition.*

According to the Shapley method [14], [15] we can univocally assign to each game the imputation which is reasonable partitioning of winnings. The following definition gives the terms of the player's participation in the coalition.

Definition 5 (Participation in the coalition). *The participation of the player in the coalition is determined by the values of the Shapley vector.*

For a simple example, consider the three-player game with coalition.

Table 1. Player usefulnesses for all possible coalitions

{X}	{Y}	{Z}	{X, Y}	{X, Z}	{Y, Z}	{X, Y, Z}
80	0	80	120	240	80	400

Example 1. *Three players are building a coaltion to utilize an existing system. A participation of each player in the possible coalition is different. Player usefulness for all possible coalitions are given in Table 1.*

Table 2. Participations of all players in different coalitions

	X	Y	Z
{X, Y, Z}	80	40	280
{X, Z, Y}	80	160	160
{Y, Z, X}	320	0	80
{Y, X, X}	120	0	280
{Z, X, Y}	160	160	80
{Z, Y, X}	320	0	80
Together:	1080	360	960

The participations of players in formed coalitions are as follows:

$$\sigma(\{X\}) = 80,\ \ \sigma(\{Y\}) = 0,\ \ \sigma(\{Z\}) = 80$$
$$\sigma(\{X, Y\}, X) = 120 - 0 = 120$$
$$\sigma(\{X, Y\}, Y) = 120 - 80 = 40$$
$$\sigma(\{X, X\}, X) = 240 - 80 = 160$$
$$\sigma(\{X, Z\}, Z) = 240 - 80 = 160$$
$$\sigma(\{Y, Z\}, Y) = 80 - 80 = 0$$
$$\sigma(\{Y, Z\}, Z) = 80 - 0 = 80$$
$$\sigma(\{X, Y, Z\}, X) = 400 - 80 = 320$$
$$\sigma(\{X, Y, Z\}, Y) = 400 - 240 = 160$$
$$\sigma(\{X, Y, Z\}, Z) = 400 - 120 = 280$$

The participation of all players in different coalitions are given in Table 2. Thus, the average participation of each player is given by $\frac{1}{6}(1080, 360, 960)$. *It is just the Shapley value of this game.*

3 The Spectrum Sharing for Cooperative Secondary Systems with the Use of the Shapley Value

In this section, we investigate the dynamic spectrum sharing in the CR network in which primary systems lease the spectrum to secondary system in exchange for cooperation in the PA game.

We assume that a primary transmitter wishes to send information to its primary receiver either directly with a rate R_{dir} or by means of the cooperation from a subset $S \subseteq S_{tot}$ of $\mid S \mid = k \leq \mid S_{tot} \mid = k$ secondary nodes/transmitters. The primary system can divide its data into two parts $(1-\alpha)/bit$ durations, and αL bit durations with $0 \leq \alpha \leq 1$. The first part is dedicated to a direction transmission from primary transmitter to the primary receiver whereas the second αL bit duration is again divided into two parts. One part, consisting of $\beta\alpha L$, with $0 \leq \beta \leq 1$, is dedicated to sending information from the primary transmitter to the primary receiver using the secondary nodes by means of the distributed space time coding [9]. The remaining $\alpha(1-\beta)L$ bits are devoted to the secondary network for the sake of its own data transmission. The problem of power allocation in the secondary system of the CR network can be solved by

maximization of its utility function while deciding about the portion of time-slots α, β and $S \subseteq S_{tot}$ subset of secondary transmitters.

Given the set S and cooperation parameters α, β, the PA optimization problem is given by

$$\max_{\alpha,\beta,S} \left(\sum_{i \in S, i=1}^{C} u_i(s_i, s_{-i}) \right) \tag{10}$$

subject to $S \subseteq S_{tot}, 0 \leq \alpha, \beta \leq 1$.

The secondary system maximizes its utility function of the formed coalition C by means of maximization of the achievable trasmission rate along with taking into consideration the cost of the transmitted energy E_c. The optimization problem for the secondary system can be expressed as

$$\max_{s_i} \left(\sum_{i=1, i \in S}^{C} u_i(s_i, s_{-i}) \right) = \max_{s_i} \{ \alpha(1-\beta) \log_2 \left(1 + \frac{\mid h_{S,ii} \mid^2 s_i}{\sigma^2 + \sum_{j=1, j \neq i}^{k} \mid h_{S,ji} \mid^2 s_j} \right) - E_c s_i \} \tag{11}$$

subject to $0 \leq s_i \leq S_{i,max}$.

Solving Eq. (11) and assuming that the strategy profile $s_i = p_i$, we can obtain the value of power for transmitter i, namely

$$p_i = \max \left(0, \frac{1-\beta}{E_c} - \frac{\sigma^2}{\mid h_{S,ii} \mid^2} - \sum_{j=1, j \neq i} \frac{\mid h_{S,ji} \mid^2}{\mid h_{S,ii} \mid^2} p_j \right) \tag{12}$$

In our approach the interaction between the primary and secondary users is modelled as cooperative game. The coalition maximizes its own utility function. Using the Shapley value we obtain the participation of each player in the game with the maximal utility function of the coalition.

$$\hat{p}_i = arg \max_i \left(u_i(p_1, \ldots, p_C) \right) = f \cdot \Phi_i(v) \cdot p_i \quad 0 \leq p_i \leq P_i^{max} \tag{13}$$

where $\Phi_i(v)$ is the Shapley value for transmitter i and f is a normalizing parameter.

procedure *power_allocation*;
compute_the_Shapley_value_for_coalition;
repeat
  for $i := 1$ **to** C **do**
  compute p_i *from* $Eq.(12)$;
  find $\hat{p}_i = f \cdot \Phi_i \cdot p_i$;
  endfor;
until *coalition_is_empty*;

Fig. 2. An algorithm for power allocation in CR network

We present an algorithm that first finds a coalition among the transmitters, and further, by means of using the Shapley value calculates the participation of each of them (see Fig. 2). Additionally, our algorithm maximizes the throughput and minimizes the average interference power.

4 Simulation Results

In this section, we present the simulation results in order to study the performance of our scheme compared with the Stackelberg equilibria in the same scenario.

We have the following general settings for the simulation. We place the primary transmiiter BS at coordinates and five secondary users which are uniformely located in the area 100 m $\times$ 100 m. The maximum power for a secondary user is $P_{max} = 100$ mW. The rest of the parameters are set as follows: antenna gain 5, threshold power $3 \cdot 10^{-7}$ W. The AWGN at all receivers has the same power $W = 5 \times 10^{-7}$ mW and the interference power threshold at all receivers is -50 dB.

By means of using the Nash and the Stackelberg equilibria [2] we find the optimal value of p_i^1 while keeping $p_1^2, \ldots, p_1^N$ fixed and then we find the optimal p_1^2 keeping the other p_1^n $(n \neq 2)$ fixed and so on. Such a process guaranteeds to convergence because each iteration increases the objective function.

We then evaluate how the Shapley vector of the formed coalition affects the power allocation in the CR networks with a varied number of secondary users. Fig. 3 depicts the achievable rate for both types of users versus the signal-to-noise ratio for the NE,

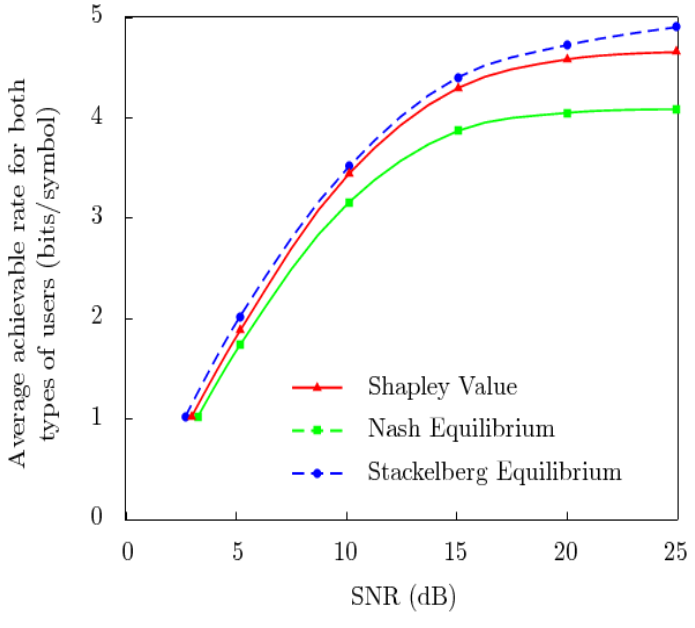


Fig. 3. Average achievable rate for both types of users versus the signal-to-noise (SNR) ratio for the Shapley value, Stackelberg and Nash equilibria approaches

SE and Shapley value (SV) approaches. As can be seen, the average achievable rate of Shapley's vector is comparable to the SE approach.

Figure 4 shows the cumulative distribution function (CDF) of the ratio of the cooperative and noncooperative approach. In this scenario, we assume that $K = 3$ operators with one primary operator and three secondary users sharing the same spectrum. It is composed of $N = 5$ carriers. The entrance of each player to the coalition took place according to their index values. In order to achieve the CDF in a noncooperative approach we propose a repeated game in which the players will be added to the coalition in a strictly defined succession. This succession must guarantee the highest values of Shapley's vector.

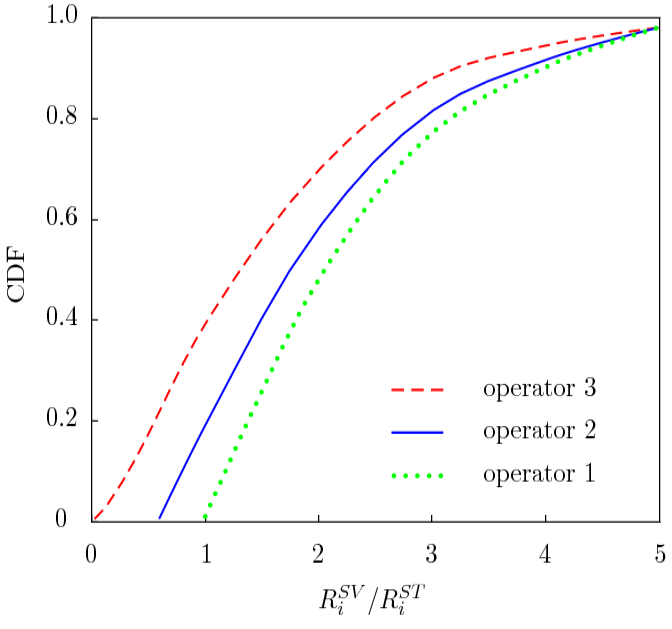


Fig. 4. Cumulative distribution function (CDF) versus the ratio of the rates determined by the Shapley and Stackelberg value (noncooperative game) approaches

5 Conclusion

In this paper, we proposed a new approach to power allocation in the cognitive radio networks. Our approach was based on the Shapley value of games. We also proposed a use of the algorithm which allows us to the maximization of the throughput and minimization of the average interference. From our simulation experiments, we concluded that our model leads to accurate results when the secondary users can form a game. The future work could also consider more extensive simulation of our method.

References

1. Akyildiz, I.F., Lee, W.-Y., Vuran, M.C., Mohanty, S.: Next Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: a Survey. Computer Networks (2006)
2. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
3. Cover, T.M., El Gamai, A., Salehi, M.: Multiple Access Channels with Arbitrarily Correlated Sources. IEEE Trans. on Information Theory 26(6), 648–657 (1980)
4. Etkin, R., Tse, D., Wang, H.: Gaussian Interference Channel Capacity to Within One Bit. IEEE Trans. Information Theory 54(12), 5534–5562 (2008)
5. Felegyhazi, M., Cagalj, M., Bidokhri, S.S., Hubaux, J.: Non-cooperative Multi-radio Channel Allocation in Wireless Networks. In: Proc. of IEEE INFOCOM, pp. 1442–1450 (2007)
6. Fudenberg, D., Tirole, J.: Game Theory. The MIT Press, Cambridge (1991)
7. Han, T.S., Kobayashi, K.: A New Achievable Rate Region for the Interference Channel. IEEE Trans. on Information Theory 27(1), 49–60 (1981)
8. Haykin, S.: Cognitive Radio: Brain-Empowered Wireless Communications. IEEE Journal on Selected Areas in Communications 23(2), 201–220 (2005)
9. Laneman, J.N., Wornell, G.W.: Distributed Space-Time Coded Protocols for Exploiting Cooperative Diversity in Wireless Networks. IEEE Trans. on Information Theory 49(10), 2415–2425 (2003)
10. Lasaulee, S., Debbah, M., Altman, E.: Methologies for Analyzing Equilibria in Wireless Games. IEEE Signal Processing Magazine. Special Issue on Game Theory for Signal Processing (September 2009)
11. Mitola, J.: Software Radios: Wireless Architecture for the 21st Century. John Wiley and Sons, Inc. (2000)
12. Nie, N., Comaniciu, C.: Adaptive Channel Allocation Spectrum Etiquette for Cognitive Radio Networks. In: Proc. of IEEE DySPAN, Baltimore, MD, pp. 209–278 (2005)
13. Scutari, G., Palomar, D., Barbarossa, S.: Asynchronous Iterative Water-filling for Gaussian Frequency-selective Interference Channels. IEEE Trans. on Information Theory 54(7), 2868–2872 (2008)
14. Shapley, L.S.: Rand Corporation Research Memorandum. Notes on the N-Person Game: Some Variants of the von Neumann-Morgenstern Definitionof Solution, RM-812 (1952)
15. Shapley, L.S.: A Value for N-Person Games. In: Kuhn, H.W., Tucker, A.W. (eds.) Contribution to the Theory of Games. Princeton University Press, Princeton (1953)
16. Stackelberg, V.H.: Marketform und Gleichgewicht. Oxford University Press, Oxford (1934)
17. Yu, W., Gints, G., Cioffi, J.M.: Distributed Multiuser Power Control for Digital Subscriber Lines. IEEE Journal on Selected Areas in Comm. 20(5), 1105–1115 (2002)
18. Zhang, Y., Zheng, J., Chen, H.-H.: Cognitive Radio Networks. Architectures, Protocols and Standards. CRC Press, Boca Raton (2010)

Mining Time-Gap Sequential Patterns

Show-Jane Yen and Yue-Shi Lee

Department of Computer Science & Information Engineering, Ming Chuan University,
5 De Ming Rd., Gwei Shan District, Taoyuan County 333, Taiwan
{sjyen,leeys}@mail.mcu.edu.tw

Abstract. Mining *sequential patterns* is to discover sequential purchasing behaviors for most of the customers from a large amount of customer transactions. An example of such a pattern is that most of the customers purchased item B after purchasing item A, and then they purchased item C after using item B. The manager can use this information to promote item B and item C when a customer purchased item A and item B, respectively. However, the manager cannot know what time the customers will need these products if we only discover the sequential patterns without any extra information. In this paper, we develop a new algorithm to discover not only the sequential patterns but also the time interval between any two items in the pattern. We call this information the *time-gap sequential patterns*. An example of time-gap sequential pattern is that most customers purchased item A, and then they bought item B after m to n days, and then after p to q days, they bought item C. When a customer bought item A, the information about item B can be sent to this customer after m to n days, that is, we can provide the product information in which the customer is interested on the appropriate date.

Keywords: Data Mining, Frequent sequence, Time-gap sequential pattern, Transaction database.

1 Introduction

Because the capacity of the storage is getting larger, large amount of data can be stored in the database. Potentially useful information may be embedded in the large databases. Therefore, how to discover the useful information exists in such databases is becoming the popular field in the computer science. The purpose of *data mining* [3, 5, 6, 8, 9] is to discover the useful information from the large databases such that the quality of decision making can be improved.

A *transaction database* consists of a set of *transactions*. A transaction typically consists of the transaction identifier, the customer identifier (the buyer), the transaction date (or transaction time), and the items purchased in this transaction. An *itemset* is a non-empty set of items. An itemset X is *contained* in a transaction T, if $X \subseteq T$. A *sequence* is an ordered list of the itemsets. A sequence s is denoted as $<s_1, s_2, \ldots, s_n>$, where s_i is an itemset. A sequence $<a_1, a_2, \ldots, a_n>$ is *contained* in another sequence $<b_1, b_2, \ldots, b_m>$ if there exist integers $i_1 < i_2 < \ldots < i_n$, $1 \leqq i_k \leqq m$ such that

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 637–646, 2012.

$a_1 \subseteq bi_1$, …, $a_n \subseteq bi_n$, and $<a_1, a_2, \ldots, a_n>$ is a *subsequence* of sequence $<b_1, b_2, \ldots, b_m>$. A *customer sequence* is the list of all the transactions for a customer, which is ordered by increasing transaction time. A *customer sequence databases* contains all the customer sequences in an enterprise.

Mining sequential patterns [2, 4, 7] is to find the sequential purchasing behaviors for most of the customers from a customer sequence database. For example, there is a sequential pattern "<{computer}{printer}{scanner}> 70%" discovered from the transaction database in a computer store, which means that seventy percent of the customers bought printer after buying computer, and then they bought scanner after purchasing printer. If a customer purchases a computer, then we can use this information to predict that this customer will purchase printer in the future, and the advertisement about printers can be sent to the customer in advance. Similarly, when the customer purchases a printer, the advertisement and promotion about scanners can be sent to the customer in advance, such that the customer can come back to the store again. However, sequential patterns only provide that the purchased order for most of the customers is computer, printer and scanner. We cannot know how long the customer will need the printer and scanner after he/she purchasing a computer, such that we cannot know when the advertisement about printer and scanner should be sent to the customer.

Therefore, this paper defines the time-gap sequential patterns and proposes an algorithm to find time-gap sequential patterns. For example, a time-gap sequential pattern "<{shirts, necktie}(20~23){jacket}(56~60){shoes}> 70%" means that seventy percent of the customers bought both the shirts and necktie, and then they bought jacket after 20~30 time units, and then after 56~60 time units, they bought shoes. The definitions about mining time-gap sequential patterns are presented as follows: Let $I = \{i_1, i_2, \ldots, i_j\}$ be a set of all items. A *sequence* is an ordered list of items, which are ordered by transaction time. A *time-sequence* is a sequence in which each item is attached a transaction time, which is represented as $<i_1(t_1), i_2(t_2), \ldots, i_n(t_n)>$, where $i_k \in I$ and t_k is the time at which i_k was purchased ($t_{k-1} \leqq t_k$, $2 \leqq k \leqq n$). A *transaction-time sequence database* contains a set of records (e.g., Table 1). Each record includes a customer ID (CID) and a *transaction-time sequence*. A *transaction-time sequence* is a time-sequence which records all the items and their purchased date, which are purchased by a customer.

Table 1. A transaction-time sequence database

CID	Transaction-time sequence
1	A(5) ,B(10), C(19), D(27), E(32)
2	A(8), B(13), F(13), C(23), D(31)
3	A(9), B(14), C(23), D(31)
4	A(13), B(19), C(29), D(37)
5	A(15), B(21), F(21), D(28), A(36)
6	C(16), A(21), B(26), F(26), D(31)
7	E(18), C(27), A(34), B(40), F(40)
8	A(18) B(24), F(24) C(27) E(33)

A transaction-time sequence *c supports* a sequence *s* if *s* is contained in *c*. The *support* for a sequence *s* is the ratio of the number of the transaction-time sequences that supports *s* to the total number of the transaction-time sequences in the database. The *support count* for a sequence is the number of the transaction-time sequences that support this sequence. If the support for a sequence *s* satisfies the user-specified *minimum support* threshold, then *s* is called a *frequent sequence*. Otherwise, *s* is a *non-frequent sequence*. The *length* of a sequence *s* is the number of the items in the sequence. A sequence of length *k* is called a *k-sequence*, and a frequent sequence of length *k* a *frequent k-sequence*. In general, before generating the frequent sequences, we need to generate the *candidate sequences*, and scan the database to count the support for each candidate sequence to decide if it is a frequent sequence. A candidate sequence of length *k* is called a *candidate k-sequence*.

A *time-gap sequence* is a sequence in which the purchased time-gap between any two adjacent items in the sequence are recorded, which is represented as $<i_1, (t_1), i_2, (t_2), \ldots, (t_{n-1}), i_n>$, where $i_k \in I$ and t_k is a purchased time-gap between i_k and i_{k+1} ($1 \leqq k \leqq n$). The *length* of a time-gap sequence is the number of items in the sequence. A time-gap sequence with length *l* is called *time-gap l-sequence*. For instance, (a, (5), b, (15), d) is a time-gap 3-sequence. A time-sequence $<i_1(t_1), i_2(t_2), \ldots, i_n(t_n)>$ with length n can be transformed into a time-gap n-sequence $<i_1, (t_2 - t_1), i_2, (t_3 - t_2), \ldots, (t_n - t_{n-1}), i_n>$. A time-gap sequence which is transformed from a transaction-time sequence is called a *transaction time-gap sequence*.

The time-gap in a time-gap sequence can also be a time interval, which can be represented as $<i_1, (r_1), i_2, (r_2), \ldots, (r_{n-1}), i_n>$, where $i_k \in I$, and $r_k = l_k \sim u_k$ ($1 \leqq k \leqq n$) which is a time-gap between i_k and i_{k+1}. For any two time-gap sequences $q = <i_1, (s_1), i_2, (s_2), \ldots, (s_{n-1}), i_n>$ and $p = <j_1, (r_1), j_2, (r_2), \ldots, (r_{m-1}), j_m>$, if there exists $k_1 < k_2 < \ldots < k_n$, such that $i_1 = j_{k1}$, … and $i_n = j_{kn}$, and $s_1 \subseteq r_{k1}$, …, and $s_{n-1} \subseteq r_{k(n-1)}$, then time-gap sequence p *contains* time-gap sequence q. If a time-gap sequence s contains a sub-sequence of a transaction time-gap sequence and the transaction time-gap sequence is transformed from a transaction-time sequence *c*, then the transaction-time sequence *c supports* the time-gap sequence s. For example, transaction-time sequence $c = <a(15), b(22), c(26), d(31), e(39)>$ supports the time-gap 4-sequence $s = <a, (5\sim8), b, (3\sim6), c, (5\sim8), d>$, because *s* contains time-gap 4-sequence <a, (7), b, (4), c, (5), d> which is a sub-sequence of the time-gap sequence <a, (7), b, (4), c, (5), d, (8), e > which is transformed from *c*.

The *support* for a time-gap sequence is the ratio of the number of transaction- time sequences which support *s* to the total number of the transaction-time sequences in the database. If the support for a time-gap sequence is no less than the minimum support threshold, then the time-gap sequence is a *time-gap sequential pattern*. A time-gap sequential pattern with length k is called a time-gap k-sequential pattern. In addition to define the time-gap sequential patterns, we also propose an algorithm to find all the time-gap sequential patterns from the transaction-time sequence database. Because there is no previous approach for mining time-gap sequential patterns, we only evaluate the performance for our algorithm in the experiments.

2 Mining Time-Gap Sequential Patterns

In this section, we describe our algorithm for mining time-gap sequential patterns. Our algorithm first scans the transaction-time sequence database to count the support for each item and find the frequent items, that is, the frequent 1-sequence. After permuting and combining every two frequent 1-sequences, all the candidate 2-sequences can be generated. Our algorithm scans the transaction-time sequence database again to count the support for each candidate 2-sequence and find all the frequent 2-sequences. For each frequent 2-sequence, our algorithm generates a *2-time table* for the frequent 2-sequence. A *k-time table* for a *k*-sequence $< x_1, x_2, \ldots, x_k >$ ($k >$ 1) consists of two fields: customer identifier (CID) which support the *k*-sequence and the time points (TPs) which record the purchased time of each item in the *k*-sequence for the customer. For example, Table 2 and Table 3 are the 2-time tables for the frequent 2-sequences <AB> and <BC>, respectively, which are generated from Table 1. Our algorithm uses all the *k*-time tables to obtain time-gap *k*-sequential patterns and (*k*+1)-time tables.

Table 2. The 2-time table for sequence <AB>

CID	Time points
1	5, 10
2	8, 13
3	9, 14
4	13, 19
5	15, 21
6	21, 26
7	34, 40
8	18, 24

Table 3. The 2-time table for sequence <BC>

CID	Time points
1	10, 19
2	13, 23
3	14, 23
4	19, 29

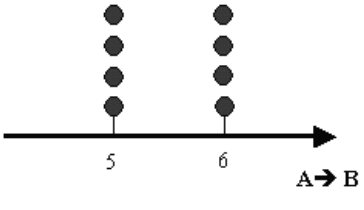


Fig. 1. The data points from Table 2 scattered in the one dimensional space A→B

Table 4. The time-gap table for sequence <AB>

CID	Time-gaps
1	5
2	5
3	5
4	6
5	6
6	5
7	6
8	6

After generating all the k-time tables (k > 1), the k-time table for a frequent k-sequence is transformed into a time-gap table for the k-sequence, which consists of two fields: CID and a sequence of time-gaps which are the differences between all the two adjacent items in the sequence of a customer. For a record (c, $\{t_1, t_2, \ldots, t_k\}$) in a k-time table, the transformed record in the time-gap table is (c, $< t_2 - t_1, t_3 - t_2, \ldots, t_k - t_{k-1}>$). The vector $< t_2 - t_1, t_3 - t_2, \ldots, t_k - t_{k-1}>$ in each record can be regarded as a *data point* in a (k-1)-dimensional space. Each dimension represents the gap between two purchased time points for the two adjacent items in the sequence. For example, the time-gap table for sequence <AB>, which is transformed from Table 2, is shown in Table 4. Figure 1 shows the data points in the time-gap table for the frequent 2-sequence <AB> (Table 4), which is scattered in the one dimensional space A→B. The 3-time table and the transformed time-gap table for sequence <ABC> are shown in Table 5 and Table 6, respectively. The two data points for the two time-gaps in Table 5 are scattered in the two dimensional space, which is shown in Figure 2. In Figure 2, x-axis represents the gap between the purchased dates of items A and B, and y-axis represents the gap between the purchased time points of items B and C.

Table 5. The 3-time table for sequence <ABC>

CID	Time points
1	5, 10, 15
2	12, 15, 17

Table 6. The time-gap table for sequence <ABC>

CID	Time-gap
1	5, 5
2	3, 2

After scattering all the data points in the time-gap table for a k-sequence in a (k-1)-dimensional space, our algorithm partitions the space into some units according to a *length parameter*ε, which a unit is denoted as $\{u_1, u_2, \ldots, u_{k-1}\}$ and $u_i = [l_i, h_i)$ $(1 \le i \le k-1)$

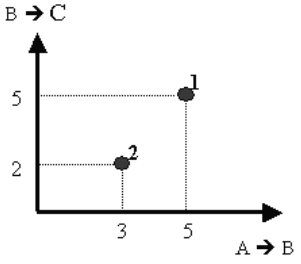


Fig. 2. The data points from Table 3 scattered in the two dimensional space

is the range in the *i*th dimension. Therefore, each dimension is partitioned into the units of the ranges [0, 1) and [r_j, r_{j+1}) (j ≥1, $r_{j+1} - r_j = \varepsilon$). For example, suppose the length parameterεis set to 3. The two dimensional space in Figure 2 can be partitioned into 9 units, which is shown in Figure 3. In Figure 3, unit U_1 represents the behavior that items A, B and C are purchased together; Unit U_2 represents the customers bought item A, and then they bought items B and C after 1~3 time units; Unit U_8 represents the customers bought item A, and then they bought item B after 1~3 time units and then after 4~6 time units, they bought item C.

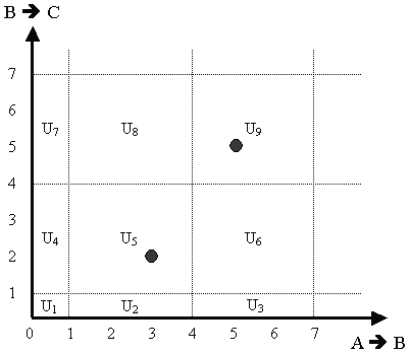


Fig. 3. The partitioned units for the two dimensional space in Figure 2

For a data point V=< v_1, v_2, …, v_d> and a unit U={u_1, u_2, …, u_d}(u_i= [l_i, h_i), (1 ≤ i ≤ d)) in a d-dimensional space, if $l_i \le v_i < h_i$ for all u_i, then the unit U *contains* the data point V. If the number of the data points contained in U is greater than a user-specified *minimum density* δ, then the unit U is called a *dense unit*. If a unit is not a dense unit, then the unit can be ignored and removed since this unit cannot represent the behaviors for most of the customers. Our algorithm applies CLIQUE [1] to cluster the dense units. For a d-dimensional space which has been partitioned into units, our algorithm first chooses a starting dense unit as follows: Let a unit be denoted as {u_1, u_2, …, u_d}(u_i= [l_i, h_i)). The unit whose l_1 is the smallest among all the dense units is chosen as the starting unit. If there are many units whose l_1 are the smallest, then the unit whose l_2 is the smallest among these units is chosen, and so on. From the starting dense unit, the adjacent dense units are continuously expanded and collected into a cluster r_1 via the 1st dimension until the adjacent unit is not dense. Through the 2nd

dimension, our algorithm continuously expands and collects the unit which adjacent to each units in r_1 into r_1 until r_1 cannot be expanded, that is r_1 cannot form a rectangle. After that, r_1 is continuously expanded through the 3rd dimension, and so on. After the expansion, a cluster r_1 can be obtained and r_1 is added into a set R. Our algorithm continues to choose a starting unit which is not included in R to be expanded until all the units are included in R.

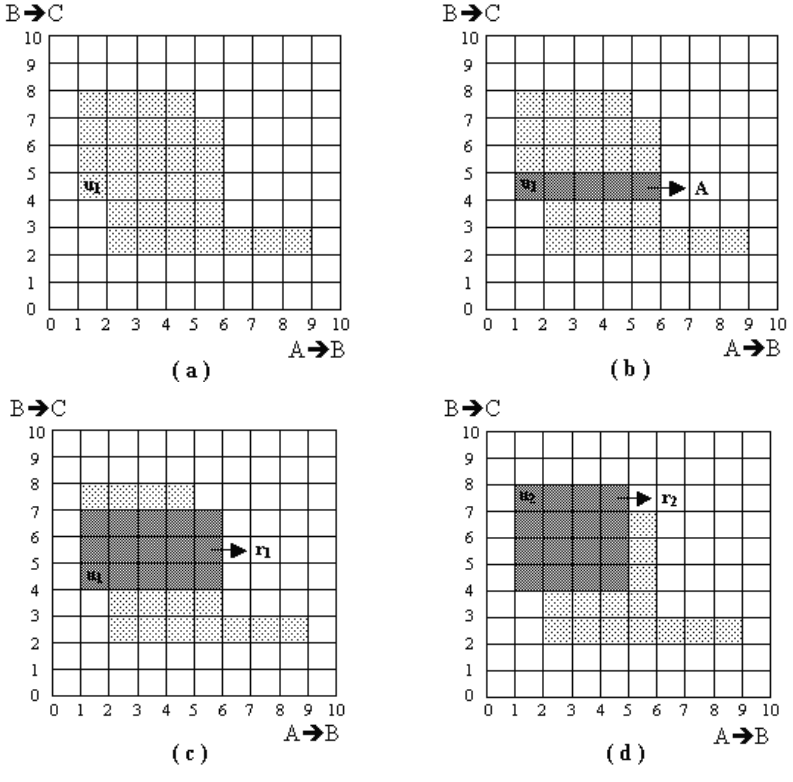


Fig. 4. Cluster the data points in the dense units

For example, suppose all the gray units in Figure 4 (a) are dense units in the two dimensional space for the frequent 3-sequence <ABC>. The unit u_1 is first chosen to be expanded through the 1st dimension A→B (Figure 4(b)), and then these expanded units are continuously expanded through the 2nd dimension B→C. The expanded cluster r_1 is shown in Figure 4(c) and the time-gap sequence generated from r_1 is <A (1~5) B (4~6) C>. Figure 4(d) shows the cluster r_2 which another starting unit u_2 is chosen to be expanded, and the time-gap sequence generated from r_2 is <A (1~4) B (4~7) C>. *The support for a cluster* (or a time-gap sequence) is the ratio of the number of the data points (or customers) in the cluster to the total number of the data points. If the support for a cluster is less than the minimum support, then this cluster can be ignored and removed from R, since this cluster cannot represent the customer purchasing behaviors. Otherwise, this cluster can form a time-gap sequential pattern.

After clustering the customers (or data points) and generating k-time-gap sequential patterns, our algorithm prunes the records in k-time tables, which are not contained in R, since these records are not useful to generate (k+1)-time-gap sequential patterns. If the number of the records in a k-time table is less than the minimum support, then the k-time table can be deleted. Our algorithm uses the pruned k-time tables to generate (k+1)-time tables. For any two frequent k-sequences p=< I_{p1}, I_{p2}, …, I_{pk}> and q=< I_{q1}, I_{q2}, …, I_{qk}>, if I_{p2}= I_{q1}, I_{p3}= I_{q2} , …, I_{pk}= $I_{q(k-1)}$, then a candidate (k+1)-sequence < I_{p1}, I_{p2}, …, I_{pk}, I_{qk}> can be generated and our algorithm performs intersection on the two sets of the CIDs in the two time tables for the two frequent sequences p and q. If the number of the CIDs in the intersect result is less than the minimum support, then the candidate can be deleted. Otherwise, the two k-time tables can be joined into a (k+1)-time table for the candidate. If the number of the records in the (k+1)-time table is no less than the minimum support threshold, then the candidate (k+1)-sequence is a frequent sequence. After that, our algorithm transforms the (k+1)-time table into a time-gap table, scatters the data points in the time-gap table into a k-dimensional space, and then cluster the dense units to find the time-gap sequential patterns about the frequent (k+1)-sequence.

For example, suppose the minimum support is 50%, that is the minimum support count is 4 for Table 1. The frequent sequences <AB> and <BC> can be joined into a candidate sequence <ABC>. After performing the intersections on the two sets of the CIDs in the two time tables for sequences <AB> and <BC>, which are shown in Table 2 and Table 3, respectively, the result set is {1, 2, 3, 4}. Since the number of the elements in the intersection result is 4 which is no less than the minimum support count, the two time-tables (Table 2 and Table 3) can be joined. The joined table is shown in Table 7, which is the 3-time table for sequence <ABC>. Since there are 4 customers in Table 7, the sequence <ABC> is a frequent sequence.

3 Experimental Results

Because there is no previous approach for mining time-gap sequential patterns, we only evaluate the performance of our algorithm. Owing to the real data is difficult to obtain, we generate the synthetic dataset to perform this experiment. Our synthetic dataset is generated by referring the synthetic data generation algorithm [10]. The parameters for generating the synthetic datasets are described as follows. For our experiment, the number of the distinct items is 1000; The average number of the transactions per customer is 45; The number of the customers (or the number of the transaction-time sequences) is 10K; The number of the maximal potentially frequent sequences is 1000; The average size of the maximal potential frequent sequences is 10 and the average time-gap between successive two items is 5. Besides, we set the length parameter$\varepsilon$to be 10 for partitioning the space into units, and the minimum density $\delta$to be 0.02 for identifying the dense units. Figure 5 and Figure 6 show the number of the time-gap sequential patterns generated by our algorithm and the execution times for our algorithm, respectively, under the minimum supports from 0.12 to 0.08.

From Figure 5, we can see that the number of the time-gap sequential patterns increases as the minimum support threshold decreases, since the number of the clusters whose supports are no less than the minimum support increases. From Figure 6, we can see that the execution time slightly increases as the minimum support decreases, since the number of the generated time tables and the number of the generated time-gap sequential patterns increase. Our algorithm needs to take more time to join the sequences and the time tables when the minimum support is small.

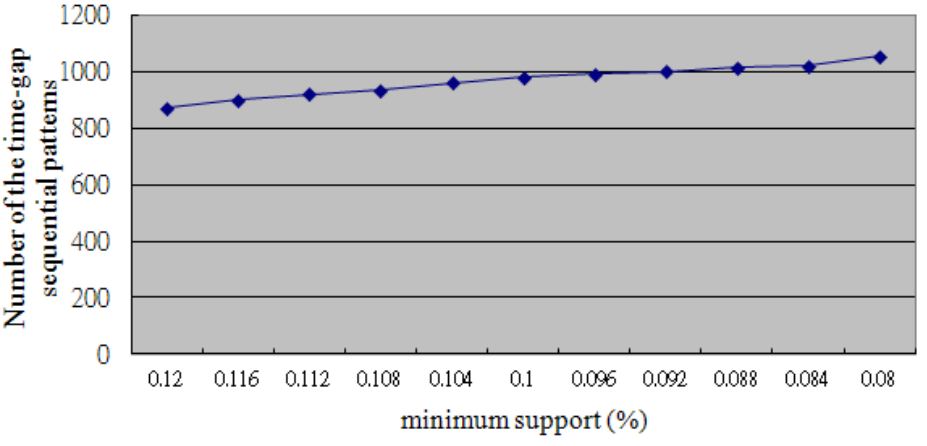


Fig. 5. The number of the time-gap sequential patterns generated by our algorithm

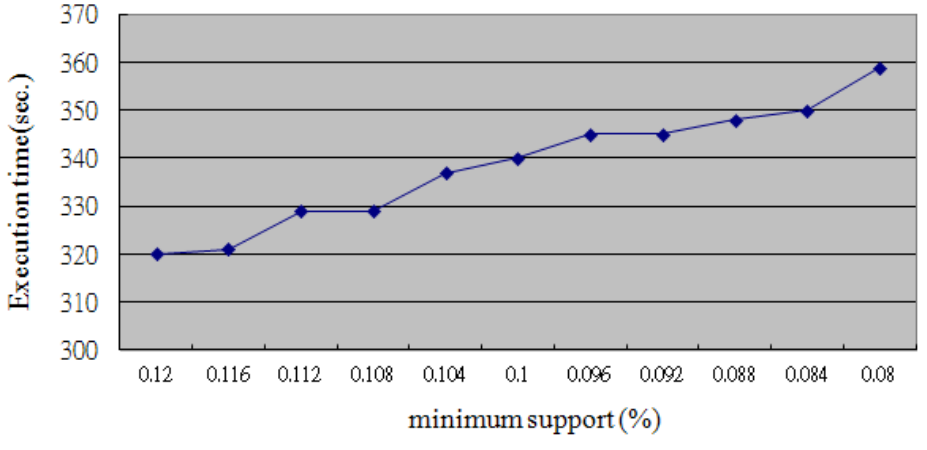


Fig. 6. The execution time for our algorithm

4 Conclusions

In this paper, we define the time-gap sequential patterns and propose an approach for mining time-gap sequential patterns from a customer transaction database. Our

algorithm generates frequent k-sequences ($k \geq 2$) and k-time tables for the frequent k-sequences, which are used to generate frequent (k+1)-sequences and (k+1)-time tables. We use CLIQUE clustering algorithm to cluster the representative data points in a time-gap tables which is transformed from k-time tables. The time-gap sequential patterns can be generated from the clusters with sufficient customers. Therefore, our algorithm can discover the time-gap sequential patterns which are the purchasing behaviors for most of the customers. The time-gap sequential patterns can be used to predict what time the customers will need the products when they purchased some other products.

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic Subspace Clustering of High Dimensional Data for Data Minig Applications. In: Proc. of the ACM SIGMOD Int'l Conference on Management of Data, Seattle, Washington, pp. 94–105 (June 1998)
2. Agrawal, R., Srikant, R.: Mining Sequential Patterns. In: Proc. International Conference on Data Engineering, pp. 3–14 (1995)
3. Ahmed, C.F., Tanbeer, S.K., Jeong, B.-S., Lee, Y.-K.: Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases. IEEE Transactions on Knowledge and Data Engineering 21(12), 1708–1721 (2009)
4. Gorawski, M., Jureczek, P., Gorawski, M.: Exploration of Continuous Sequential Patterns Using the CPGrowth Algorithm. In: Nguyen, N.T., Zgrzywa, A., Czyżewski, A. (eds.) Advances in Multimedia and Network Information System Technologies. AISC, vol. 80, pp. 165–172. Springer, Heidelberg (2010)
5. Hong, T.P., Lin, C.W., Wu, Y.L.: Incrementally fast updated frequent pattern trees. Expert Systems with Applications: An International Journal 34(4), 2424–2435 (2008)
6. Han, J., Mao, R., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation: a frequent-pattern tree approach. Data Mining and Knowledge Discovery 8, 53–87 (2004)
7. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Daya, U., Hsu, M.C.: PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. In: International Conference on Data Engineering, pp. 215–224 (2001)
8. Yen, S.-J., Lee, Y.-S., Wu, C.-W., Lin, C.-L.: An Efficient Algorithm for Maintaining Frequent Closed Itemsets over Data Stream. In: Chien, B.-C., Hong, T.-P., Chen, S.-M., Ali, M. (eds.) IEA/AIE 2009. LNCS (LNAI), vol. 5579, pp. 767–776. Springer, Heidelberg (2009)
9. Yen, S.J., Lee, Y.S., Wang, C.K., Wu, J.W., Ouyang, L.Y.: The Studies of Mining Frequent Patterns Based on Frequent Pattern Tree. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS (LNAI), vol. 5476, pp. 232–241. Springer, Heidelberg (2009)
10. IBM Synthetic Data Generator, `http://www.almaden.ibm.com/software/quest/Resorces/index.shtml`

Intelligent Post-processing via Bounding-Box-Based Morphological Operations for Moving Objects Detection*

Chun-Ming Tsai

Department of Computer Science, Taipei Municipal University of Education,
No. 1, Ai-Kuo W. Road, Taipei 100, Taiwan, R.O.C.
cmtsai2009@gmail.com, cmtsai@tmue.edu.tw

Abstract. The detection of moving objects is a critical first step in video surveillance. Numerous background subtraction, frame differencing, optical flow algorithms and a number of post-processing techniques (including noise removal, binary morphological operations, and area thresholding) are used to extract the moving objects. However, these post-processing methods are time consuming and inefficient in real-time applications; for example, noise removal and binary morphological operations require scanning the video frame many times. The study presents an innovative post-processing technique, using bounding-box-based morphological operations, for grouping concentrated connected components and the removal of spread and small connected components for moving objects detection. Results demonstrate that the proposed method is more effective and efficient than traditional post-processing methods.

Keywords: Video surveillance, moving object detection, bounding-box-based morphological, post-processing.

1 Introduction

Moving object detection (MOD) is the fundamental step in a visual surveillance system, as well as in a variety of intelligent applications of computer vision, including traffic systems, parking lots, healthcare systems, home-care systems, and homeland security.

The steps of the traditional MOD method include: (1) a differencing image is obtained by background subtraction [1], optical flow [2], frame differencing [3], or hybrid methods [4-5]; (2) a thresholding image is obtained using a fixed threshold (FT) value; (3) noises are removed by a noise filter (NF); (4) broken objects are grouped by binary morphological (BM) operations; and, (5) the bounding box of the moving object is obtained by connected component labeling (CCL). If the threshold value is determined improperly, many broken objects and noises can be produced. In these conditions, the traditional BM operations (erosion, dilation, or a combination of these two operations) [6] are employed to remove noises and to merge broken objects.

* This paper is supported by the National Science Council, R.O.C., under Grants NSC 99-2221-E-133-002- and NSC 100-2221-E-133-004-.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 647–657, 2012.

However, these operations are time-consuming for real-time applications unless using "hierarchical queues" [7] to speed the implementation. For example, let A denotes a binary image and S denotes a structuring element. Suppose the sizes of A and S are $N \times N$ and $M \times M$, respectively. The time complexity for the dilation or erosion operator is $N^2 \times M^2$. Another problem is determining the shape and the size of the structuring element in order to obtain a "good" result. Further, if the objects are slow-moving, many broken objects will be produced. If using BM operations for these cases, these broken objects will be removed. Hence, innovative bounding-box-based morphological (BBM) operations are proposed to merge and recover broken CCs and remove small CCs. The BBM operations include BB-based dilation, erosion, opening, and closing operations. These operations will be described as follows.

2 Bounding-Box-Based Morphological (BBM) Operations

2.1 Geometry-Based BB Merging Method

The BBM operations are based on a geometry-based BB merging method [8]. The geometry types of the BB include (1) intersection at the edge (E_T, E_B, E_L, and E_R represent the cases where one BB intersects another at the top edge, bottom edge, left edge, and right edge, respectively.); (2) intersection at the corner (C_{LT}, C_{RT}, C_{LB}, and C_{RB} represent the cases where one BB intersects another at the left-top, right-top, left-bottom, and right-bottom corners, respectively.); (3) embedding (E_O represents the case where one BB is embedded into another). The formula of the geometry-based BB merging method can be defined as follows. With I and J as sets in the BB, the *geometry-based BB merging method* of I and J, denoted as $I \cup_{\mathrm{BB}} J$, is defined as

$$I \cup_{BB} J = \left\{ \begin{array}{l} K \mid K.top = \min(I.top, J.top), \\ K.bottom = \max(I.bottom, J.bottom), \\ K.left = \min(I.left, J.left), \\ K.right = \max(I.right, J.right), \\ for I \in BBs, J \in BBs. \end{array} \right\}, \tag{1}$$

where *min* and *max* represent the minimum and maximum functions, respectively.

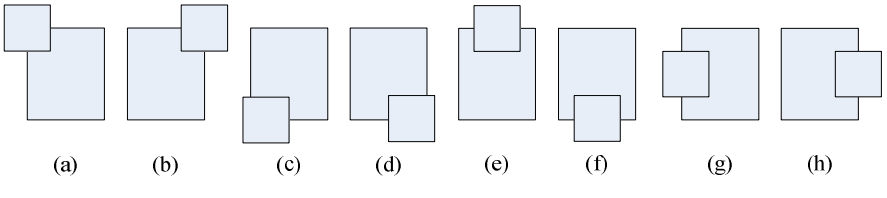


Fig. 1. Examples for intersection of the BBs. Figures 1(a) to 1(d) are examples of intersection at the corners, while 1(e) to 1(h) are examples of intersection at the edges.

Figure 1 shows examples for the geometry types of the intersection of the BBs. Figures 1(a) to 1(d) show the intersections at the corners, which are represented by

C_{LT}, C_{RT}, C_{LB}, and C_{RB}, respectively. Figures 1(e) to 1(h) show the intersections at the edges, are represented by E_T, E_B, E_L, and E_R, respectively. From the definition of the geometry-based BB merging method, if two bounding boxes belong to the above-mentioned BBs with one of the nine geometry types, a new merged BB is produced.

2.2 BB-Based Dilation and Erosion Operations

A discussion of the BBM operations in detail follows for BB-based dilation and BB-based erosion. These operations are fundamental steps in the BB-based morphological processing. With I as the set in the BBs and d_D as the set in the dilation constants (left constant: d_L; right constant: d_R; top constant: d_T; and bottom constant: d_B), the *BB-based dilation* of I by d_D, denoted by $I \oplus_{BB} d_D$, is defined as

$$I \oplus_{BB} d_D = \begin{Bmatrix} I \mid I.left + d_L, I.right + d_R, \\ I.top + d_T, I.bottom + d_B \\ for I \in BBs, d_D \in (d_L, d_R, d_T, d_B). \end{Bmatrix}, \tag{2}$$

Figure 2(a) shows a simple BB. The dashed line in Fig. 2(b) shows the original BB for reference, and the solid line shows the result obtained by using the dilation constant. Herein, the left, right, top, and bottom dilation constants are equal. Figure 2(c) shows the dilation that is achieved with different dilation constants. The left dilation constant is equal to the right dilation constant. The top dilation constant is equal to the bottom dilation constant.

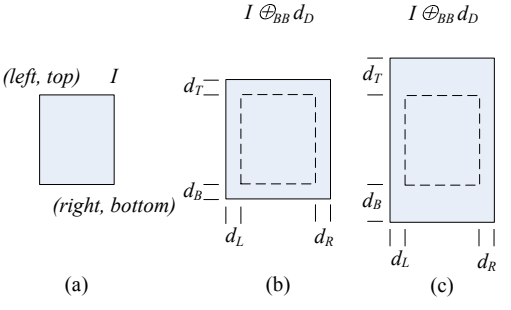


Fig. 2. Example of BB-based dilation operation. (a) Original BB I with left, top, right, and bottom coordinate. (b) Result obtained by applying BB-based dilation with equal dilation constants ($d_D = d_L = d_R = d_T = d_B$). (c) Result obtained by applying BB-based dilation with different dilation constants ($d_L = d_R$, $d_T = d_B$).

With I as the set in the BBs and d_E as the set in the erosion constants (left constant: d_L; right constant: d_R; top constant: d_T; and bottom constant: d_B), the *BB-based erosion* of I by d_E, denoted as $I \ominus_{BB} d_E$, is defined as

$$I \oplus_{BB} d_D = \begin{Bmatrix} I \mid I.left + d_L, I.right + d_R, \\ I.top + d_T, I.bottom + d_B \\ for I \in BBs, d_D \in (d_L, d_R, d_T, d_B). \end{Bmatrix}, \tag{3}$$

Figure 3 shows a process similar to that shown in Fig. 2. As before, BB *I* is shown as a dashed line for reference in Figs. 3(b) and (c). The shaded regions with solid lines indicate the results obtained by applying the BB-based erosion operation. Figure 3(b) shows the results obtained with equal dilation constants and Fig. 3(c) shows the results obtained with unequal dilation constants. (Note that the original BB is eroded down to a line.)

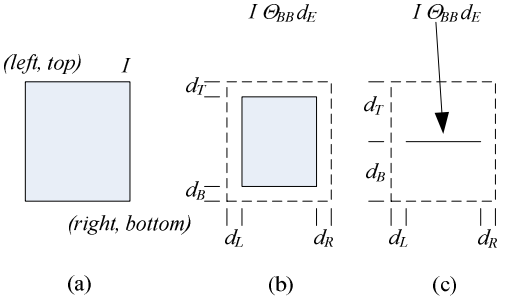


Fig. 3. Example of BB-based erosion operation. (a) BB *I* with left, top, right, and bottom coordinates. (b) Result obtained by applying BB-based erosion with equal erosion constants ($d_E = d_L = d_R = d_T = dB$). (c) Result obtained by applying BB-based erosion with different erosion constants ($d_L = d_R$, $d_T = d_B$).

2.3 BB-Based Opening Operation

Following the fixed thresholding, the binary results can produce much noise pertaining to the connected components. These noises may arise from small movements such as leaves and tree branches moving in the wind, the environment, the camera's hardware, light shadows and human shadows. They can be eliminated by the BB-based opening operation. The proposed BB-based opening operation is described as follows. The *BB-based opening* of BBs *I* and *J* by opening constant *c* ($c \in d_E$ and d_D), denoted $(I, J)\ o_{BB}\ c$, is defined as

$$(I,J)o_{BB}c = \left((I\Theta_{BB}d_E)\cup_{BB}(J\Theta_{BB}d_E)\right)\oplus_{BB} d_D, \tag{4}$$

which indicates that the opening of BBs *I* and *J* by *c* is simply the BB-based erosion of *I* by d_E and *J* by d_E, followed by the geometry-based BB merging method, to merge the result and the BB-based dilation of the result by d_D.

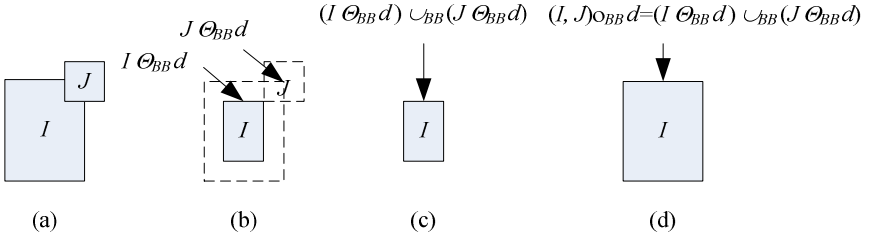


Fig. 4. Example of BB-based opening operation. (a) Original BBs *I* and *J*. (b) Result obtained by applying BB-based erosion with equal erosion constant for (a). (c) Result obtained by applying geometry-based BB merging method for (b). (d) Final result for BB-based opening operation.

Figure 4(a) shows two simple BBs, *I* and *J*. Note that the two BBs intersect at the corner. Figure 4(b) shows the result when the BBs *I* and *J* are eroded by equal erosion constants. Note that the *J* BB has been subsequently removed. The result obtained by applying the geometry-based BB merging method to Fig. 4(b) is shown in Fig. 4(c). Figure 4(c) is dilated by equal dilation constants and the final result of the BB-based opening operation is shown in Fig. 4(d). It can be noted that, depending on the dilation and erosion constants, either the two intersected BBs are split into two, or the bigger BB is retained and the smaller BB is removed. In the real applications, Fig. 4(c) is omitted. Thus, the BB-based opening can remove the small BBs and reserves the large BBs.

2.4 BB-Based Closing Operation

Following the fixed thresholding, the binary results can produce many broken connected components. These broken connected components can be divided into concentrated and spread connected components, which are removed by the BB-based opening operation. The concentrated connected component results result from the large threshold value. Herein, the BB-based closing operation is proposed to merge these concentrated connected components into a complete object. This BB-based closing operation is described as follows. The *BB-based closing* of BBs *I* and *J* by the closing constant *c* ($c \in d_D$ and d_E), denoted $(I, J) \bullet_{BB} c$, is defined as

$$(I,J)\bullet_{BB} c = \left(\left(I \oplus_{BB} d_D\right) \cup_{BB} \left(J \oplus_{BB} d_D\right)\right) \Theta_{BB} d_E, \tag{5}$$

which implies that the closing of the BBs *I* and *J* by *c* is simply the BB-based dilation of *I* by d_D and *J* by d_D, followed by the geometry-based BB merging method to merge the result and the BB-based erosion of the results by d_E.

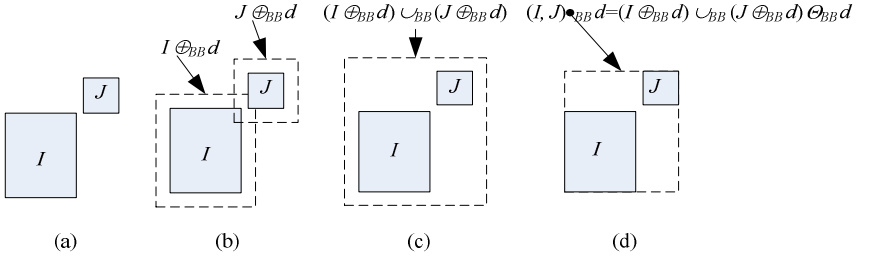


Fig. 5. Example of BB-based closing operation. (a) Original BBs I and J. (b) Result obtained by applying BB-based dilation with equal dilation constant for (a). (c) Result obtained by applying geometry-based BB merging method for (b). (d) Final result for BB-based closing operation.

Figure 5(a) shows two simple BBs, *I* and *J*. Figure 5(b) shows the result of the dilation of BBs *I* and *J* by equal dilation constants. Note that the two BBs intersect at the corners. The result obtained by applying the geometry-based BB merging method to Fig. 5(b) is shown in Fig. 5(c). Figure 5(c) is eroded by equal erosion constants and the final result of the BB-based closing operation is shown in Fig. 5(d). Notably, the two BBs are merged into one, which represented by the dashed line.

3 Experimental Results

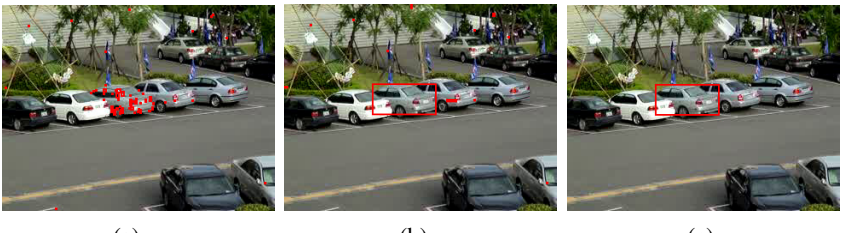
The above-mentioned methods are implemented by using Visual C# 2008 on a 3400 MHz Pentium 4 CPU. The video clips tested in our experiments were recorded at the campus of Taipei Municipal University of Education. The web camera used for this purpose is a Logitech QuickCam with video frame resolution of 320 × 240 pixels and a frame rate of 25 fps. The total time taken for capturing the video clips is approximately 1.5 minutes, which includes 2,272 frames. The video clips include moving humans, a slow-moving car, a moving car, and leaves rustling in the wind.

The steps of the tested MOD method include: (1) a differencing image is obtained by two-frame differencing (2FD); (2) a thresholding image is obtained by a fixed threshold value (FT); (3) connected component objects are obtained by connected component labeling (CCL); (4) the centralized BBs are merged and the small spread BBs are removed by the proposed BBM operations. The results of the MOD and the execution times are compared with traditional MOD method. In other words, the comparison is between 2FD + FT + CCL + BBM and 2FD + FT + noise filter (NF) + binary morphological (BM) operations + CCL methods. The sizes of the noise filter and the binary morphological operations are set as 5 × 5.

(a) (b) (c)

Fig. 6. Example of slow-moving objects. (a) Frame #698. (b) Frame #699. (c) Binarization result.

The parameters for BBM operations are derived from a supervised learning method that inputs training bounding-boxes. A support vector machine (SVM) is used to find a hyper-plane that splits the training samples into two classes with the widest margin between them. More detail for training procedures can be found at [11].

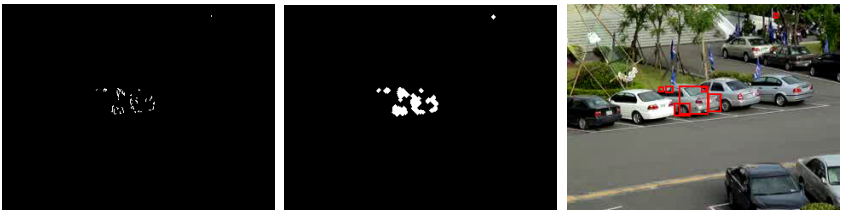


(a) (b) (c)

Fig. 7. Example of slow-moving objects detect by the proposed BB-based morphological (BBM) operations. (a) CCL's results; (b) grouping results obtained by the proposed BB-based closing operation; (c) removal results obtained by the proposed BB-based opening operation.

In the video clip showing slow-moving objects, the motion is insignificant and the slow-moving objects can be detected by the proposed BBM operations. An example of this is shown in Fig. 6, in which a car is backing slowly from the parking lot. The original frames #698 (6(a)) and #699 (6(b)) are subtracted by two-frames differencing. Figure 6(c) shows the binarization result obtained by a fixed thresholding value.

Figures 7(a) to 7(c) show the results by the proposed method: (a) the CCL's result; (b) the grouping result obtained by our BB-based closing operation; and (c) the removal result obtained by the proposed BB-based opening operation, respectively. The car has been properly detected by the proposed method.



(a) (b) (c)

Fig. 8. Example of slow-moving objects detect by traditional binary morphological (BM) operations. (a) Noise-removing results. (b) Grouping results obtained by BM operations. (c) CCL's results.

Figures 8(a) to 8(c) show the results by traditional methods: (a) a noise-removing result as obtained by a noise filter; (b) a grouping result as obtained by BM operations; and (c) CCL's result, respectively. From this figure, the slow-moving car is still a broken object.

From the above comparisons, it can be seen that the proposed method can extract the slow-moving objects whereas the traditional method cannot. To obtain the whole object, the traditional result (8(c)) requires additional processing.

The performance time analyses of the proposed method and traditional methods for the campus video clip are shown in Table 1, which shows that the proposed BBM operations are superior to the traditional BM operations.

Table 1. Comparison of time performance by using the proposed and the traditional methods for campus video clip (320×240 pixels)

Methods	Average execution times
Proposed method	*8.866 (ms)*
Traditional method	*32.368 (ms)*

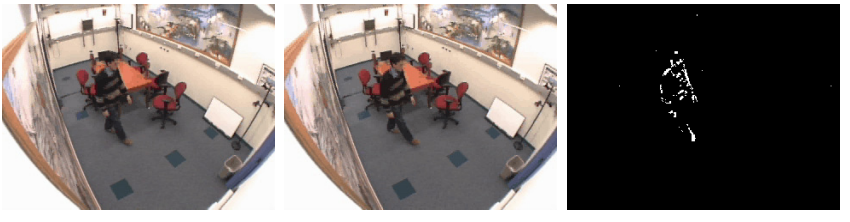
To evaluate the performance of the motion objects detection, connected-component measures [9] are applied. A *connected-component* is classified as *small*, *median*, and *large* by the following rules. A connected-component is *small* if the number of the binary pixels *NP* in the component is less than 4 and greater than 1, *large* if *NP* is greater than the image area multiplied by a constant, *k*, which was predetermined as 0.001, and otherwise, it is *median*. Furthermore, a *connected-component ratio* is the number of connected-components divided by the total number of connected-components.

The average ratios of small, median, and large connected-components for the campus video are listed in Table 2. From this table, the score of small CC ratio using the proposed method is smaller than that of the traditional method. This implies that the traditional method produces more noise or broken moving objects than the proposed method. The score of a large CC ratio of the proposed method is larger than the traditional method. This implied that the proposed method produces more whole moving objects than traditional method. According to the evaluation, the proposed method has a better performance than the traditional method.

Table 2. Average ratios of small, median, and large connected-components for campus video clip

Methods	Small CC ratio	Median CC ratio	Large CC ratio
Proposed method	*0.0078*	*0.0560*	*0.1677*
Traditional method	*0.0856*	*0.0771*	*0.0679*

To further test the post-processing algorithms, the proposed and the traditional methods were applied to the ground truth data, which was made available by Prati et al. [10]. They provided three hundred images from an indoor sequence containing a moving person along with a manual segmentation into foreground (human), shadow, and background. Herein, only the foreground and background are used.

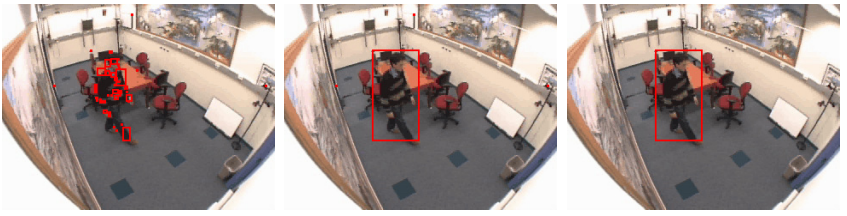


(a) (b) (c)

Fig. 9. Example of moving object in Prati et al.'s video clip. (a) Frame #269. (b) Frame #270. (c) Binarization result.

In the Prati et al.'s video clip showing a moving object, it can be seen that the human's motion is significant. The moving object can be detected by the proposed post-processing. An example for this case is shown in Fig. 9. A man is moving in the intelligent room. The original frames #269 (9(a)) and #270 (9(b)) are subtracted by two-frames differencing. Figure 9(c) shows the binarization result obtained by a fixed thresholding value.

Figures 10(a) to 10(c) show the results using the proposed method: (a) CCL's result; (b) a grouping result as obtained by the proposed BB-based closing operation; and (c) a removal result as obtained by the proposed BB-based opening operation, respectively. The man has been detected by the proposed method.



(a) (b) (c)

Fig. 10. Example of moving object in Prati et al.'s video clip which detect by the proposed post-processing. (a) CCL's results; (b) grouping results obtained by the proposed BB-based closing operation; (c) removal results obtained by the proposed BB-based opening operation.

Figures 11(a) to 11(c) show the results using the traditional post-processing method: (a) a noise-removing result as obtained by a noise filter; (b) a grouping result as obtained by BM operations; and (c) CCL's result, respectively. From this figure, the moving man is still a broken object.

(a) (b) (c)

Fig. 11. Example of moving object in Prati et al.'s video clip. The object detected by traditional binary morphological (BM) operations. (a) Noise-removing results. (b) Grouping results obtained by BM operations. (c) CCL's results.

From the above comparisons, it can be seen that the proposed method can extract the moving object and the traditional method cannot. To obtain the object, the traditional result (11(c)) needs additional processing.

Table 3. Comparison of detection results obtained by using the proposed and the traditional grouping methods for Prati et al.'s video clips (320×240)

Methods	The proposed method	The traditional method
Total objects	*218*	*218*
True positive rate (whole objects)	*81.652%*	*45.871%*
True positive rate (partial objects)	*18.348%*	*10.599%*
True positive rate (broken objects)	*0*	*43.53%*
False negative rate	*0*	*0*
False positive rate	*0*	*77.064%*

For the Prati et al.'s video clips, the detection results of the proposed and the traditional post-processing methods are shown in Table 3. In this video clip, one person is entering the intelligent room, doing his arm stretch, and walking in the room. His

motion is significant when he is near the CCD camera. His motion is insignificant when he is far from the CCD camera. From Table 3, the broken objects in the true positive rate for traditional grouping method is high. This indicates that the traditional method results will require additional processing. Furthermore, the false positive rate of the traditional method is high, indicating that the traditional method produces much noise. Thus, another noise filter needed to be applied to remove the noise. From Table 3, it is evident that the proposed method is superior to a traditional post-processing method.

Table 4. Average ratios of small, median, and large connected-components for Prati et al.'s video clip

Methods	Small CC ratio	Median CC ratio	Large CC ratio
Proposed method	*0.0000*	*0.1308*	*0.6007*
Traditional method	*0.4065*	*0.2134*	*0.1593*

The average ratios of small, median, and large connected-components for the Prati et al.'s video are listed in Table 4. From this table, the score of the small CC ratio of the proposed method is smaller than the traditional method, implying that the traditional method produces more noise or more broken moving objects than the proposed method. The score of large CC ratio of the proposed method is larger than the traditional method, which implied that the proposed method produces more whole moving objects than the traditional method. According to the evaluation, the proposed method has a better performance than the traditional method.

4 Conclusions

Innovative bounding-box-based morphological operations are proposed for moving object detection to be used in post-processing stage. This stage includes grouping concentrated and broken connected components and removing spread and small connected components. Results show that the proposed method is more effective and efficient than traditional methods. In particular, the proposed BBM operations can properly detect slow-moving objects.

References

1. Tsai, D.M., Lai, S.C.: Independent component analysis-based background subtraction for indoor surveillance. IEEE Trans. on IP 18(1), 158–167 (2009)
2. Wixson, L.: Detecting Salient Motion by Accumulating Directionally Flow. IEEE Trans. on PAMI 22(8), 774–779 (2000)
3. Kim, C., Hwang, J.: Fast and Automatic Video Object Segmentation and Tracking for Content-Based Applications. IEEE Trans. on CSVT 12, 122–129 (2002)
4. Tian, Y.L., Hampapur, A.: Robust Salient Motion Detection with Complex Background for Real-time Video Surveillance. In: Proc. of IEEE Workshop on MVC, Breckenridge, CO, USA, vol. 2, pp. 30–35 (January 2005)

5. Yoo, Y., Park, T.-S.: A moving object detection algorithm for smart cameras. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, AK, USA, pp. 1–8 (June 2008)
6. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 3rd edn. Prentice Hall (2008)
7. Soille, P.: Morphological Image Analysis: Principles and Applications, 2nd edn. Springer (2002)
8. Tsai, C.M., Lee, H.J.: Recognition of Road Names in Street Maps. In: Proc. of CSIA, Hsinchu, Taiwan, pp. 331–338 (1997)
9. Tsai, C.M., Lee, H.J.: Binarization of Color Document Images via Luminance and Saturation Color Features. IEEE Trans. on Image Processing 11(4), 434–451 (2002)
10. Prati, A., Mikic, I., Cucchiara, R., Trivedi, M.M.: Analysis and Detection of Shadows in Video Streams: A Comparative Evaluation. In: IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, vol. 2, pp. 571–576 (December 2001)
11. Tsai, C.-M.: An Intelligent Method to Extract Characters in Color Document with Highlight Regions. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 143–152. Springer, Heidelberg (2011), doi:10.1007/978-3-642-21827-9_15

An Integrated Agent Model for Attention and Functional State

Tibor Bosse[1], Rianne van Lambalgen[1], Peter-Paul van Maanen[1,2], and Jan Treur[1]

[1] Vrije Universiteit Amsterdam, Department of Artificial Intelligence,
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
{tbosse,rm.van.lambalgen,treur}@few.vu.nl
[2] TNO Human Factors, P.O. Box 23, 3769 ZG, Soesterberg, The Netherlands
peter-paul.vanmaanen@tno.nl

Abstract. To provide personalized intelligent ambient support for persons performing demanding tasks, it is important to have insight in their state of attention. Existing models for attention have difficulties in distinguishing between stressed and relaxed states. To solve this problem, this paper proposes to extend an existing model for attention with a model for 'functional state'. In this integrated agent model, output of a functional state model (experienced pressure) serves as input for the attention model; the overall amount of attention is dependent on the amount of experienced pressure. An experiment was conducted to test the validity of the integrated agent model against the validity of an earlier model based on attention only. Results pointed out that the integrated model had a higher validity than the earlier model and was more successful in predicting attention.

1 Introduction

For persons performing complex and demanding tasks, it is crucial to have sufficient *attention* for the various subtasks involved. This is particularly true for tasks that involve the continuous inspection of (computer) screens. For instance, an air traffic controller inspecting the movements of aircrafts can not permit him- or herself to miss part of the events that occur. The same holds for a naval operator monitoring the movements of hostile vessels on a radar screen. In such situations, a person may be supported by an intelligent ambient agent system [1], that keeps track of where his or her attention is, and provides some personalized assistance in case the attention is not where it should be, see, e.g., [4], [13].

The current paper is part of a larger project that aims to develop such an agent-based intelligent ambient support system. The main application domain of this system will be naval missions, and one of its main goals will be to support naval operators that work in the control room of the vessels. For example, in case such an operator is directing its attention on the left part of a radar screen, but ignores an important contact that just entered the radar screen from the right, such a system may alert him or her about the arrival of that new contact. To be able to provide this kind of intelligent personalized support, the system somehow needs to maintain a model of the cognitive

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 658–668, 2012.

state of the person: in this case the human's focuses of attention[1]. It should have the capability to attribute mental, and in particular attentional (e.g., [12]) states to the human, and to reason about these.

In previous work, an initial version of such a model has been developed [3], and evaluated positively [13]. This model takes two types of sensor information as input, namely information about the human's gaze (e.g., measured by an eye tracker), and characteristics of stimuli (e.g., the colour and speed of airplanes on radar screens, or of the persons on surveillance images). Based on these types of information, it estimates where the human's attention is and uses this to decide whether adaptive support is needed.

However, one important shortcoming of that model is that it assumes that the *total amount of attention* a person can spend at a particular time point (in the remainder of this paper referred to as *A(t)*) is static and known beforehand. However, it is known from literature like [8] that factors like *A(t)* usually vary over time, depending on states and characteristics of a person [6]. More specifically, it may depend on a human's *functional state*. According to [11], (an operator's) functional state refers to 'the multidimensional pattern of processes that mediate task performance under stress and high workload, in relation to task goals and their attendant physiological and psychological costs'. It is usually assumed to be based on notions like the person's *experienced pressure* and *exhaustion*.

In recent years, researchers have started to develop computational models for the concept of functional state. One of the most sophisticated models is presented in [2]. This model takes task demands, situational aspects, and some of the human's personal characteristics as input, and uses these to asses the human's functional state. Inspired by these developments, the goal of the research reported in the current article has been to develop an integrated agent model for attention and functional state. Our main hypothesis is that the integrated model (which will be called *attention+* from now on) has a higher validity (i.e., is more accurate in estimating where a person's attention is) than the original model (called *attention-*) from [3].

The structure of this paper is as follows. First, the original models for attention [3] and functional state [2] will be briefly described, as well as a proposal to integrate them. Next, an experiment is described that has been performed to compare the validity of the *attention+* model with that of the *attention-* model. The context of the experiment is a shooting task, which is representative for complex tasks that are currently performed in the naval domain. After that, the results of the experiment are analysed. The paper is concluded by a discussion.

2 The Two Submodels and Their Integration

The introduced integrated agent model (i.e. the *attention+* model) is composed of two main submodels, namely (1) a basic attention model (i.e., the *attention-* model) and (2) a functional state model. Below, both of them will be briefly summarised. Next, a detailed explanation is provided about how they are combined.

[1] Note that in this paper, a rather wide definition of the term 'attention' is used, covering not only visual attention, but also 'mental' attention for objects that have been observed some time earlier, often referred to as 'situational awareness' [5].

2.1 Attention Submodel

The attention submodel was taken from [3]. The model uses three types of input: information about the human's *gaze direction*, about *locations* (or spaces) and about *features* of objects on the screen (see Figure 1, where the circles denotes the italicised concepts, and the arrows indicate influences between them). Based on this, at each time point t it makes an estimation of the *current attention distribution*: an assignment of attention values $AV(s,t)$ to a set of attention spaces s at that time. The attention distribution is assumed to have a certain persistency. At each point in time the new *attention level* is related to the previous attention, by:

$$AV(s,t) = \lambda \cdot AV(s,t\text{-}1) + (1 - \lambda) \cdot AV_{norm}(s,t)$$

Here, λ is the decay parameter for the decay of the attention value of space s at time point $t-1$, and $AV_{norm}(s,t)$ is determined by normalisation for the total amount of attention $A(t)$, described by:

$$AV_{\text{norm}}(s,t) = \frac{AV_{\text{new}}(s,t)}{\sum_{s'} AV_{\text{new}}(s',t)} \cdot A(t)$$

$$AV_{\text{new}}(s,t) = \frac{AV_{\text{pot}}(s,t)}{1 + \alpha \cdot r(s,t)^2}$$

Here, $AVnew(s,t)$ is calculated from the potential attention value of space s at time point t and the relative distance of each space s to the gaze point (the centre). The term $r(s,t)$ is taken as the Euclidian distance between the current gaze point and s at time point t (multiplied by an importance factor α which determines the relative impact of the distance to the gaze point on the attentional state, which can be different per individual and situation):

$$r(s,t) = d_{\text{eucl}}(gaze(t), s)$$

The potential attention value $AVpot(s,t)$ is a weighted sum of the features of the space (i.e., of the types of objects present) at that time (e.g., luminance, colour):

$$AV\text{pot}(s,t) = \sum_{maps\ M} M(s,t) \cdot w_M(s,t)$$

For every feature there is a saliency map M, which describes its potency of drawing attention (e.g., [12]). Moreover, $M(s,t)$ is the unweighted potential attention value of s at time point t, and $w_M(s,t)$ is the weight used for saliency map M, where $1 \leq M(s,t)$ and $0 \leq wM(s,t) \leq 1$.

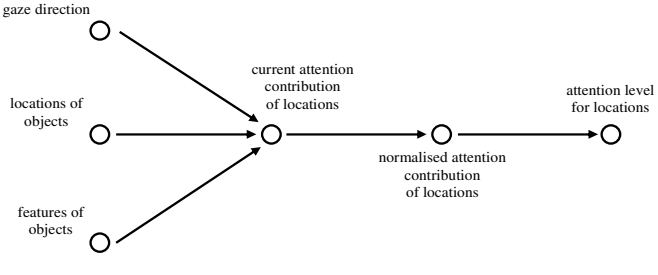


Fig. 1. Overview of the Attention model

For a more detailed description of the model and the underlying theories, see [3], [4].

2.2 Functional State Submodel

The functional state (FS) submodel was adopted from [2] and determines a person's *functional state* as a function of task properties and personal characteristics. The model is based on two different theories: (1) the cognitive energetic framework [10], which states that effort regulation is based on human recourses and determines human performance in dynamic conditions; (2) the idea, that when performing sports, a person's generated power can continue on a *critical power* level without becoming more exhausted [9]. The FS of a human represents the dynamical state of the person. In the model (see Figure 2), this is defined by a combination of exhaustion, motivation and experienced pressure, but also the amount of generated and provided effort. Due to space limitations no further details of the model are provided here. However, for a detailed description, see [2].

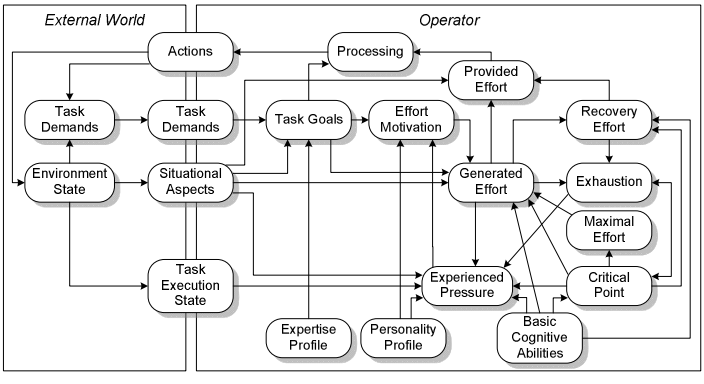


Fig. 2. Overview of the Functional State model

The most important variable from the FS model that is used in this paper is the *experienced pressure*. Here, this variable is used to determine the amount of available attention (the precise relation is explained in the next section). In the FS model, experienced pressure is related to a number of factors, such as the amount of exhaustion, the amount of effort related to the critical point and the performance quality. The strength of these relations is dependent on personality characteristics like exhaustion sensitivity, performance norm and performance sensitivity.

2.3 Integrating the Attention and Functional State Model

One of the drawbacks of the *attention-* model is that it does not take into account that the amount of attention may vary over time. However, in reality, this amount of attention is influenced by different aspects of the functional state, in particular by the experienced pressure. Experienced pressure results in variances in concentration and motivation, which are directly related to attention. This is the idea behind the integration of the previously explained submodels: the output variable experienced pressure of the Functional State submodel is used as an input variable for the total amount of attention $A(t)$ in the Attention submodel. This is done in the following way:

$$A(t) = a + b \cdot (1 - EP(t))$$

where $EP(t)$ is the experienced pressure at time point t, and $0 \leq a \leq 1$ and $-1 \leq b \leq 1$ are parameters that can be tuned. If the used Functional State submodel is valid, this means that also the *attention+* model will have an improved validity, due to the FS's capability to alter $EP(t)$ at the appropriate times and therefore dropping or taking into account the part of the estimation where *attention-* was the least certain of.

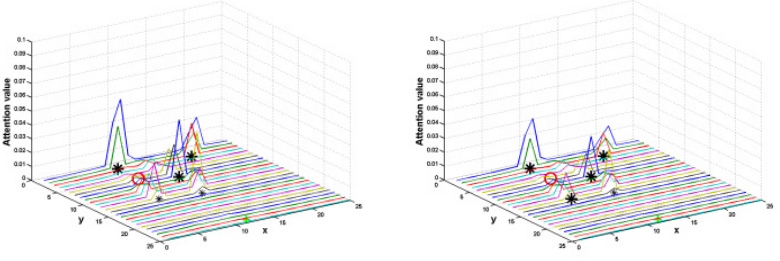


Fig. 3. a. Output *attention-* (left) and *attention+* (right) with a low situational demand

In Figure 3a and b an example is given of this capability. A visualisation of the outcomes of the models *attention-* and *attention+* is shown for the situations in which there is a low situational demand (a) and a high situational demand (b). Here, the area determined by the x- and y-axis represents a radar screen, the circles denote contacts, and the z-axis indicates the estimated level of attention. The situational demand is related to the amount of contacts to be handled with a certain time interval.

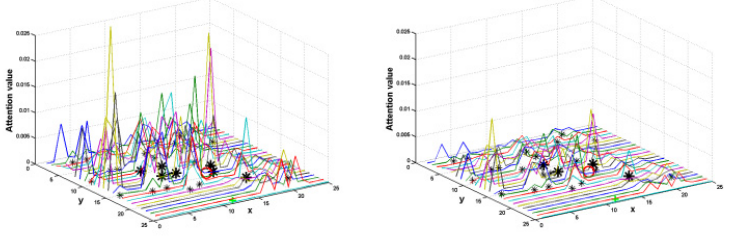


Fig. 3. b. Output *attention-* (left) and *attention+* (right) with a high situational demand

For fixed decision criteria *attention+* is able to adapt to the expected change of the functional state of the user, whereas *attention-* is not. In Figure 3 this means that more objects are estimated to be attended to in the low situational demand (4 opposed to 3) and less in the high situational demand condition (4 opposed to 7).

3 Experiment

The goal of the experiment was to investigate the difference in validity between the attention model connected to the FS model (*attention+*) and the original attention model (*attention-*). The hypothesis is that the validity of the *attention+* model is higher than the validity of the *attention-* model.

3.1 Participants

Three female and two male participants with a mean age of 24.67 took part in this study. All participants already had some experience with the task environment.

3.2 Simulation-Based Training Environment

The main task that was used in this study consists of identifying incoming contacts and, based on the outcome of identification, deciding to eliminate the contact (by shooting) or allowing it to land (by not shooting). A screenshot of this simulation-based training environment is displayed in Figure 4. The object at the bottom of the screen represents the participant's (stationary) weapon. In addition, contacts (allies and enemies in the shape of a dot with a radius of 5 pixels) appear at a random location on the top and fall down to random locations at the bottom of the screen.

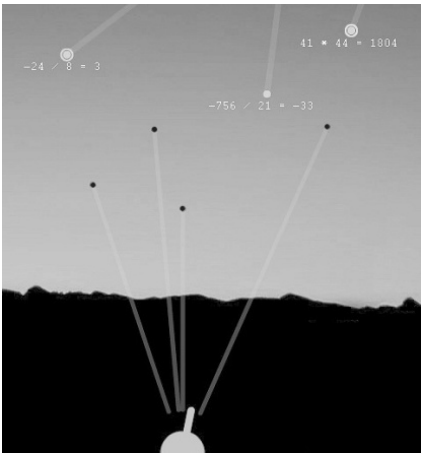


Fig. 4. Screenshot of the Task Environment

Before a contact can be identified, it has to be perceived. This is done by a mouse click at the contact, which reveals a mathematical equation underneath the contact. The identification task is to check the correctness of the mathematical equation (which is less difficult in less demanding situations). A correct equation means that the contact is an ally; an incorrect equation indicates that the contact is an enemy. Identification is done by pressing either the left or right arrow for respectively an ally or enemy. When a contact is identified a green (for an ally) or a red (for an enemy) circle appears around the contact.

The contacts that are identified as an enemy have to be shot before they land. A missile is shot by executing a mouse click at a specific location; the missile will move from the weapon to that location and explode exactly at the location of the mouse click. Any contact within a radius of 50 pixels of the exploding missile is destroyed.

3.3 Procedure

The experiment consisted of 4 blocks of 20 minutes of the simulation-based task environment. In the first 10 minutes of one block, task demands were low (contacts appear

every 10 to 20 seconds) and in the second 10 minutes of one block, task demands were high (contacts appear every 2.25 to 4.5 seconds). In the first and third block, 'freezes' were made after each 2.5 minute, in the second and fourth block no freezes were made. When a freeze was made, the experiment was put on hold and the following sentence was shown: "Gameplay frozen. Select contacts, press space when done." At this moment, participants had to select all contacts which they thought to have recently paid attention to. After selection, a computer version of the NASA-TLX was shown, where participants had to indicate their performance and mental effort.

First, a Tobii x50 eye tracker (http://www.tobii.se) was connected to measure eye movements as input for the attention model. After calibration of the eye tracker, the experiment was started and onscreen instructions were given on the task environment and freezes. The instructions were followed by a practice block of 1 minute low task demands and 1 minute high task demands to get familiar with the environment. After practice, participants started with the first block. Before each block the eye tracker status was checked and after each block, the participant was given a three minute break before continuing with the next block.

When the participant finished the experiment, the data on the task difficulty (situational demands) and performance quality were used for tuning of the parameters in the FS model. This was done using a Simulated Annealing technique. This method initially selects a random parameter setting as the best available parameter setting, then introduces a small change in these settings to generate a neighbour of the current parameter settings in the search space. If this neighbour is found to be a more appropriate representation of the observed human behaviour then it is marked as the best known parameter setting, otherwise a new neighbour is selected to evaluate its appropriateness. For more details about how this technique was applied, see [2].

Next, the obtained personal parameters were used to calculate the experienced pressure and the exhaustion in the FS model, which served as input for predicting attention in the $attention_+$ model. Furthermore, both eye movements and features of contacts (luminance, colour, ...) served as input for the attention model. At this point, also the remaining parameters of the attention model (i.e., λ, α, and the different w_M, see Section 2.1) and of the connection between both models (i.e., a and b, see Section 2.3) are tuned using Simulated Annealing, to obtain an optimal performance.

3.4 Data Analysis

The output of the *attention-* and *attention+* models have been compared with subjective data retrieved during freezes in the experiment. In 40 minutes, after each 2.5 minutes of the task execution time a freeze was initiated, where the participant was asked to point out to what objects she was paying attention to. At the same time, *attention-* and *attention+* also pointed out what they thought was the case. Each freeze in an easy condition was coupled with one in a hard condition in order to be able to evaluate the performance of the models given that the task demand changes over time (see Table 1).

Table 1. Freeze couples

Freeze couple nr	Freeze nr (easy)	Freeze nr (hard)
1	1	5
2	2	6
3	3	7
4	4	8
5	9	13
6	10	14
7	11	15
8	12	16

The procedure used to compare the models with the subjective data retrieved during the freezes is described in the next section.

4 Results

This section presents the results of the experiment. An example of the experienced pressure as predicted from the functional state model is shown in Figure 5, for participant 1 and 2. These two participants were quite extreme cases, in the sense that participant 1 experienced much pressure, and participant 2 experienced little pressure. As can be seen, both participants experience less pressure during the blocks with low task demands (time point 0-1500, 3000-4500, and 6000-7500) than during the blocks with high task demands. Recall that these fluctuations of experience pressure were used to determine the values of the total amount of attention *A(t)* in the *attention+* model. Unfortunately, results of participant 5 could not be used, as the functional state model provided unreliable data.

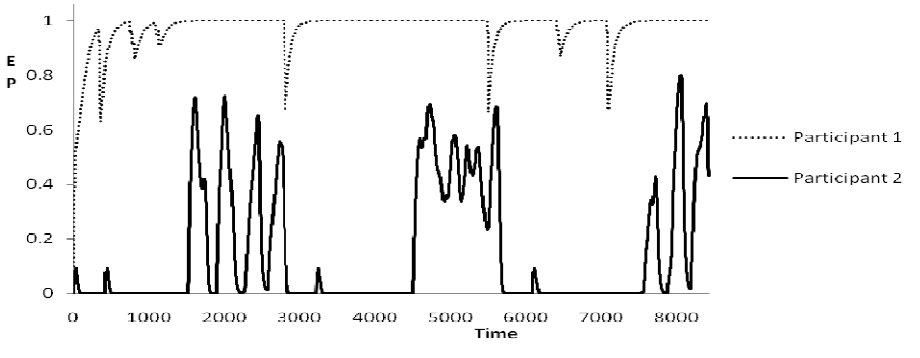


Fig. 5. Estimated Experienced Pressure over time of participant 1 and 2 during three blocks

To evaluate the models, a performance measure was chosen based on the calculation of true positives (hit rate) and false positives (false alarm rate), which can be extracted from confusion matrices, as is shown in Table 2.

Table 2. Confusion matrix

		Participant		
		t	f	$total$
Model	t'	Hits	False Alarms	T'
	f'	Misses	Correct Rejections	F'
	$total$	T	F	

In this table, *t* and *f* represent whether the participant indicated that he allocated attention to an object or not, respectively, and *t'* and *f'* indicate that one of the models indicated it or not, respectively. Moreover, *Hits/T* results in the hit rate and *False Alarms/F* results in the false alarm rate. Based on these notions, the *sensitivity score d'*, which is a measure for the performance of the model, is determined as follows:

$$d' = zscore(H) - zscore(FA)$$

where *zscore(X)* is a function that represents *X* in terms of standard deviations from the average.

The results of the experiment show that the average performance of the *attention+* model ($M+ = 0.736$, $SD+ = 0.118$) was significantly higher than for the *attention-* model ($M- = 0.661$, $SD- = 0.128$), with $t(31) = 4.709$, $p < 0.001$ (paired t-test). Average d-primes per participants are displayed in Figure 6.

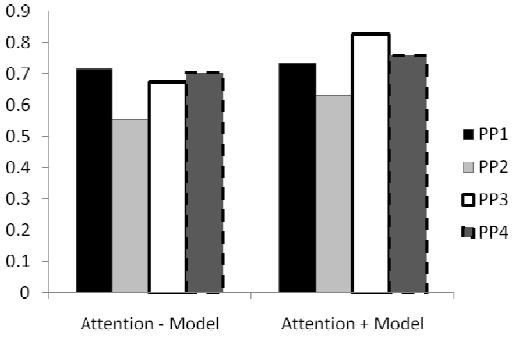


Fig. 6. Mean d-prime per model averaged over participants.

5 Discussion

For personalized ambient agents supporting persons performing demanding tasks, it is important that they have insight in various aspects of the person's mental state [5], [14]. One important aspect is the distribution of the person's attention over the objects (s)he observes. However, existing models for human attention (e.g., [3]) assume that the total amount of attention a person can spend is static. As a result, such models cannot distinguish situations in which a person experiences a lot of pressure from situations in which (s)he is completely relaxed, whereas in reality these situations result in very different behaviours.

To solve this problem, the contribution of the current paper was to extend the original attention model from [3] with a component to keep track of a person's functional state. A first experiment provided evidence that the validity of the *attention+* model was slightly higher than the validity of the *attention-* model.

Despite this encouraging result, the limitations of the approach should not be ignored. First, the amount of participants in the experiment (only 5) was too low to be able to draw strict conclusions. Second, the results were difficult to evaluate, due to a number of complicating factors. For example, the presence of the 'freezes' used for the subjective evaluation may have interfered with the task. Third, the approach assumes that participants are sufficiently capable of estimating where their own attention is. Although some evidence exists that this is indeed the case [13], this assumption can be tested more precisely. Finally, it is an open question to what extent the results can be generalised to other scenarios and circumstances.

In future work, it is planned to address these concerns. For example, experiments with higher numbers of participants are planned, both with the current setup and within a different experimental context. In addition, more work will be spent on fine-tuning of the parameters involved in the model (using standard techniques such as simulated annealing and gradient-based parameter estimation). Finally, on the long term, it is planned to actually implement an intelligent ambient agent system supporting humans in demanding circumstances, and test this in more realistic scenarios.

Acknowledgement. The authors wish to thank Rogier Oorburg and Michael de Vos for their contribution to the training environment, and Alexei Sharpanskykh and S. Waqar Jaffry for their contribution to the tuning of the parameters of the model.

References

1. Aarts, E., Harwig, R., Schuurmans, M.: Ambient Intelligence. In: Denning, P. (ed.) The Invisible Future, pp. 235–250. McGraw Hill, New York (2001)
2. Bosse, T., Both, F., Hoogendoorn, M., Jaffry, S.W., Lambalgen, R., van Oorburg, R., Sharpanskykh, R., Treur, J., De Vos, M.: Design and Validation of a Model for a Human's Functional State and Performance. International Journal of Modeling, Simulation, and Scientific Computing 2, 1–31 (2011)
3. Bosse, T., van Maanen, P.-P., Treur, J.: Simulation and Formal Analysis of Visual Attention. Web Intelligence and Agent Systems Journal 7, 89–105 (2009)
4. Bosse, T., van Lambalgen, R., van Maanen, P.P., Treur, J.: A System to Support Attention Allocation: Development and Application. Web Intelligence and Agent Systems Journal (in press, 2012)
5. Endsley, M.: Toward a Theory of Situation Awareness in Dynamic Systems. Human Factors 37, 32–64 (1995)
6. Gonzalez, C.: Task workload and cognitive abilities in dynamic decision making. Human Factors 47, 92–101 (2005)
7. Green, D.M., Swets, J.A.: Signal Detection Theory and Psychophysics. Wiley, New York (1966)

8. Hancock, P.A., Williams, G., Manning, C.P., Miyake, S.: Influence of task demand characteristics on workload and performance. The International Journal of Aviation Psychology 5, 63–86 (1995)
9. Hill, D.W.: The critical power concept. Sports Medicine 16, 237–254 (1993)
10. Hockey, G.R.J.: Compensatory control in the regulation of human perfromance under stress and high workload: a cognitive-energetical framework. Biological Psychology 45, 73–93 (1997)
11. Hockey, G.R.J.: Operator Functional State as a framework for the assessment of performance degradation. In: Hockey, G.R.J., Gaillard, A.W.K., Burov, O. (eds.) Operator Functional State, pp. 8–23. IOS Press, Amsterdam (2003)
12. Itti, L., Koch, C.: Computational Modeling of Visual Attention. Nature Reviews Neuroscience 2, 194–203 (2001)
13. van Maanen, P.-P., de Koning, L., van Dongen, K.: Design and Validation of HABTA: Human Attention-Based Task Allocator. In: Mühlhäuser, M., Ferscha, A., Aitenbichler, E. (eds.) AmI 2007 Workshops. CCIS, vol. 11, pp. 286–300. Springer, Heidelberg (2008)
14. Wilson, G.F., Russell, C.A.: Performance enhancement in an uninhabited air vehicle task using psychophysiologically determined adaptive aiding. Human Factors 49, 1005–1018 (2007)

Tracking and Constraining Authorization Provenance

Jinwei Hu[1], Khaled M. Khan[1], Yun Bai[2], and Yan Zhang[2]

[1] Department of Computer Science and Engineering, Qatar University, Qatar
[2] School of Computing and Mathematics University of Western Sydney, Australia
{jinwei,k.khan}@qu.edu.qa, {ybai,yan}@scm.uws.edu.au

Abstract. Authorization provenance concerns how an authorization is derived. It appears important to define authorization provenance to (1) analyze policy bases, (2) defend against a class of attacks, and (3) audit authorizations. In this paper, we study a notion of authorization provenance, based on a recently proposed logic in the literature. By examining a collection of properties, we show this definition captures the intuitions of authorization provenance. We also present an application of our notion of authorization provenance: specifying and enforcing a new type of security requirements.

1 Introduction

Authorization provenance is information about how an authorization is derived. Modelling authorization provenance is challenging in decentralized environments, as mechanisms like delegation make authorization provenance complex. Besides the resource guard who controls access to resources, other agents (e.g., delegatees) play a role in authorization decision-making as well.

Suppose for example that Alice is the warden of a building and that the request to access the building would be allowed only if "Alice believes $access$" can be proved. Consider the following cases:

CASE1	Alice believes $access$
CASE2	Alice trusts Bob on $access$ Bob trusts Cathy on $access$ Cathy believes $access$

Observe that Alice's belief in $access$ is concluded in different ways. In CASE1, it is because Alice herself, whereas Bob and Cathy also have an effect in CASE2.

Among others, provenance information helps enforce and analyze security. Putting restrictions on authorization provenance may prevent insiders' misuse of their privileges. Suppose that the management board of the building is composed of Alice and Bob, and that it is required that whether or not to allow access be determined only by the board members. In this case, in order to enter the building, one has to prove "due to Alice and Bob, Alice believes $access$" but not simply "Alice believes $access$". In CASE2, Cathy could not obtain the access, because her statement is indispensable to the conclusion that "Alice believes $access$". Hence, the delegation from Bob to Cathy is actually ignored, thus preventing Bob's misuse and neglect. As also pointed out in

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 669–678, 2012.

[6,13], host security may be compromised if provenance is not taken into account when making authorization decisions.

In the literature, a logic DBT (Due to, Belief and Trust) is designed to represent belief, trust, provenance, and their relations [5]. DBT enables explicit representation of authorization provenance. This work follows the lines of [5].

In this paper, we attempt to track and constrain authorization provenance with respect to logic-based policy bases. Based on DBT, we define two forms of authorization provenance: *simple provenance* and *nested provenance*. We study their properties and thus show that the definition captures important intuitions of authorization provenance (Section 3). We present an example application of the definition of provenance: specification and enforcement of constraints on authorization provenance (Section 4). These constraints can model novel security requirements.

2 Background

We recall the syntax and the semantics of DBT. Consider a countable set of agents AG. DBT has three types of modal operators for each agent i: $\mathtt{B}_i$, $\mathtt{T}^i_j$, and $\mathtt{D}_i$. $\mathtt{B}_i\varphi$ means that agent i *believes* φ. $\mathtt{T}^i_j\varphi$ reads that agent i *trusts* agent j on φ. $\mathtt{D}_i\varphi$ means that "*due to* agent i, φ holds". A subset AE of AG is called an *agent expression*. Given an $AE \subseteq AG$, there is also an operator $\mathtt{D}_{AE}$ based on $\mathtt{D}_i$ for each $i \in AE$. $\mathtt{D}_{AE}\varphi$ means that, due to the set AE of agents together, φ holds. Let Prop be a set of primitive propositions. Given $p \in$ Prop, DBT formulas are inductively defined:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \Rightarrow \varphi \mid \mathtt{B}_i\varphi \mid \mathtt{D}_i\varphi \mid \mathtt{D}_{AE}\varphi \mid \mathtt{T}^i_j\varphi.$$

The semantics of DBT formulas is defined based on *Kripke models*. A Kripke model $\mathcal{M}$ is a tuple $\langle W, \pi, \mathcal{B}_i, \mathcal{D}_i, \mathcal{T}^i_j\rangle$ $(i, j \in AG; i \neq j)$, where W is a set of states, $\pi : W \mapsto 2^{\text{Prop}}$ is a labeling function which maps each state to a subset P of Prop such that, in this state, any $p \in P$ is true and any $p \in$ Prop$\backslash P$ is false, $\mathcal{B}_i \subseteq W \times W$ is a serial, transitive and Euclidean binary relation on W, $\mathcal{D}_i \subseteq W \times W$ is a binary relation on W, and $\mathcal{T}^i_j \subseteq W \times 2^W$ is a binary relation between W and its power set.

Definition 1. *($\models$) Given a model $\mathcal{M} = \langle W, \pi, \mathcal{B}_i, \mathcal{D}_i, \mathcal{T}^i_j\rangle$, $w \in W$, and a formula φ, let $\mathcal{D}_{AE} = \bigcap_{i\in AE} \mathcal{D}_i$. We define the satisfaction relation $\models$ as follows: (1) $\langle \mathcal{M}, w\rangle \models p$ iff $p \in \pi(w)$, (2) $\langle \mathcal{M}, w\rangle \models \neg\varphi$ iff $\langle \mathcal{M}, w\rangle \not\models \varphi$, (3) $\langle \mathcal{M}, w\rangle \models \varphi_1 \wedge \varphi_2$ iff $\langle \mathcal{M}, w\rangle \models \varphi_1$ and $\langle \mathcal{M}, w\rangle \models \varphi_2$, (4) $\langle \mathcal{M}, w\rangle \models \varphi_1 \Rightarrow \varphi_2$ iff $\langle \mathcal{M}, w\rangle \not\models \varphi_1$, or $\langle \mathcal{M}, w\rangle \models \varphi_2$, (5) $\langle \mathcal{M}, w\rangle \models \mathtt{B}_i\varphi$ iff $\langle \mathcal{M}, v\rangle \models \varphi$ for all v such that $(w, v) \in \mathcal{B}_i$, (6) $\langle \mathcal{M}, w\rangle \models \mathtt{D}_i\varphi$ iff $\langle \mathcal{M}, v\rangle \models \varphi$ for all v such that $(w, v) \in \mathcal{D}_i$, (7) $\langle \mathcal{M}, w\rangle \models \mathtt{D}_{AE}\varphi$ iff $\langle \mathcal{M}, v\rangle \models \varphi$ for all v such that $(w, v) \in \mathcal{D}_{AE}$, and (8) $\langle \mathcal{M}, w\rangle \models \mathtt{T}^i_j\varphi$ iff $(w, \lceil\varphi\rceil) \in \mathcal{T}^i_j$, where $\lceil\varphi\rceil = \{v \in W \mid \langle \mathcal{M}, v\rangle \models \varphi\}$.*

We define a decentralized authorization system as a tuple $\langle AG, LPB, L, Q, M\rangle$. AG is the set of agents involved in the system. For example, an agent could be a resource-requester, a process running one behalf of a user, and an organization (e.g., a university). We assume that AG includes a member agent called *resource guard*, denoted as G. The guard, as the root of trust, makes authorization decisions.

Credentials are stored in a decentralized way. Each agent maintains a local policy base, which may store credentials that the agent signed and credentials that concern the agent. LPB maps each agent in AG to a set of credentials. We refer to the set of credentials that an agent i maintains as i's *local policy base*. For instance; Alice's local policy base includes a credential that Alice trusts Bob on $access$.

L is the authorization logic used to represent the credentials. In this paper L is DBT. We assume that credentials only encode agents' beliefs and trust. For example, we write $\mathtt{T}^{Alice}_{Bob}\,access$ for the above-mentioned credential. Q is a set of queries of the form $\mathtt{D}_{AE_n}\cdots\mathtt{D}_{AE_1}\mathtt{B}_i p$. A query usually asks whether an access is allowed. A query for the building-access example is $\mathtt{D}_{\{Alice,Bob\}}\mathtt{B}_{Alice}\,access$.

M is a mechanism searching for proofs of queries. We assume M is the proof-carrying authorization mechanism [1]. In this case, the resource guard would not permit access unless it is provided with a valid proof which concludes the access is entailed by the policy. The proof is usually submitted by the agent who requests the access. We assume that M is able to access local policy bases to compose proofs. In this case, the union of local policy bases could be treated as one monolithic policy base, thus eliminating the needs of LPB and M. In other words, we abstract the decentralized authorization system as $\langle AG, \text{DBT}, Q, \text{PB}\rangle$, where PB is the monolithic policy base. Formally, a policy base PB is a finite set of $\mathtt{WFF}_{pa}$ formulas defined as below.

$$\phi ::= \mathtt{B}_i p \mid \mathtt{T}^i_j p$$

3 Defining Authorization Provenance

In this section, we examine a simple form of authorization provenance, which is abstracted as the set of agents whose statements are referenced in the deduction of the authorization. The intended function of $\mathtt{D}_{AE}$ is to record the agents who affect agents' beliefs. It appears plausible to define provenance in terms of $\mathtt{D}_{AE}$.

Definition 2 (Authorization Provenance). *Given PB and a query* $q = \mathtt{D}_{AE_n}\cdots\mathtt{D}_{AE_1}\mathtt{B}_i p$ *such that PB* $\models q$*, we say* $\langle AE_n, \cdots, AE_1\rangle$ *is the* provenance *of* $\mathtt{B}_i p$ *in PB, denoted as* $\mathsf{prov}_{PB}[\mathtt{B}_i p]$*. We say* $\mathsf{prov}_{PB}[\mathtt{B}_i p]$ *is a* simple provenance *if* $\mathsf{prov}_{PB}[\mathtt{B}_i p] = \langle AE\rangle$*, and a* nested provenance *if* $\mathsf{prov}_{PB}[\mathtt{B}_i p] = \langle AE_n, \cdots, AE_1\rangle$*, where* $n > 1$.

Simple provenance is useful when defending against attacks that utilize systems' neglect of privilege source [6,13]. It has been shown that simply tracking provenance as a set of agents is the key to defend against trojan horses in discretionary access control [6]. Also, the security problem of delegation can be resolved by monitoring delegation's provenance [13]. We now show some properties of simple provenance; nested provenance will be discussed in Section 4.

Distinguishability. Suppose that there is more than one way to conclude φ: in addition to AE, AE' could be the provenance of φ. One should be able to express and query φ with these two provenance (i.e., $\mathtt{D}_{AE}\varphi$ and $\mathtt{D}_{AE'}\varphi$) separately. Provenance distinguishability is the basic motivation and design objective of DBT. With the operator $\mathtt{D}_{AE}$, one is able to express and query provenance.

Traceability. Any agent contributing to the derivation of φ, should be included in the simple provenance of φ. Re-delegations and attribute-based delegations can easily result in nested provenance of a belief. With this property, we are able to collapse the accumulated provenance into a simple one - a set of agents, no matter how complicated the derivation is. Proposition 1 implies that we can keep provenance concise and simple.

Proposition 1. $\models D_{AE_n} \cdots D_{AE_1} \varphi \Rightarrow D_{AE_1 \cup \cdots \cup AE_n} \varphi$.

With simple provenance, the following property shows that provenance is correctly recorded during the applications of delegations.

Proposition 2. *1.* $\models D_{AE_1} T^i_j \varphi \wedge D_{AE_2} B_j \varphi \Rightarrow D_{AE_1 \cup AE_2 \cup \{j\}} B_i \varphi$.
2. $\models D_{AE_1} T^i_j \varphi \wedge D_{AE_2} T^j_k \varphi \Rightarrow D_{AE_1 \cup AE_2 \cup \{j\}} T^i_k \varphi$.

Proposition 2 shows that, even though delegations and beliefs come along with their own provenance, these provenance would be recorded and merged with new ones in the event of delegations taking effect.

Cooperation. Agents may cooperate to finish a task. If a task ψ is divided into n subtasks $\varphi_1, \cdots, \varphi_n$, then the union of agent sets, each of which finishes one sub-task, are responsible for the original task.

Proposition 3. $\models (\varphi_1 \wedge \cdots \wedge \varphi_n \Rightarrow \psi) \wedge (D_{AE_1} \varphi_1 \wedge \cdots \wedge D_{AE_n} \varphi_n) \Rightarrow D_{AE_1 \cup \cdots \cup AE_n} \psi$.

Example 1. Suppose a warehouse $task$ is composed of four steps: $prepare$, $payment$, $issue$, and $invoice$. That is, we have $(prepare \wedge payment \wedge issue \wedge invoice) \Rightarrow task$. Suppose further that each of the agents A, B, C, and D accomplish each of the steps, respectively. That is, we also have $\mathtt{D}_A prepare \wedge \mathtt{D}_B payment \wedge \mathtt{D}_C issue \wedge \mathtt{D}_D invoice$. According to Proposition 3, it holds that $\mathtt{D}_{\{A,B,C,D\}} task$; namely, we know it is because $\{A, B, C, D\}$ that the task is done.

Transferability. Provenance has a flavor of responsibility in certain cases. Responsibility may be transferred from one agent to another.

Proposition 4. *If* $\models D_{AE_1 \cup AE_2} \varphi$ *and* $\models D_{AE_2} \varphi \Rightarrow D_{AE_3} \varphi$, *then* $\models D_{AE_1 \cup AE_3} \varphi$.

Suppose that a software manufacturer Mf releases a software called Sw, and that A installs Sw on her computer. For some reasons, Sw automatically executes some malicious script downloaded from a web-site Wb and the script did some damages to her computer (e.g., files being destroyed). Naturally, the responsibility of Sw could be transferred to Mf, as Mf is the producer of Sw. Then, besides Wb, A may complain about Mf. Put formally, from $\models \mathtt{D}_{\{Wb,Sw\}} damage$ and $\models \mathtt{D}_{Sw} damage \Rightarrow \mathtt{D}_{Mf} damage$, we have $\models \mathtt{D}_{\{Wb,Mf\}} damage$ by the transferability.

Proposition 4 implies that the responsibility on the set AE_2 of agents can be transferred to the agent set AE_3. This kind of transferability often happens between agents who are related with each other, like "be a supervisor of".

Limited Responsibility. As mentioned before, provenance can be used to trace responsibility: If i delegates the judgement of φ to j, then when j utters her belief in φ, j is the provenance of, and is also responsible for, i's belief in φ. Note that the responsibility is assigned from i's viewpoint. However, j may not speak of φ directly. Suppose that i delegates only the judgement of φ to j, but that j utters her belief in ψ, which is more informative than φ. In this case, j is only responsible for i's belief in φ. In other words, j's responsibility is limited to $\mathtt{B}_i\varphi$.

Proposition 5. *If* $\models \psi \Rightarrow \varphi$, *then* $\models T_j^i\varphi \wedge B_j\psi \Rightarrow D_jB_i\varphi$.

4 Constraints on Authorization Provenance

While simple provenance suffices in some cases, more informative provenance comes in handy. [5] presents a class of provenance-aware queries with nested provenance; one can syntactically extract useful information about the delegation in PB from a query's provenance, if the query is entailed by the PB. A query $q = \mathtt{D}_{AE_n} \cdots \mathtt{D}_{AE_1}\mathtt{B}_i p$ is *provenance-aware*, if $AE_n = \{i_n, \cdots, i_1\}$, $AE_t = AE_n \backslash \{i_n, \cdots, i_{t+1}\}$ $(n-1 \leq t \leq 1)$, and $i \notin AE_n$, where $\{i_n, \cdots, i_1\} \subseteq AG$. We utilize the following theorem to enforce constraints on authorization provenance.

Theorem 1 ([5]). *If PB* $\models D_{\{i_l,\cdots,i_l\}} \cdots D_{\{i_2,i_1\}} D_{i_1} B_i p$, *there is a delegation of* p *from* i *to* i_l, *from* i_l *to* i_{l-1}, $\cdots$, *and finally from* i_2 *to* i_1.

4.1 Definition

Several attacks are found related to authorization provenance. For example, Wang et al. [13] point out users may abuse delegations to circumvent security policies (in particular, *separation of duty* policies).

Example 2. Consider a task of issuing checks [13]: In a company, the task of issuing checks is modeled by two authorizations pre and app, which stand for "prepare check" and "approve check", respectively. In order to prevent fraudulent transactions, a separation of duty policy $sod\langle pre, app\rangle$ that pre and app must be performed by two *different* treasurers is required. Also, for the sake of resiliency, the company allows a treasurer to delegate her authority to a clerk in case she is not able to work.

Suppose that a treasurer, A, and a clerk of the company, B, decide to collude to issue checks for themselves. They can accomplish in three steps: A delegates the authority pre to B; B performs pre to prepare a check for A; and A performs app to approve the check prepared by B.

The observation is that both requests to pre and app are (in part) from A. Similarly, A can create dummy agents (e.g., processes) or manipulate behind the scene (e.g., trojan horses) [6]. Unfortunately, systems that employ logic-based policy bases are also vulnerable to these attacks. The defense mechanism proposed in [13] addresses the problem in work-flow systems; it is unclear how to adapt that approach in face of logic-based policies. Moreover, in spite of the importance of separation of duty policies, other compromises involving provenance deserve investigation in their own right.

Fig. 1. The BP-NFA γ_A

The attacks are mainly caused by the ignorance of how an authorization is derived. This implies that we may resolve them by putting constraints on authorization provenance. We follow the convention to model constraints as finite state automata [9]. We work on non-deterministic finite automata (NFA). Let the alphabet of the automaton be $\Sigma = AG \cup \mathtt{WFF}_{pa} \cup \{\epsilon\}$. An NFA γ is a tuple $(V, \Sigma, \delta, v_0, F)$, where V is a finite set of states, Σ is an alphabet, $\delta : V \times \Sigma \mapsto 2^V$ is a transition function, $v_0 \in V$ is a start state, and $F \subseteq V$ is a set of accept states [10].

Definition 3 (Basic Provenance NFA (BP-NFA)). *A* basic provenance NFA *is an NFA* $(V, \Sigma, \delta, v_0, F)$ *that meets the two conditions: (1)* $|F| = 1$*, and (2) for any state* $v \in V$ *and any* $\sigma \in \Sigma$*,* $\delta(v, \sigma) \subseteq F$ *if and only if* σ *is of the form* $\mathtt{B}_i p$*. Suppose that* $F = \delta(v, \phi)$*; we say that* γ *ends with* ϕ*, denoted as* $\mathsf{end}_\gamma[\phi]$.

The intuition of the two conditions is that each BP-NFA recognizes only one query at a time. Given a provenance aware query $q = \mathtt{D}_{AE_n} \cdots \mathtt{D}_{AE_1} \mathtt{B}_i p$, we define a string over Σ as $s_n s_{n-1} \cdots s_1 s_0$, where, for $1 \leq t \leq n$, $s_t = AE_t \backslash AE_{t-1}$ and $s_0 = \mathtt{B}_i p$; denote the string as $str[q]$. We say a BP-NFA γ *recognizes* q if γ accepts $str[q]$.

Take the BP-NFA γ_A in Fig. 1 for example. Consider the access to A's files, which is authorized if one can prove $\mathtt{D}_{AG}\mathtt{B}_G access(files, A)$. γ_A ends at $\mathtt{B}_G access(files, A)$. γ_A recognizes $\mathtt{D}_{\{A,C\}}\mathtt{D}_C\mathtt{B}_G access(files, A)$ but not $\mathtt{D}_{\{B,A,C\}}\mathtt{D}_{\{A,C\}}\mathtt{D}_C\mathtt{B}_G access(files, A)$, for the string of the latter does not start with A.

Some constraints are put on a combination of authorization provenance but not a single one. For example, the separation of duty policy $sod\langle pre, app\rangle$ forbids any user from executing both pre and app.

Definition 4 (Concatenated Provenance NFA (CP-NFA)). *Given two BP-NFA,* $\gamma_1 = (V_1, \Sigma, \delta_1, v_{0,1}, F_1)$ *and* $\gamma_2 = (V_2, \Sigma, \delta_2, v_{0,2}, F_2)$*, define the concatenation of* γ_1 *and* γ_2 *as* $\gamma_1 \circ \gamma_2 = (V, \Sigma, \delta, v_0, F)$ *such that* $V = V_1 \cup V_2$*,* $v_0 = v_{0,1}$*,* $F = F_2$*, and*

$$\delta(v, \sigma) = \begin{cases} \delta_1(v, \sigma) & \text{if } v \in V_1 \text{ and } v \notin F_1 \\ \delta_1(v, \sigma) & \text{if } v \in F_1 \text{ and } \sigma \neq \epsilon \\ \delta_1(v, \sigma) \cup \{v_{0,2}\} & \text{if } v \in F_1 \text{ and } \sigma = \epsilon \\ \delta_2(v, \sigma) & \text{if } v \in F_2. \end{cases}$$

CP-NFA are used to recognize a set of queries. Denote the concatenation of two strings str_1 and str_2 as $str_1 \diamond str_2$. Given a set Q of provenance aware queries and a CP-NFA γ, we say γ *recognizes* Q if $Q = \{q_1, \cdots, q_n\}$ and there exists a sequence $\langle q_1, \cdots, q_n\rangle$ such that γ accepts the string $str[q_1] \diamond \cdots \diamond str[q_n]$.

Suppose that the request to pre (respectively, app) is accompanied with a proof whose conclusion is $\mathtt{D}_{AE_n} \cdots \mathtt{D}_{AE_1} \mathtt{B}_G pre$ (respectively, $\mathtt{D}_{AE'_n} \cdots \mathtt{D}_{AE'_1} \mathtt{B}_G app$). Let X

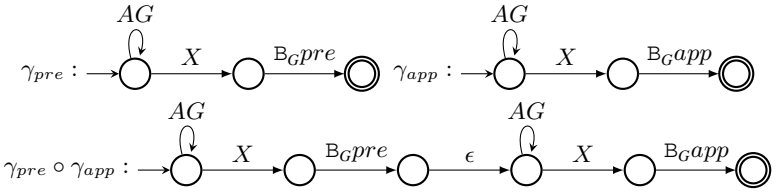


Fig. 2. $\gamma_{sod} = \gamma_{pre} \circ \gamma_{app}$

be a meta-variable over AG. The CP-NFA γ_{sod} in Fig. 2 recognizes a combination of requests of pre and app which are both issued by X, regardless of intermediate delegation. For instance, γ_{sod} recognizes $\{\mathtt{D}_{\{A,X\}}\mathtt{D}_X\mathtt{B}_G pre, \mathtt{D}_{\{B,C,X\}}\mathtt{D}_{\{C,X\}}\mathtt{D}_X\mathtt{B}_G app\}$.

Definition 5 (Constraints on Authorization Provenance). *A constraint on authorization provenance is a tuple $\langle \gamma, sign \rangle$, where γ is either a BP-NFA or a CP-NFA and $sign \in \{+, -\}$. Given a set Q of provenance aware queries, we say Q* satisfies *$\langle \gamma, sign \rangle$ if γ recognizes Q when $sign = +$, and γ does not recognize Q when $sign = -$. Given a set C of constraints on provenance, we say Q satisfies C if for all $c \in C$, Q satisfies c.*

A constraint $\langle \gamma, + \rangle$ requires that the provenance of the involved authorizations matches the pattern specified by γ, whereas $\langle \gamma, - \rangle$ means that the provenance must not be recognized by γ.

4.2 Example Constraints

Discretionary access control safety. We can express discretionary access control safety by a machine which accepts the provenance starting with the owner of the object. Consider again the BP-NFA γ_A in Fig. 1. Then the constraint $\langle \gamma_A, + \rangle$ requires the provenance of $\mathtt{B}_G access(files, A)$ starts with A; that is, every access to A's files originates from A or her delegation.

Group-related constraints. We restrict an authorization to a group of users by $\langle \gamma_{group}, + \rangle$, where γ_{group} is shown in Fig. 3. The BP-NFA γ_{group} specifies that only if the access is directly requested by $group$ members (in the sense that a member is at the end of the provenance) would it be allowed.

Fig. 3. Example BP-NFA related to groups

Also, blacklist can be enforced via a constraint $\langle \gamma_{blacklist}, - \rangle$. By this constraint, any authorization with provenance beginning with a blacklisted user would be declined. Traditional approaches to blacklisting may fall short in face of delegations, for they do not take authorization provenance into account.

Separation of duty. With γ_{sod} in Fig. 2, $\langle \gamma_{sod}, - \rangle$ enforces $sod\langle pre, app \rangle$ in the traditional sense. However, it fails to enforce $sod\langle pre, app \rangle$ in the presence of delegation. For example, $\{\mathtt{D}_{\{A,B\}}\mathtt{D}_B\mathtt{B}_G pre, \mathtt{D}_A\mathtt{B}_G app\}$ satisfies $\langle \gamma_{sod}, - \rangle$, but does not complies with $sod\langle pre, app \rangle$. With the BP-NFA $\gamma_{gen-pre}$ and $\gamma_{gen-app}$ in Fig. 4, the CP-NFA $\gamma_{gen-sod} = \gamma_{gen-pre} \circ \gamma_{gen-app}$ checks if the requests of pre and app are both from X, either directly or indirectly via delegation. The constraint $\langle \gamma_{gen-sod}, - \rangle$ prevents users from circumventing $sod\langle pre, app \rangle$ with the help of delegation.

Fig. 4. $\gamma_{gen-sod} = \gamma_{gen-pre} \circ \gamma_{gen-app}$

Consumable resources. Some resources are consumable in the sense that the times of usage are limited. For example, a member of $board$ can invite at most two agents to a conference by means of delegation; that is, her usage of invitation is limited. Let X be a meta-variable over the members of $board$; the BP-NFA $\gamma_{consume}$ in Figure 5 recognizes one consumption of X's invitation. Hence, $\langle \gamma_{consume} \circ \gamma_{consume} \circ \gamma_{consume}, - \rangle$ forbids X from inviting more than two users.

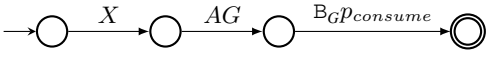


Fig. 5. $\gamma_{consume}$

4.3 Enforcement

Suppose that the resource guard *G* needs to enforce a set C of constraints. We assume *G* maintains a history $\mathcal{H}$ of provenance aware queries that correspond to previous authorizations. Each time an authorization is granted, the associated query is added to $\mathcal{H}$. When a new access request arrives, *G* checks if the policy base entails the corresponding query q; since proof-carrying authorization is used, this amounts to checking the proof. If so, *G* proceeds to verify if C is satisfied by Algorithm 1. The request is allowed if an answer "true" is returned; and $\mathcal{H}$ is updated to $\mathcal{H} \cup \{q\}$. Otherwise, the requested is denied.

The algorithm can be optimized in several ways. First, when only constraints involving BP-NFA are required, the history is not needed. Each time a request comes, we check the corresponding query against BP-NFA. BP-NFA concerns a single agent's authorizations. On the other hand, CP-NFA often involves collusion among multiple agents, which is more costly. Violations against BP-NFA seem more common. Second, one may index historical queries with respect to CP-NFA; this is likely to avoid searching through the history. Finally, constraints based on CP-NFA are usually temporary.

Algorithm 1. Algorithm for enforcing constraints on authorization provenance

```
begin
    foreach ⟨γ, sign⟩ ∈ C do
        if γ is a BP-NFA then
            if γ recognizes q then
                if sign = − then return false;
            else
                if sign = + then return false;
        if γ is a CP-NFA of the form γ1 ∘ ··· ∘ γm then
            if end_γ1[B_G p] ∨ ··· ∨ end_γm[B_G p] then
                for each 1 ≤ t ≤ m, let H(γt) = {D_AEn ··· D_AE1 B_G p ∈ H ∪ {q} | end_γt[B_G p]};
                foreach sequence ⟨q1, ··· , qm⟩ such that, for 1 ≤ t ≤ m, qt ∈ H(γt) do
                    if γ recognizes ⟨q1, ··· , qm⟩ then
                        if sign = − then return false;
                    else
                        if sign = + then return false;
    return true;
end
```

For example, the constraint $\langle \gamma_{consume} \circ \gamma_{consume} \circ \gamma_{consume}, - \rangle$ is only effective during the conference; related queries can be discarded after the conference. This helps reduce $\mathcal{H}$'s size. We leave it to future work to investigate the optimization in detail.

5 Related Work and Conclusions

Various mechanisms have been devised to track, store, and query provenance in database and file systems [7,11]. While one may borrow ideas from these techniques, it is not clear how to adapt them into distributed authorization arena. Actually, authorization provenance deserves investigation in its own right. For example, of importance to authorization provenance are agents who affect the authorization, instead of the data inputs and the processes that the inputs undergo. Meanwhile, the security of provenance attracts considerable efforts [2,4,8]; the aim is to protect the confidentiality and integrity of provenance information itself and to provide access control to provenance. This paper, however, focuses on exploring provenance of authorization and its potential usage in access control area, but not on the security of provenance. Provenance is related to the notion of *causality*. van der Meyden [12] discusses the relation between causality and distributed knowledge. He argues that only a predefined set of agents may cause an agent to know a proposition (i.e., certain information), otherwise the system is considered insecure. Our work differs from [12] in two aspects. First, in distributed authorization, the set of agents who may affect the knowledge of an agent can not always be predefined; this uncertainty is a sacrifice for the flexibility via delegation. Second, the emphasis of [12] is on defining secure systems in terms of information flow policies, which defines the causal relation allowed among agents. It is hard to conceive of how to define authorization provenance using the framework in [12]. The following on work by Chong and van der Meyden [3] studies information flow properties using epistemic logic under the similar framework. They put no emphasis on authorization provenance.

In this paper, we defined a notion of authorization provenance, based on the logic DBT. We showed that this notion possesses a collection of interesting properties and

thus captures the intuitions of authorization provenance. As an application of authorization provenance, we also illustrated the specification and enforcement of a new type of security requirements. There are several avenues for future work. On the one hand, more efficient algorithms for enforcing constraints are worth pursuing; on the other hand, we will consider the notion of roles when tracking provenance.

Acknowledgment. This publication was made possible by the support of an NPRP grant (NPRP 09-079-1-013) from the Qatar National Research Fund (QNRF). The statements made herein are solely the responsibility of the authors.

References

1. Appel, A.W., Felten, E.W.: Proof-carrying authentication. In: ACM Conference on Computer and Communications Security, pp. 52–62 (1999)
2. Braun, U., Shinnar, A., Seltzer, M.: Securing provenance. In: Proc. of the 3rd USENIX Workshop on Hot Topics in Security (HotSec) (July 2008)
3. Chong, S., van der Meyden, R.: Deriving epistemic conclusions from agent architecture. In: TARK (July 2009)
4. Hasan, R., Sion, R., Winslett, M.: The case of the fake picasso: Preventing history forgery with secure provenance. In: Proceedings of the 7th USENIX Conference on File and Storage Technologies, FAST (2009)
5. Hu, J., Zhang, Y., Li, R., Lu, Z.: A logic for authorization provenance. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, pp. 238–249 (2010)
6. Mao, Z., Li, N., Chen, H., Jiang, X.: Trojan horse resistant discretionary access control. In: ACM Symposium on Access Control Models and Technologies (2009)
7. Muniswamy-Reddy, K.-K., Holland, D.A., Braun, U., Seltzer, M.I.: Provenance-aware storage systems. In: Proc. of the USENIX Annual Technical Conference, pp. 43–56 (2006)
8. Ni, Q., Xu, S., Bertino, E., Sandhu, R., Han, W.: An access control language for a general provenance model. In: Proceedings of the 6th VLDB Workshop on Secure Data Management (2009)
9. Schneider, F.B.: Enforceable security policies. ACM Trans. Inf. Syst. Secur. 3(1), 30–50 (2000)
10. Sipser, M.: Introduction to the Theory of Computation (2005)
11. Tan, W.C.: Provenance in databases: Past, current, and future. IEEE Data Eng. Bull. 30(4), 3–12 (2007)
12. van der Meyden, R.: On notions of causality and distributed knowledge. In: International Conference on Principles of Knowledge Representation and Reasoning, pp. 209–219 (2008)
13. Wang, Q., Li, N., Chen, H.: On the Security of Delegation in Access Control Systems. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 317–332. Springer, Heidelberg (2008)

A Framework for Collaborative Business Development Based on Middle Agent Model

Hiroki Nakagawa, Akihiko Nagai, and Takayuki Ito

Nagoya Institute of Technology
Nagoya, Japan

Abstract. With a middle-agent based framework, we can expect to figure out a business model that is focused on distributors for win-win situation by revealing the effect in cooperation and collaboration. Distributors can create good cooperation and collaboration by mediating. We give an example of the collaborative development of new products where a distributor mediated. We modeled distributors in a collaborative development and implemented a tool for an agent-based simulation, in which we imagined a product market. We investigated the role and effect of middle agents, distributors. In addition, we proposed a framework for a new business model.

Keywords: Middle Agent, Innovation, Business Model, Management of Technology.

1 Introduction

In this paper, we propose a middle-agent-based framework to innovate a business model to promote consensus building among many companies. This framework is a tool to analyze what is needed for consensus building as well as the effect and influence of collaborative development, which is unclear in the real world. New knowledge about consensus building can be acquired with this framework. Win-win relations for many companies can be realized by promoting consensus building for multiple collaborative development among them.

Cooperation and collaboration help companies collect many resources as well as creating new business and developing new products.

In the semiconductor industry, semiconductor companies develop Application Specific Standard Products (ASSPs) in collaboration with user companies. The collaborative development of products enables them to accurately acquire a market's need and create new technology, But building a consensus is difficult because collaborative development with a user company causes misunderstandings over multi-issues intimately related to each business. With whom and how to build a consensus is an important problem in the collaborative development of products. Many researchers have discussed the methodology of consensus building without discussing who will manage it. For example, Imada pointed out that consensus building based on autonomy and individuality is more important than

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 679–688, 2012.

social integration by a central agreement and analyzed its structure and function without considering management behavior [3]. This situation is the same in the methodology of contracts. Even though Ito discussed consensus building, he concentrated on stakeholder's profits.

In the real world, most transactions among companies are one-to-one. Our paper focuses on consensus building in multiple collaborative development among many companies. In this paper, we utilize middle-agent-based framework to promote consensus building among many companies. When we design a business model among many companies in the real world, they not only match and adjust in collaborative development but also promote and innovate a business model (Section 2).

This paper is organized as below. The next section shows a concrete example in which a semiconductor distributor mediated between semiconductor and user companies while developing an ASSP. The third section shows a model for negotiating collaborative development transactions. The fourth section shows an agent-based simulation result from that model. Finally, the last section provides a conclusion.

2 Pachinko Manufacturer Needed a High-Performance Graphic LSI (Large Scale Integration)

Pachinko manufacturers noticed that graphic LSI added an important element to enhance the sense of fun. To solve this problem, Company B, a large pachinko manufacturer, developed a graphic LSI in collaboration with a semiconductor company.

B's Approach to Graphic LSI. First, Company B tackled the development of its own graphic LSI. It approached Company A, a semiconductor distributor, who cooperated with AXELL, a semiconductor company with image processing technology for animation and achievement. A proposed that the three parties develop an ASSP together. B agreed and started development, which was discontinued soon. A proposed the productization of a graphic LSI as an alternative to the development of ASSP to AXELL and B again. B abandoned its graphic LSI but still wanted it, so they decided to participate in the productization of a graphic LSI. This LSI sold rapidly, and in 2009, its sales achieved 23% of all sales. It contributed to A's achievement. In this paper, we analyze Company A's approach to the graphic LSI.

2.1 Why Did the Two Parties Fail to Build a Consensus?

Minimum demand quantity and demand function are two reasons why AXELL and Company B failed to build a consensus.

The Reason Why Minimum Demand Quantity Prevented a Consensus. AXELL and B had to agree about minimum demand quantity and demand function. They failed to agree about the former. For AXELL, minimum demand

quantity is the quantity that is replaced with these costs. For B, minimum demand quantity is the amount of graphic LSIs that it could buy. Since Company B didn't need to cover the development and production costs, AXELL had to solve this problem. B could buy not more than 50,000 graphic LSIs. The other hand, AXELL had to sell not less than 300,000 graphic LSIs annually for FUJITSU to produce them. B's quantity was 50,000 and AXELL's quantity was 300,000. Building a consensus was impossible because of the huge disparity between these two figures.

The Reason Why Demand Function Prevented a Consensus from Being Built. Let us think about demand function next. Functions can be classified into three categories by how much they are demanded: required, convenient, and specific. "Must have" is an essential function for graphic LSIs. "Nice to have" is a recommended function but it isn't essential. "Specific" is a function that B particularly demanded. When producing the ASSP, there was difference in the demand function between AXELL and Company B. B demanded too many functions. Generally in the productization of ASSPs, semiconductor companies give priority to "required" functions and avoid "convenient" or "specific" functions because of the high cost of development and productization. B considered all of functions "required" and demanded all of them. On the other hand, for AXELL, demand functions were those that each pachinko manufacturer recognized as "required." AXELL didn't know which functions were "required." B wanted all functions; AXELL didn't know which function was demanded. They couldn't build a consensus. Table I summarizes this situation. The huge difference in their minimum demand quantity and demand function prevented them from building a consensus. Failing to realize the productization of the ASSP is undesirable.

Table 1. Difference in minimum demand quantity and demand function

Item	Company B	AXELL
demand quantity	Total 50,000/year	Total 300,000/year
demand function	wanted all functions all "required" 7 essential functions	realize "required" surely didn't realize "nice to have" and "specific" didn't know essential functions

2.2 Participation of a Semiconductor Distributor in Consensus Building

Semiconductor Distributor Mediates to Build a Consensus. In this example, we summarize how a semiconductor distributor filled the gap in the minimum demand quantity and demand function over which they failed to agree. Concerning minimum demand quantity, since Company A thought that it could sell the graphic LSIs for pachinko to other companies, not only Company B, they contracted to buy 300,000 LSIs with AXELL. This proposal filled the gap in the minimum demand quantity between the supplier and the consumer. For its

demand function, Company A inspected 20 pachinko manufacturers, whose customers classified the functions demanded by Company B, and extracted the "required" functions. This proposal, which was based on their investigation about "required" functions, was important for AXELL who now recognized the functions that they should develop. This investigation led to an LSI that contained functions that Company B didn't want, but it agreed with A's proposal anyway. The investigation also showed that there were "specific" and unnecessary functions because they caused a lack of internal memory if the contents become complex. They also cost more and needed high technology to mount the substrate. Due to mediation by A, the semiconductor distributor enabled AXELL and B to fill the gap in the minimum demand quantity and demand function and to build a consensus. Many competitors could get the same high-performance graphic LSI as B but it was worth developing an LSI with "required" functions because they could differentiate their product from others by the contents. The "required" functions secondarily simplified procurement for pachinko manufacturers and lowered market prices. Greater sales increased the profits for AXELL and A.

3 Transaction Model in Product Development

In this section, we focus on distributors as middle-agent and model a co-development transaction among suppliers and users. The model is utilized for middle agent simulation to examine multiple co-development among many companies.

We propose a middle-agent framework for co-development based on the simulation result. This framework is a tool in order to analyze both what condition is needed for consensus building and how co-development affects companies, which is unclear in the real world. New knowledge about consensus building can be acquired with this framework. This framework can facilitate multiple co-development among many companies.

3.1 Product Market

In this paper, we suppose a market where suppliers s, users b, and distributors d develop products l and deals with product g. Product g is dealt with by two parties, like the good in [7], but it is different from the good because the product doesn't exist initially. The product results from negotiation with a supplier and a user. The product's functions are decided in negotiation, so the same products aren't always distributed. For example, if the product has not less than three functions, it is denoted by $g = (f_1, f_2, f_3)$ with function f. The product's functions are decided by the suppliers, users, and distributors.

3.2 Agent

Suppliers sell products to users. Suppliers have technologies to develop functions mounted on products. Each supplier can develop different functions. They have

secret information about functions they can develop and each function's price. This price is calculated by the sum of cost $c(f)$ and supplier's profit $p_s(f)$. The supplier's utility from product g is profit $u_b(g)$ made by selling the product and is calculated by Eq. 1:

$$u_s(g) = p_s(g). \tag{1}$$

Users buy products from suppliers and have secret information about the functions that they demand and the valuation for function v_b. They also have limitations on maximum demand quantity. A user's utility from product g is different from the product's valuation and its price and is calculated by Eq. 2:

$$u_b(g) = v_b(g) - c(g) - p_s(g). \tag{2}$$

Distributors play the role of both suppliers and users. They mediate with suppliers and users to build consensuses. For this role, they collect secret information from suppliers and users. When a distributor mediates a negotiation, even if a supplier or a user tells the distributor its secret information, the information doesn't directly benefit the distributor. The information benefits the distributor only when the distributor has a successful negotiation. The distributor hence can collect secret information. Based on their technical and demand information, distributors help suppliers and users agree on a product's functions and quantity. Distributors add their profit to the product's price and sell it. The margin is the distributor's utility:

$$u_d(g) = m_d(g). \tag{3}$$

The utility each agent gets is uq when dealing with q products.

3.3 Transaction Process

Users propose their demand functions and negotiate with suppliers. Distributers mediate the negotiation with two parties to help build a consensus We show three kinds of interaction between different agents to explain the actions of suppliers, users, and distributers in this market.

Suppliers and Users. We suppose a case in which a user directly negotiates with a supplier, arranges a product's functions, and handles it. Users propose products with the functions they want to suppliers. In contrast, suppliers present the proposed product prices. Users buy the product by price, and the transaction succeeds with probability $p(q_s, q_b)$, which is denoted by the supplier's minimum demand quantity q_s and the user's maximum demand quantity q_b if the user's valuation exceeds the price presented by the suppliers. Compared with supplier's minimum demand quantity q_s, the bigger user's maximum demand quantity q_b is, the bigger probability $p(q_s, q_b)$ becomes. That is, building a consensus is difficult unless the user buys as many products as demanded by the suppliers. The negotiation breaks down if the valuation is smaller than the price.

Suppliers and Distributors. Consider the case in which suppliers interact with distributors. Suppliers disclose their secret information to distributors. Such secret information is their technology about which functions they can mount on the product and how many products they demand. Distributors are third parties who profit by effecting deals with suppliers and users. Suppliers tell distributors their secret information because the distributors must produce a proposal on which both parties agree to get a profit.

Users and Distributors. Now we examine a case between users and distributors. Distributors act as suppliers for users. In practical terms, this means that users propose products to develop, not to suppliers, but to distributors. Distributors check the user's demand information and the supplier's technical information (each at function prices) to select a supplier as a trading partner. They consider which functions the product should have and propose them. The selected supplier presents its price to the distributor, based on the functions that the proposed product has. Distributors also add profit to the price and present this added price. The transaction succeeds with probability $p(q_s, q_b)$ if the user's valuation exceeds the new price. Also between users and distributors, the transaction depends on the supplier's and the user's quantities. Distributors can search for other users in reference to each user's demand information. Then they help build a consensus by forging deals over the same product with other users (Fig. 1).

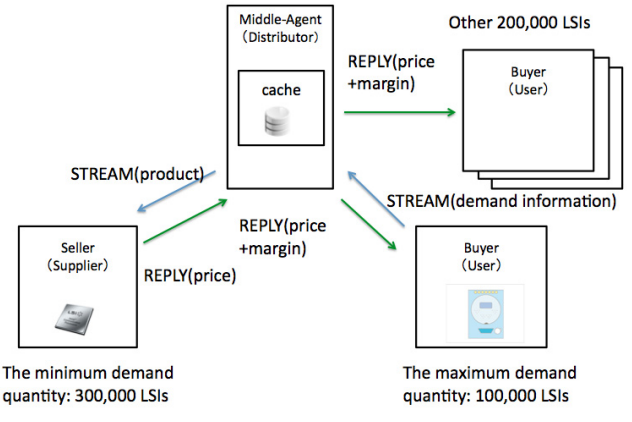


Fig. 1. Model of distributor mediation

3.4 Proposal for Product Function by Distributors

Mediators propose product functions to ask users about their demands and respond with supplier's technology. The strategy proposed by product mediators affects consensus building and each agent's utility. In the real world, mediators employ various strategies [1] [2]. In the semiconductor market, the semiconductor mediator proposed an LSI that has the functions wanted by many users,

that is "required" and sells them to the semiconductor company after agreeing with the quantity and the functions. On the other hand, a mediator might propose a product that only responds to the user's demand. In our model, "required" and "specific" functions are decided by user's needs information that a distributor collected. "Required" functions are those that many users want. In contrast, "specific" functions are those that a few users want. There are other kinds of distributors. One distributor wants larger margin for insurance. The other distributor maximizes customer's profit for a good co-development. In our simulation, mediator agents used the following strategies:

A General demand
In this strategy, the distributor considers general demands based on demand information that it collected to find the combination of functions that mixi-mize the sum of each user's valuation and proposes product g_a that satisfies Eq.4:

$$g_a = \text{argmax}_g \sum_b v_b(g). \tag{4}$$

B Customer demand
The distributor finds the combination of functions that maximize the customer's user's valuation and proposes product g_b that satisfies Eq. 5:

$$g_b = \text{argmax}_g v_b(g). \tag{5}$$

C Myopia
In this strategy, the distributor finds a combination of functions that maximizes its own margin and proposes product g_c that satisfies Eq. 6:

$$g_c = \text{argmax}_g m_d(g). \tag{6}$$

D Hyperopia
Here, the distributor finds a combination of functions that maximizes the sum of the utilities of the supplier and the user for whom the distributer is mediating and proposes product g_d that satisfies Eq. 7:

$$\begin{aligned} g_b &= \text{argmax}_g \{u_s(g) + u_b(g)\} \\ &= \text{argmax}_g \left(v_b\left(g\right) - c\left(g\right)\right). \end{aligned} \tag{7}$$

4 Simulation

4.1 Setting

We simulated a model in which suppliers, users, and distributors develop and deal with products g, research the effect on distributors, distributors, and consider the result. The simulation tool was developed in Java. In the simulation, we randomly paired the number of agents * 100 times. The pair consists of two

different kinds of agents. For example, there is no pair of a supplier and another supplier. We ran 2,000 simulations and examined the average. The following are the fixed parameters: 100 suppliers, 100 users, 20 functions, the maximum number of functions the product has, the number of functions each supplier can develop, and the number of functions users can demand. Distributors use the general demand strategy and search for a supplier who can develop the product most cheaply from the collected information and mediate a transaction. In this simulation, the other agent parameters include supplier's and user's secret information and the distributor's margin.

4.2 Simulation Results

Effect on Distributors. Fig. 2 indicates the utility obtained by each agent from the product transaction by the number of distributors. The horizontal axis represents the number of distributors and the vertical axis represents the utility values. Suppliers, users, and distributors got higher utility values when distributors were involved. The average agent utility values increased because the increasing rate of the supplier's and the user's utility values exceeded the decreasing rate of the distributor's utility value. Distributors increase the total profit. The distributor's utility value decreased because the number of distributors increased reflects competition among distributors. Trading partners are hampered to some extent if suppliers and users make a deal. In contrast, distributors can get a profit whoever they mediate if they build a consensus. Suppliers and users don't intend to develop another product and make a deal once they have satisfied the quantity condition. Therefore, the utility value average of the distributors drops because distributors get less profit as the other distributors gets. Fig. 3 shows the number of consensuses built by the number of distrib-

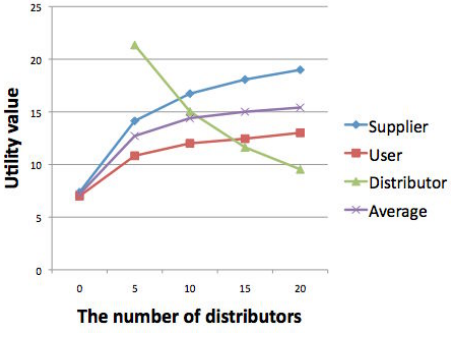


Fig. 2. Each agent utility

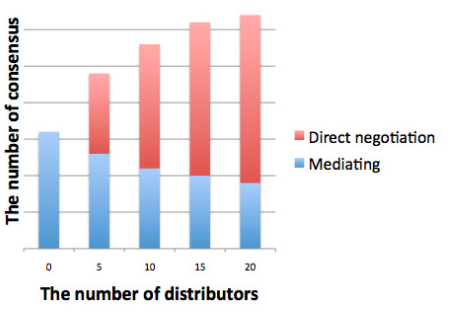


Fig. 3. Number of consensuses

utors. The number increased when distributors existed. The more distributors that existed, the more consensuses that were built. The number of consensuses decreases when a supplier and a user negotiate directly without mediation as

the number of distributors increases. A supplier and user can pair off without mediation but the effect on mediating takes precedence over direct negotiation and builds a consensus.

Examining Distributor Strategies. Here, we simulated which proposal strategy benefits distributors or all agents and the character properties of each strategy. The result without a distributor is shown for comparison. In the simulation setting, the number of distributors is fixed to 5, and the other parameters are the same as above. Each strategy is shown as follows: A: General demand, B: Customer demand, C: Myopia, and D: Hyperopia. First, we compare strategies A and B. The supplier's utility value is higher in A, and the user's utility value is higher in B. The distributor of strategy A emphasizes many user needs to broaden the market for the supplier. Therefore, the distributor probably isn't concerned about customer user needs, so the supplier's utility value is higher and the user's one is lower in strategy A. The distributor of strategy B only considers the customer user need, which is the product that contains "specific" functions so that the market becomes narrower. The supplier's utility value is lower and the user's is higher in strategy B. The distributor of strategy C can't match suppliers and users because they behave selfishly. As a result, they can't help build a consensus and the utility value of all agents decreases. This situation is the same as having no distributor. In contrast, the distributor of strategy D emphasizes the profit of both the supplier and the user without considering their profit to simplify building a consensus; the distributor's utility value also increases. The result is indicated in Fig. 4.

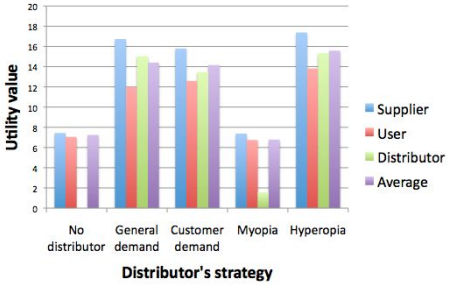


Fig. 4. Agent utility by distributor strategy

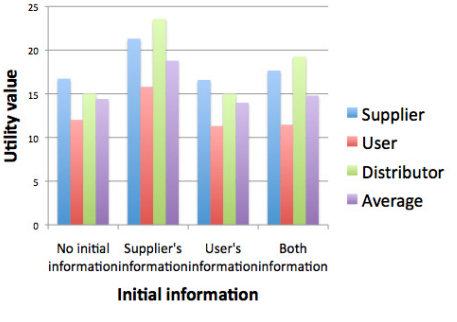


Fig. 5. Agent utility based on initial information

Impact on Secret Information. In this simulation, we examine how the information that distributors collect from supplier's technology and user's demand affects negotiations. Originally, the information was collected from suppliers and users during the transaction. But in this simulation, we gave the distributors the initial information about all suppliers, all users, or both at the beginning to examine the impact of the quantity of information and on what kind of information. Fig. 5 indicates the utility value of each agent based on the initial

information. We assumed that the utility value of each agent increases when distributors have information about both suppliers and users. In fact, the utility value is highest when distributors have only the supplier's information. User information seems to inhibit profit. Distributors exploit user information to sell products to many users. Fig. 3 also shows that consensuses increase as distributors broaden the market. Therefore, the average profit per transaction decreases instead of broadening the market. By selling the same products to many users, distributors have no chance in the future to sell suitable products to users who already have them; the total utility value falls.

5 Conclusion

We focused on a middle-agent solution for consensus building with cooperation and collaboration. For the purpose of proposing a middle-agent framework, we modeled product transaction and ran simulations with it based on a concrete example where a semiconductor distributor mediated. This framework is a tool to analyze the condition and the effect on co-development, which is unclear in the real world. New knowledge about consensus building can be acquired with this framework. The model mediated by distributors as middle agents is a usable framework for increases in the utility values of each agent and builds consensuses when middle agents behave hyperopically. Under this condition, the average profit per transaction decreases. By realizing win-win relations, this framework can facilitate multiple co-development among many companies.

References

1. Nagai, A., Tanabe, K.: Senryakujoho no kyoyu katsuyo niyoru kyodo innovation -AASP gata system LSI kaihatsu no jire. The Development Engineering Society of Japan 30(2), 133–142 (2011)
2. Nagai, A., Tanabe, K.: Shijo to gijutsu wo tsunagu handotaisyosya no enabler kino. Japan Society for Intellectual Production 6(1), 9–14 (2009)
3. Imada, T., Inohara, T.: Syakairiron ni okeru goikese no ichizuke -syakaitogo kara syakaihensyu he. Goikesegaku, Keiso Shobo, pp. 17–35 (2011)
4. Ito, H.: Keyaku no keizairiron. Yuhikaku (2004)
5. Kato, N., Nakajo, M., Kunifuji, S.: Goikese process wo jushisita group ishiketteishien system no kaihatsu. IPSJ Journal 38(12), 2629–2638 (1997)
6. Parsons, T., Shinmei, M.: Politics and Social Structure, pp. 1973–1974. Free Press, Seishin Shobo, New York (1969)
7. Yarom, I., Rosenschein, J.S., Goldman, C.V.: The Role of Middle-Agents in Electronic Commerce. IEEE Intelligent Systems 18 (November 2003)
8. Greenwald, A.R., Kephart, J.O.: Shopbots and Pricebots. In: Proceedings of IJCAI 1999 (International Joint Conferences on Artificial Intelligence), Stockholm, Sweden, July 31-August 6 (1999)
9. Decker, K., Sycara, K., Williamson, M.: Middle-Agents for the Internet*. In: Proc. 15th Int'l Joint Conf. on Artificial Intelligence (IJCAI 1997), pp. 578–583. Morgan Kaufmann (1997)

Agents' Cooperation Based on Long-Term Reciprocal Altruism

Xiaowei Zhao, Haoxiang Xia, Hong Yu, and Linlin Tian

School of Software Technology,
Institute of Systems Engineering,
Dalian University of Technology, Dalian 116024, China
vivian_dlut@163.com, {hxxia,hongyu}@dlut.edu.cn, lelele247@yahoo.com.cn

Abstract. Cooperation among agents is critical for agents' Artificial Intelligence (AI). In multi-agent system (MAS), agents cooperate with each other for long-term return and build such partnership in most of the time. However, the partnership could be broken easily if one agent did not or refused to grant a favor to another. Will it be helpful to MAS or individual agent, if agent has controllable level of tolerance? That is the main question of this paper. In order to find an answer, we propose a cooperative strategy, "flexible reciprocal altruism model (FRAM)". In FRAM, agent has a controllable rate of tolerance and is willing to grant favors for long-term return. Agent can determine whether to grant a favor to another based on their past interactions. As a result, granting unmatched favors by accident will not break the relationship between two agents immediately. Experiments show that our strategy performs well with different cost/value tradeoffs, numbers of agents, and load.

Keywords: Cooperation, Reciprocal Altruism, Multi-Agent System.

1 Introduction

A key idea behind multi-agent systems (MAS) is to establish cooperation between different agents so as to solve problems more effectively and efficiently than a single agent can do. Example applications include cooperative target observation [1], foraging [2], and peer-to-peer systems [3]. What's more, cooperation is the precondition for further coordination, which is vital to improve the performance of the entire system [4]. There are two modes to realize the cooperation in multi-agent systems. One mode is two (or more than) agents fulfill a task and share the reward, the other mode is reciprocal altruism.

Cooperation based on reciprocal altruism is a win-win situation for both multi-agent systems and individuals. However, the agents typically have limited resources and capabilities that restrict the amount of cooperation they can handle at a given time. Thus, a decision mechanism is necessary for an agent to approve or decline a coming request of cooperation. In an open system, such decision is commonly made by the agent in an autonomous and distributed manner, since the centralized control is absent in such system. For an agent, the approval of one

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 689–698, 2012.

request usually means the declination of another at the same time. Therefore, such decision is not straightforward in an agent society, as in the analogous situations in human society. It is deserved to study inter-agent cooperation strategies for the sake of both the society's (i.e. the multi-agent system) performance and the benefit of the intelligent agent itself.

One useful approach to agent cooperation is to consider the agents as providing favors to the other agents. Most of the existing work on favors has naturally followed from some version of reciprocal altruism: one agent is willing to incur a cost now to provide a value to another agent in return for (hopefully) receiving value in the future from the other agent, who in turn will incur some cost. Existing work in reciprocal altruism in multi-agent systems has focused to date on rational agents, trust, social laws, and reputation. We instead tackle the issue from the viewpoint of cooperation with tolerance based on long-term return: whether an agent is willing to help another agent depends on their interactive history and the tolerance of agent (the degree of altruism). This flexible reciprocal altruism leads to direct consequences: if agent i refuses to do agent j a favor, agent j will consider the coordinative history between them. If the damage is not beyond the tolerance of agent j, agent j will keep the relationship with agent i instead of retaliating at once. The mode we propose amortizes agent's loss over all previous interactions with another agent, and it's more flexible and encourages agents cooperate more and more.

2 Related Work

The social science and economics communities have devoted considerable attention to cooperation, in particular, the nature of altruism and why it evolved cite[5][6]. Continuing the same line of reasoning, several researchers examined how reciprocity and altruism effect the development of social agents [7][8][9][10]. More closely related to our work, Trivers examined how reciprocal altruism evolved in nature[11].

Using Triver's work as motivation, Axelrod developed some of the original theory of cooperation in a game-theoretic setting [12]. He developed an evolutionary model based on the probability that two agents would interact again in the future, and showed how stable behavior arises, when agents exhibit deterministic reciprocity towards each other. Sen and his colleagues extended this work to probabilistic reciprocity, using publicly available discount factors to encourage sharing of resources [13][14][15]. This is related to our approach and we elaborate on a portion of this work in next section. Hazard further broadened Sen's work by using private discount factors.

Closely related to reciprocity and cooperative game theory is the notion of trust and its extension in the form of reputation. Since interactions form the basis of a multi-agent system, the multi-agent community has devoted considerable effort towards understanding trust in large scale systems [16][17]. Hand in hand with trust is the idea of reputation [18][19][20]. While trust and reputation are significant research areas within multi-agent systems, our flexible reciprocal

altruism does not use either mechanism *per se*; instead it relies on historical pairwise interactions.

3 Model Description

In our model (FRAM), an agent will achieve benefit by fulfilling tasks which on the other hand can not be accomplished by any agent itself. Agent with task will ask for help from others to achieve its goal. The agent who receives the request from asker will decide whether to grant favors by considering the whole history with the asker. The helper's reciprocal behavior will lead to some cost and it will not receive an immediate benefit in return. If the asking agent has provided many significant favors in the past, the agent who receives the request is willing to offer the grantee some unmatched favors in the future. By then, a cooperative partnership is built up based on long term reciprocal altruism Imagine that agent i is asking a favor from agent j and agent j is determining whether to grant agent i a favor. Let favor$_{ji}$ be the sum total value to agent j of the favors which agent j has granted agent i up to this point and favor$_{ij}$ is the one which agent i has granted agent j. Let refuse$_{ji}$ be the sum total value to agent j of the favors which agent j has refused or failed to grant agent i up to this point and refuse$_{ij}$ is the one which agent i has done to agent j. Agent i will grant the proposed favor if:

$$(favor_{ji} - favor_{ij}) - (refuse_{ji} - refuse_{ij}) + cost \leq \delta \tag{1}$$

The first part of inequality (1), i.e. "(favor$_{ji}$ - favor$_{ij}$) - (refuse$_{ji}$ - refuse$_{ij}$)", stands for the interactive history between two agents, while "cost", the second part of the inequality, means the favor that agent i asks for from agent j. The idea here is that the degree to which agent j will go out on a limb for agent i is no more than the value of δ. The attribute δ is essentially a measure of risk tolerance: highly altruistic agents may have a high value of δ, whereas risk-averse agents will have a low value. If agent i and agent j never interact with each other, there will be no history and it will lead to:

$$(favor_{ji} - favor_{ij}) - (refuse_{ji} - refuse_{ij}) = 0$$

Rearranging the inequality 1, we get:

$$cost \leq \delta \tag{2}$$

Here shows that δ determents the degree of risk tolerance to help an unknown agent. If δ is 1, it means agent can offer a favor (value 1) or accept a refuse (value 1) from its partner. Along with the increasing value of δ, agent gets more and more generous. When δ reaches certain value which is big enough, agent will always help others without considering the transaction history and the behavior of agent can be seen as complete altruism. When δ goes down to minus, agent

will behave in a cautious way and will never take the risk to offer a favor first until it receives other's help with certain amount.

It's worthwhile to compare FRAM with other strategies. Now consider a simple norm, Tit-for-Tat (TFT) which is well discussed by Axelrod [12], disregards the relative history between two agents and the granting decision is only based on the very last experience. Here, agent A will only grant B a favor if B offer A the same amount of favor in the very last time-step. For Tit-for-Tat, agent is simply copy the other's behavior and ignore long history and other factors in systems. TFT is a sensitive strategy when it encounters noise. Imagine if agent A offered agent B a favor, but B did not receive it due to noise. In TFT systems described above, ultimately each agent believes the other owes it, and refuses further interactions. This is called Trembling Hands effect.

We have considered two extreme situations, good man and cheater. Good-man is a certain kind of agent who will always grant favors without considerations of its own gain and loss. Good-man keeps the relationships with anyone in system and maximizes system utility. It is easy to notice that the main shortcoming of good-man is its cost from unmatched granting favors. On the other hand, cheater always wanders among agents and asks for help greedily, but never help others in return. Good-man will never detect (or detect but ignore) cheater's greedy behavior for its optimism, if it meets cheater. If Tit-for-Tat meets cheater, it will refuse to grant any favor once it is betrayed. Since Tit-for-Tat never receives any help from cheater, it will terminate the relationship forever. In our model, if an agent A meets an unknown agent B (cheater) for the first time, A will always agree to offer B a small favor no greater than δ. If B keeps asking for unmatched favors, agent A will refuse to help.

4 Experimental Design

Our experimental problem is an abstraction of common basic factory floor models. In order to force agents cooperate with others, every task in our experiments needs two agents to complete. In every time step, certain numbers of tasks are put into the community. New tasks stochastically arrive to some agents. Every agent with task has to ask another random free agent for a favor to fulfill it. Agent who receives the request considers whether to help based on inequality 1. If it refuses to grant a favor, the agent with task loses the chance to accomplish the task and the task will be dropped on the floor. Since the utility of agent is limited, an agent can only grant a favor once in a time step. If an agent who has already offered a favor in a time step receives another request, it has to refuse for the reason of being busy. Both grantor and grantee (or refuser) update their interaction history.

In our experiments, the number of agent (N) and the number of task (M) put into systems are constants. We add a new attribute K to stands for the ratio of N/M. We suppose all tasks have the same cost and the same benefit. The ratio of gain and loss (G) will be analyzed. The δ is the most important Parameter will be analyzed as well. Each set of parameter values is used to simulate 5 runs,

where each run consists of 10,000 time-steps. This number of rounds is chosen to derive a stable distribution of population characteristics.

5 Results

Our primary statistics are number of tasks performed by all the agents in systems and the performance of individuals in the allotted time. We choose these two statistics for different reasons. The first statistic can describe the utility of MAS with different kinds of strategy (good-man, TFT, FRAM). The second statistic, the performance of individuals, includes the amount of favor which an individual has offered, the amount of refuse which an individual has given and the total value (based on different benefit-to-cost Ratio) an individual can get in the allotted time.

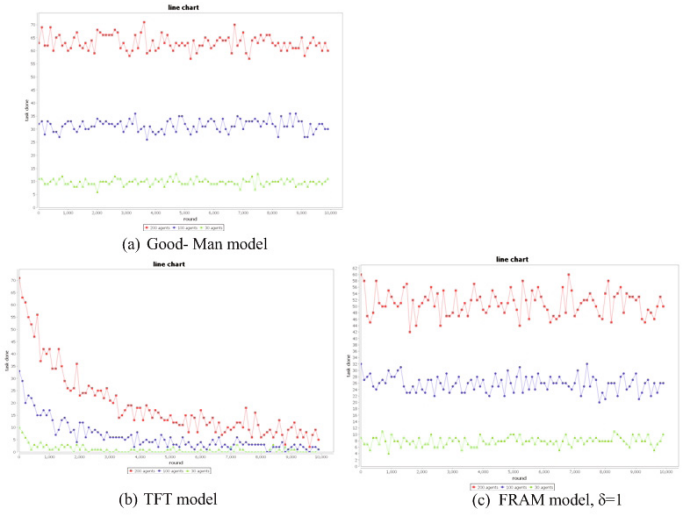


Fig. 1. Total number of tasks per time-step for 200,100 and 30 agents (K=N/M=2) in different systems. (10,000 time-steps).

5.1 Experiment 1: Good-Man, TFT and FRAM

Firstly we compare three different strategies in their own experiment systems. In each system, all the agents use one of three strategies above to coordinate with others. We choose the number of tasks performed by all the agents as our primary statistics to compare the utility of different systems.

Fig. 1 shows the number of tasks done for 10,000 time-steps. As would be obvious, good-Man never refusing others except being occupied resulted in the most number of tasks being accomplished (see Fig. 1a). When number of agent (N) was 200, 63.36% tasks were done in good-man community. When N was 100, 63.56% tasks were done. When N was 30, 64.44% tasks were done.

On the other hand, TFT got lower and lower numbers of tasks completed with time going by (see Fig. 1b). Once an agent got a refuse from its partner, it would revenge in next time-step and the partner would repay in kind, and so on. Thus, the performance of Tit-for-Tat was getting more and more terrible and agents stopped helping each other in the end.

Although we set very small value of δ (only 1) in FRAM model, cooperation could still continue among agents (see Fig. 1c). The the rate of task done was lower than Good-man but still considerable. When N was 200, 51.14% tasks were done. When N was 100, 51.16% were done. When N was 30, 51.74% tasks were done. Once refuse would not lead to a break of partnership. Agents were patient in a limited level and helped each other for lone-term return.

5.2 Experiment 2: Task Load, Number of Agents, Degree of Altruism

We use the model we propose (FRAM) in this experiment. We intend to figure how the values of task load (M), number of agents (N), the ratio of N/M (K) and the degree of altruism (δ) influence agents' interactions. We still choose the number of tasks performed by all the agents as our primary statistics to compare the utility of systems with different factors.

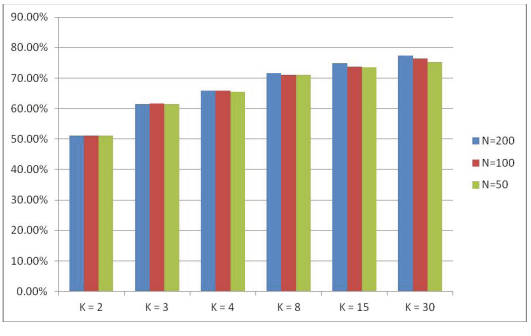


Fig. 2. Rate of tasks done with different K in FRAM systems (δ=1,time-steps=10,000)

We tested systems with different values of K at first. We set the value of δ=1 in this part of experiment. As shown in Fig. 2, K influenced the performance of cooperation dramatically. The rate of tasks done was getting bigger and bigger when K increased. It was easy to understand that the biger K, the more free agents in systems. This led to lower rate of being busy when agent was asked to offer a favor.

We tested systems with different values of δ secondly. We set the value of K=2 in this part. As we can see in Fig. 3, bigger δ made agents be more tolerant and refused others less. Another interesting issue shown in Fig. 3 is when δ was big enough (reached 30), the rate of tasks done was almost equivalent to the rate of good-man model.

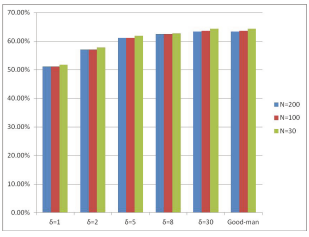


Fig. 3. Rate of tasks done with different δ in FRAM systems (K=2,time-steps=10,000)

In order to disclose the influence of δ, we set N, M and K constants and increased the value of δ gradually. Fig. 4 shows the rate of tasks done rose sharply at the beginning when δ was from 1 to 3 (rate changed from 61.59% to 73.14%), and then it rose smoothly when δ was from 4 to 9 (rate changed from 73.14% to 73.69%). If δ increased over 10, the rate of tasks done was almost stable and changed less than 3%.

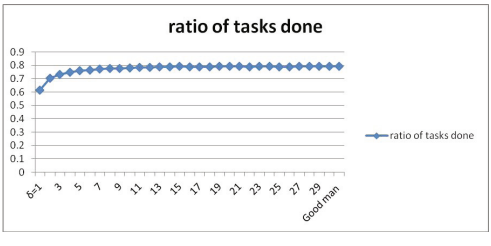


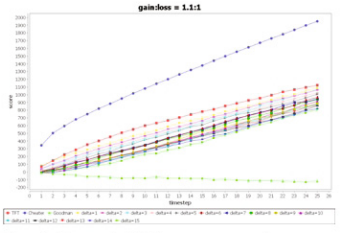
Fig. 4. Total ratio of tasks done with different δ in FRAM systems. (N=200, K=3, time-steps=10,000).

5.3 Experiment 3: Mixed Models

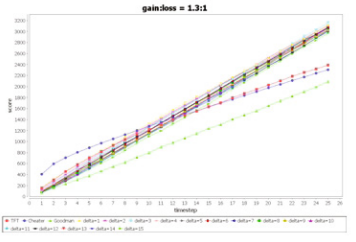
In this experiment we choose mixed types of agent (two or more than two strategies) and observe their different performances. Instead of total number of tasks fulfilled, we used the performance of individuals as statistic, which includes the amount of favor an individual has provided, the amount of refuse an individual has given and the total value an individual can get in the allotted time.

In this part, there are four kinds of agents in the community, cheater, good man, TFT and FRAM with different δ. We try to find out which strategy is optimal in mixed systems. With the numbers of tasks done and the numbers of favors offered by each kind of agents, we can compare the performance of the agents by calculating their gain and loss. In this section, there are two main factors could affect the result. The first factor is the number of different type of agent in the system (agent distribution), the other one is the ratio of gain and loss. In our experiment, an agent will get benefit (gain) if it gets help from others and it will make loss if it offers help to others.

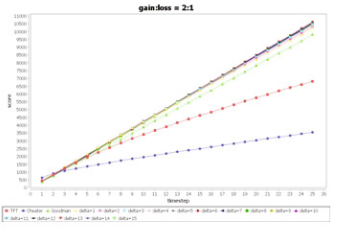
We calculate the profits of the agents with different strategies, as shown in Fig. 5 and Fig. 6, respectively. In Fig. 5, the number of cheaters is small (8) and most agents in the system are cooperative including good man, TFT and FRAM. When the ratio of gain and loss is small (less than 1.2:1, see Fig. 5a),



(a) gain:loss = 1.1:1, best strategy : cheater

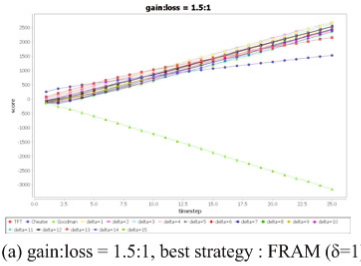


(b) gain:loss = 1.3:1, best strategy : FRAM (δ=3)

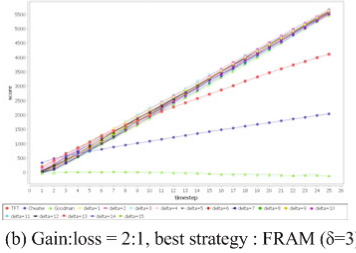


(c) gain:loss = 2:1, best strategy : FRAM (δ=13)

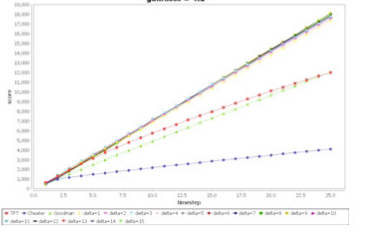
Fig. 5. Average scores of different agents, time-steps=50,000, sampling step=2000, Agent numbers: good man=8, TFT=9, cheater=8, FRAM=75



(a) gain:loss = 1.5:1, best strategy : FRAM (δ=1)



(b) Gain:loss = 2:1, best strategy : FRAM (δ=3)



(c) Gain:loss = 4:1, best strategy : FRAM (δ=11)

Fig. 6. Average scores of different agents, time-steps=50,000, sampling step=2000, Agent numbers: good man=5, TFT=5, cheater=50, FRAM=40

the best strategy is cheater. Along with the increasing ratio, the performance of cheater is getting down, and FRAM is getting better and better (see Fig. 5b and Fig. 5c).

In Fig. 6, the number of cheater is big (50) and that means the half agents in the system will not cooperate with others. As we can see, when the ratio of gain and loss is small (1.5:1, see Fig. 6a), good man lost lots of value. With the setting of 50 cheaters in the system, we have to increase the ratio to more than 2 if we want to protect good man (see Fig. 6b and Fig. 6c). Compare Fig. 5 and Fig. 6, we can see that increasing the ratio of gain and loss will help "good" agents to get higher scores. Bigger number of cheaters affects good-man strategy dramatically but can not affect FRAM. TFT strategy is never the best but never the worst. FRAM always performs steadily. But we do not find a universal optimal strategy on the basis of our different assumptions.

6 Conclusions and Future Work

We have presented a simple formula which enables agents to cooperate with each other with a certain level of tolerance to achieve long-term return. Though at any time an agent can cheat another agent and walk away, the second agent's amortized loss is bounded. This optimistic approach allows agents to optimize the collective performance of the system, and is particularly apropos to environments with moderate numbers of agents, significant and highly variable task load, and moderate noise.

For future work: our existing approach lacks a time-discounting procedure: agents have perfect memory of all previous interactions. We have not yet examined the effects of different ways of doing this. We may also examine possible approaches to transitivity of agents' experiences, and agent can determine whether to grant a favor from neighbors' or partners' interaction history. Last, we have not experimented with self-adapting values of δ. It might be helpful to consider dynamic δ values, producing nonlinear functions or ones designed to cut off free-riders more effectively.

Acknowledgments. This work is partly supported by the Natural Science Foundation of China as a key project under grant no. 71031002.

References

1. Luke, S., Sullivan, K., Balan, G., Panait, L.: Tunably decentralized algorithms for cooperative target observation. In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, Utrecht Netherlands (2005)
2. Gheorghe, M., Holcomb, M., Kefalas, P.: Computational models of collective foraging. Biosystems 61(2-3), 133–141 (2001)
3. Camorlinga, S., Barker, K., Anderson, J.: Multiagent systems for resource allocation in peer-to-peer systems. In: Proceedings of International Symposium on Information Communication and Technologies, pp. 1–6 (2004)

4. Ciacarini, P., Omicini, A., Zambonelli, F.: Multiagent System Engineering: The Coordination Viewpoint. In: Jennings, N.R. (ed.) ATAL 1999. LNCS (LNAI), vol. 1757, pp. 250–259. Springer, Heidelberg (2000)
5. Dugatkin, L.A., Wilson, D.S., Farrand III, L., Wilkens, R.T.: Altruism, tit for tat and 'outlaw' genes. Evolutionary Ecology 8(4) (1994)
6. Lehmann, L.: The evolution of cooperation and altruism: a general framework and a classification of models. Evolutionary Biology 19(5) (2006)
7. Briggs, W., Cook, D.: Flexible social laws. In: Proceedings 14th International Joint Conference on Artificial Intelligence, pp. 688–693 (1995)
8. Grimaldo, F., Lozano, M., Barber, F.: A multiagent framework to animate socially intelligent agents. In: Innovations in Hybrid Intelligent Systems. Springer (2008)
9. Lerman, K., Shehory, O.: Coalition formation for large-scale electronic markets. In: Proceedings of Fourth International Conference on Multi-Agent Systems (2000)
10. Matsubayashi, K., Tokoro, M.: A collaboration mechanism on positive interactions in multi-agent environments. In: Proceedings of International Joint Conference on Artificial Intelligence (1993)
11. Trivers, R.L.: The evolution of reciprocal altruism. The Quarterly Review of Biology 46(1), 35–37 (1971)
12. Alexrod, R.: The Evolution of Cooperation. Basic Books (1984)
13. Saha, S., Sen, S.: Reciprocal negotiation over shared resources in agent societies. In: Proceedings of Autonomous Agents and Multiagent Systems (2007)
14. Sen, S.: Reciprocity: A foundational principle for promoting cooperative behavior among self-interested agents. In: Proceedings of the Second International Conference on Multiagent Systems, pp. 329–332 (1996)
15. Sen, S.: Believing others: Pros and cons. Artificial Intelligence (2002)
16. Ramchurn, S.D., Huynh, D., Jennings, N.R.: Trust in multi-agent systems. The Knowledge Engineering Review 19(1), 1–25 (2004)
17. Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: Proceedings of International Joint Conference on Artificial Intelligence (2007)
18. Pujol, J.M., Sanguesa, R., Delgado, J.: Extracting reputation in multi agent systems by means of social network topology. In: Proceedings of Autonomous Agents and Multiagent Systems, pp. 467–474 (2002)
19. Sabater, J., Sierra, C.: Reputation and social network analysis in multi-agent systems. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 475–482. ACM, New York (2002)
20. Stephens, C.: Modeling Reciprocal Altruism. British Journal for the Philosophy of Science 47(4), 533–551 (1996)

A Vietnamese Text-Based Conversational Agent

Dai Quoc Nguyen[1], Dat Quoc Nguyen[1], and Son Bao Pham[1,2]

[1] Faculty of Information Technology
University of Engineering and Technology
Vietnam National University, Hanoi
{dainq,datnq,sonpb}@vnu.edu.vn
[2] Information Technology Institute
Vietnam National University, Hanoi

Abstract. This paper introduces a Vietnamese text-based conversational agent architecture on specific knowledge domain which is integrated in a question answering system. When the question answering system fails to provide answers to users' input, our conversational agent can step in to interact with users to provide answers to users. Experimental results are promising where our Vietnamese text-based conversational agent achieves positive feedback in a study conducted in the university academic regulation domain.

1 Introduction

A text-based conversational agent is a program allowing the conversational interactions between human and machine by using natural language through text. The text-based conversational agent uses scripts organized into contexts comprising hierarchically constructed rules. The rules consist of patterns and associated responses, where the input is matched based on patterns and the corresponding responses are sent to user as output.

We focus on the analysis of input text in building a conversational agent. Recently, the input analysis over user's statements have been developed following two main approaches: using keywords (ELIZA [10], ALICE [9], ProBot [7]), and using similarity measures (O'Shea et al. [6], Graesser et al. [1], Traum [8]) for pattern matching. The approaches using keywords usually utilize a scripting language to match the input statements, while the other approaches measure the similarity between the statements and patterns from the agent's scripts.

In this paper, we introduce a Vietnamese text-based conversational agent architecture on a specific knowledge domain. Our system aims to direct the user's statement into an appropriate context. The contexts are structured in a hierarchy of scripts consisting of rules in FrameScript language [3]. In addition, our text-based conversational agent was constructed to integrate in a Vietnamese question answering system. Our conversational agent provides not only information related to user's statement but also provides necessary knowledge to support our question answering system when it is unable to find an answer.

In section 2, we provide some related works about the text-based conversational agents, the FrameScript scripting language [3], and present our overall

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 699–708, 2012.

conversational agent architecture in section 3. We describe our experiments and discussions in section 4. The conclusion and future works will be presented in section 5.

2 Related Works

2.1 Text-Based Conversational Agents

ELIZA [10] is one of the earliest text-based conversational agents based on a simple pattern matching by identifying keywords from user's statement. ELIZA then transforms the user's statement into an appropriate rule and generates output response.

ALICE [9] is a text-based conversational agent as chat robot utilizing an XML language called Artificial Intelligence Markup Language (AIML). AIML files consist of *category* tags representing rules; each *category* tag contains a pair of *pattern* and *template* tag. The system searches the *pattern* according to user's input, and produces the appropriate *template* as a response. O'Shea et al. [6] proposed a text-based conversational agent framework calculating the similarity between patterns from scripts and the user's input. The highest ranked pattern is selected and its associated response is returned as output. Graesser et al. [1] presented a conversational agent called AUTOTUTOR matching input statements in the use of Latent Semantic Analysis. Traum [8] adapted the effective question answering characters [2] to build a conversational agent also employing Latent Semantic Analysis for pattern matching.

Sammut [7] presented a text-based conversational agent called ProBot that is able to extract data from users. ProBot's scripts are typically organized into hierarchical contexts consisting of a number of organized rules to handle unexpected inputs. Concurrently, McGill et al. [3] derived from ProBot's scripts [7] build the rule system in FrameScript scripting language. FrameScript [3] provides the rapid prototyping of conversational interfaces and simplifies the writing of scripts.

2.2 FrameScript Scripting Language

FrameScript [3] is a language for creating a multi-modal user interfaces. It evolves from Sammut's Probot [7] to enable rule-based programming, frame representations and simple function evaluation.

Each script in FrameScript includes a list of rules matched based on input statements and used to give an appropriate response. Rules are grouped into particular contexts of the form: *context_name :: rule_set*. The scripting rules in the FrameScript language consist of patterns and responses with the form: *pattern ==> response*.

Every context is represented as a script, and a script is considered as a topic in a domain. The domain is responsible for ensuring that the input statement is matched according to the correct scripts. A script has a *trigger* to determine

whether or not an input activates that context or topic. If the trigger does not exist, any input will activate the topic. If an input is matched with a topic's trigger, the topic becomes the current context and the current topic.

Pattern expressions allow the use of the alternatives and existing pattern expressions. Response expressions contain two different types: sequences and alternatives. Sequence of responses has a list of possible responses surrounded by brackets: [response 1 | response 2 | ... | another response]. Responses utilize the '#' and '^' to perform actions such as to change the current context. For example, *#goto(a_script)* or *#goto(a_script, <<trigger>>)* transforms a conversation or interaction from the current context to another one. If '^' is followed by an integer, the numbered pattern component associated with the integer is placed in the output response. Some examples using '^' are described in our companion paper [5].

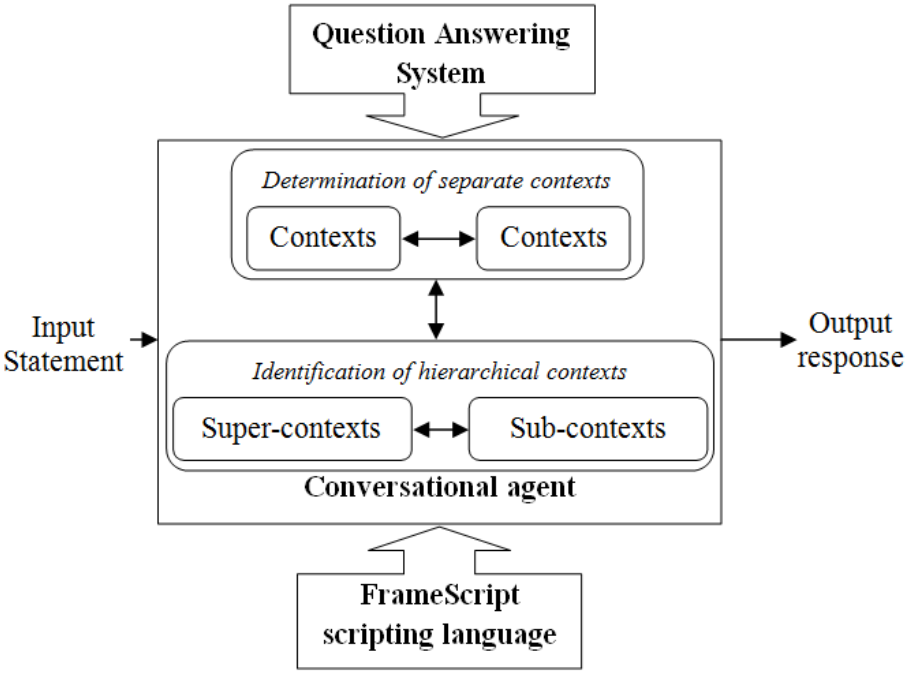


Fig. 1. Architecture of our Vietnamese text-based conversational agent

3 Text-Based Conversational Agent for Vietnamese

The architecture of our text-based conversational agent is shown in figure 1, in which our Vietnamese question answering system is similar to VnQAS [4].

Our question answering system consists of two components: the Natural language question analysis and the Answer retrieval. The communication between two components is an intermediate representation of the input question. If our Vietnamese question answering system is unable to give an answer for a user's question, the question will be considered as the input for our conversational agent. Our contribution focuses on presenting a Vietnamese text-based conversational agent as a backup component to provide necessary information related to the user's question.

The architecture enables the construction of hierarchical rules for conversational contexts using FrameScript language [3]. There are two steps to build these conversational contexts hierarchically: *determining separate contexts* and *identifying hierarchical contexts*. The first step identifies all possible contexts as well as the transitions between contexts. The next step is used to organize these contexts into a hierarchy which can handle unexpected inputs.

Table 1. Script examples of "subjects"

mon_hoc_tien_quyet ::
trigger{* môn học tiên quyết$_{prerequisite\ subject}$ *}
* môn học tiên quyết$_{prerequisite\ subject}$ * ==>
[***Môn học tiên quyết*** của một môn học là môn học bắt buộc sinh viên phải hoàn thành trước khi học môn học đó. Bạn có muốn biết thêm thông tin về ***môn học điều kiện***?
The **prerequisite subject** *of a subject is a imperative subject that students must have completed before learning that subject. Do you want to know about the* **conditional subject**?]
{* Có$_{Yes}$ * | * có$_{yes}$ * } ==>
[#goto(mon_hoc_dieu_kien, <<* môn học điều kiện$_{conditional\ subject}$ *>>)]
{* Không$_{No}$ * | * không$_{no}$ * } ==>
[Bạn có muốn biết thêm thông tin về khóa luận? #goto(khoa_luan)
Do you want to know about the thesis?]
;;

mon_hoc_dieu_kien ::
trigger{* môn học điều kiện$_{conditional\ subject}$ *}
* môn học điều kiện$_{conditional\ subject}$ * ==>
[***Môn học điều kiện*** là các môn học giáo dục thể chất, giáo dục quốc phòng – an ninh và kỹ năng mềm. Bạn có muốn biết thêm thông tin về ***khóa luận***?
The **conditional subjects** *are subjects such as health education,national defence education , and soft skills.Do you want to know about the* **thesis**?]
{* Có$_{Yes}$ * | * có$_{yes}$ * } ==> [#goto(khoa_luan)]
{* Không$_{No}$ * | * không$_{no}$ * } ==>
[Bạn muốn biết thông tin gì?*What information do you want to know ?*]
;;

3.1 Determining Separate Contexts

Given an input statement, the system will identify the current context. The current context is determined by matching pattern expressions of rules to produce associated responses, and to possibly transform to a next context or turn to a previously visited context.

The hierarchical rules are organized into contexts, therefore, the transformation among contexts are based on the change of states from a script to other scripts by using the transformation rules. These transformations depend on the first step aiming to control the conversational stream into the specific knowledge domain. In table 1, scripts *"mon_hoc_tien_quyet"* and *"mon_hoc_dieu_kien"* represent contexts of *"môn học tiên quyết*$_{prerequisite\ subject}$*"* and *"môn học điều kiện*$_{conditional\ subject}$*"* respectively. Both of example scripts representing the corresponding contexts have a *trigger* to identify a matched input to become current context. These scripts contain the transformation rules in the form of *#goto(a_script)* to change the state of current context.

When our text-based conversational agent encounters user's question: *"môn học tiên quyết là gì?"* (*"What is the prerequisite subject?"*). The trigger of script *"mon_hoc_tien_quyet"* will process this input, and context *"môn học tiên quyết* $_{prerequisite\ subject}$*"* will become the current context. If user replies with *"Có*$_{Yes}$*"* for the question:

"Bạn có muốn biết thêm thông tin về môn học điều kiện?"
"Do you want to know about the conditional subject?"

the response expression in transformation rule will perform changing the current context to the context *"môn học điều kiện*$_{conditional\ subject}$*"*. In table 2, we have an example of the transformation between these contexts.

Table 2. Transformations between contexts

User: ***môn học tiên quyết*** là gì? *What is the* **prerequisite subject**?
CA: ***Môn học tiên quyết*** của một môn học là môn học bắt buộc sinh viên phải hoàn thành trước khi học môn học đó. Bạn có muốn biết thêm thông tin về ***môn học điều kiện***?
The **prerequisite subject** *of a subject is a imperative subject that students must have completed before learning that subject. Do you want to know about the* **conditional subject**?
User: ***Có***. Tôi muốn biết. *Yes. I want to know.*
CA: ***Môn học điều kiện*** là các môn học giáo dục thể chất, giáo dục quốc phòng – an ninh và kỹ năng mềm. Bạn có muốn biết thêm thông tin về ***khóa luận***?
The **conditional subjects** *are subjects such as health education, national defence education, and soft skills. Do you want to know about the* **thesis**?
User: Tôi không muốn biết về nó. *No, I do not.*
CA: Bạn muốn biết thông tin gì? *What information do you want to know ?*

Table 3. Order of transformation rules

```
mon_hoc ::
trigger{* môn học_subject *}
* môn học tiên quyết_prerequisite subject * ==>
    [ #goto(mon_hoc_tien_quyet, ^0) ]
* môn học điều kiện_conditional subject * ==>
    [ #goto(mon_hoc_dieu_kien, ^0) ]
* môn học_subject * ==>
[ Các loại môn học gồm có các môn học bắt buộc, các môn học tự chọn,
các môn học tiên quyết của một môn học, các môn học điều kiện và khóa luận.
The subjects consist of imperative subjects, optional subjects, prerequisite subjects,
conditional subjects and thesis. ]
;;

quy_che_dao_tao ::
    * môn học tiên quyết_prerequisite subject * ==>      //Rule 1
        [ #goto(mon_hoc_tien_quyet, ^0) ]
    * môn học điều kiện_conditional subject * ==>      //Rule 2
        [ #goto(mon_hoc_dieu_kien, ^0) ]
    * môn học_subject * ==> [ #goto(mon_hoc, ^0) ]      //Rule 3
;;
```

3.2 Identifying Hierarchical Contexts

After determining the separate contexts, contexts would be arranged into a hierarchy to handle unexpected inputs. Therefore, the second step is used to identify the relation among contexts. Specifically, this is the relationship between super-context and its sub-contexts. A context is the super-context of other contexts if the transformation rule for this context is placed after the transformation rules for the sub-contexts. This aims to recognize the suitable contexts to satisfy input statements.

In table 3, we have the context *"môn học $_{subject}$"* represented by script *"mon_hoc"*, and its two sub-contexts *"môn học tiên quyết$_{prerequisite\ subject}$"* and *"môn học điều kiện $_{conditional\ subject}$"*. Assuming context *"quy chế đào tạo $_{academic\ regulation}$"* described by script *"quy_che_dao_tao"* is being the current context, and the Rule 3 is placed in front of both Rule 1 and Rule 2 in script *"quy_che_dao_tao"*, with the input statement: *"môn học tiên quyết là gì $_{What\ is\ the\ prerequisite\ subject}$?"* or *"môn học điều kiện là gì $_{What\ is\ the\ conditional\ subject}$?"*, the current context is always moved to context *"môn học$_{subject}$"* without transforming to appropriate contexts. Therefore, the output response providing definition of *"môn học$_{subject}$"* does not satisfy user's requirement.

Consequently, the transformation rule for super-context must be constructed to follow transformation rules for its sub-contexts as shown in the script *"quy_che_dao_tao"* in table 3. The transcript of dialogue described in table 4 illustrates the ordered transformation among those contexts.

Table 4. Ordered transformation between contexts

User: Các loại ***môn học*** trong giảng dạy tín chỉ? *Which kinds of* **subjects** *are in the credit−based teaching?* **CA:** Các loại ***môn học*** gồm có các môn học bắt buộc, các môn học tự chọn, các môn học tiên quyết của một môn học, các môn học điều kiện và khóa luận. *The kinds of* **subjects** *consist of imperative subjects, optional subjects, prerequisite subjects, conditional subjects and thesis.*
User: ***môn học điều kiện*** là gì?*What is the* **conditional subject***?* **CA:** ***Môn học điều kiện*** là các môn học giáo dục thể chất, giáo dục quốc phòng – an ninh và kỹ năng mềm. Bạn có muốn biết thêm thông tin về ***khóa luận***? *The* **conditional subjects** *are subjects such as health education, national defence education, and soft skills. Do you want to know about the* **thesis***?*
User: Tôi không muốn biết về nó.*No, I do not.* **CA:** Bạn muốn biết thông tin gì?*What information do you want to know ?*

4 Experiments and Discussion

4.1 Experimental Results for Vietnamese Text-Based Conversational Agent

For this experiment, we built conversational interactions of 16 contexts from a chapter in the academic regulations of the Vietnam National University, Hanoi. Our goal is to support students understanding the academic regulations of the university via a question answering system.

Table 5 describes the transformations among contexts. The conversational interactions between users and our Vietnamese text-based conversational agent start at the default context considered as current context named *"quy chế đào tạo*$_{academic\ regulation}$*"*. The following context *"hình thức dạy học*$_{teaching\ form}$*"* gives students the common information about teaching forms in the university. Then contexts of *"tự học bắt buộc*$_{imperative\ self-study}$*"*, *"lên lớp*$_{on\ class}$*"* and *"thực hành*$_{practice}$*"* aim to detail kinds of teaching forms. Table 5 also shows that the conversational interactions may generate repeated transformations among contexts such as *"giờ tín chỉ*$_{credit\ hour}$*"* and *"tín chỉ*$_{credit}$*"*. Thus we use a method for logging these interactions in order to propose the change to another context.

In our experiment, we collected the inputs from 30 students interacting with our Vietnamese text-based conversational agent. A session contains all communications between one student and the system. On average, a student has 14 interactions (inputs) with the system in a session to retrieve the desire information. We had in total 417 interactions from 30 students. When a response of user input correctly provides the desire information, this response is regarded as satisfying the requirement of students. We achieved an accuracy of 79.4% with 331 input statements having satisfying response from our system.

Table 5. List of transformations among contexts

Context	**Transferred contexts**
quy định$_{regulation}$	hình thức dạy học$_{teaching\ form}$, tín chỉ$_{credit}$
hình thức dạy học $_{teaching\ form}$	lên lớp$_{on\ class}$, thực hành$_{practice}$, tự học bắt buộc$_{imperative\ self-study}$
lên lớp$_{on\ class}$	thực hành$_{practice}$, tự học bắt buộc $_{imperative\ self-study}$
thực hành$_{practice}$	tự học bắt buộc$_{imperative\ self-study}$, lên lớp$_{on\ class}$
tự học bắt buộc $_{imperative\ self-study}$	lên lớp$_{on\ class}$, thực hành$_{practice}$
tín chỉ$_{credit}$	chương trình đào tạo $_{training\ program}$, giờ tín chỉ$_{credit\ hour}$
chương trình đào tạo $_{training\ program}$	hình thức đào tạo$_{training\ form}$
giờ tín chỉ$_{credit\ hour}$	tín chỉ$_{credit}$, môn học$_{subject}$
môn học$_{subject}$	môn học bắt buộc$_{imperative\ subject}$, môn học tự chọn $_{optional\ subject}$, môn học tiên quyết$_{prerequisite\ subject}$, môn học điều kiện$_{conditional\ subject}$, chương trình đào tạo$_{training\ program}$, khóa luận$_{thesis}$

Table 6. Unsatisfying analysis

Reason	Number of user inputs
Constructing of patterns is not appropriate	75
Organizing of hierarchical contexts is not compatible	11

Table 6 presents the error analysis for the 86 inputs which their responses did not satisfy the students. The causes came from the construction of patterns and the organization of hierarchical contexts. It clearly shows that most cases come from constructing patterns of rules. This could be easily rectified by refining or adding more script rules. Table 7 shows the students' degree of satisfaction when interacting with our conversational agent. We provided a scale of 1 to 5 for 30 students to separately evaluate based on the information provided by our text-based conversational agent, that is, 1: bad, 2: normal, 3: good, 4: very good, 5: excellent. The feedback is that most students find the system interesting and highly value the system because of its practical use. To the best of our knowledge, this is the first text-based conversational agent for Vietnamese.

Table 7. The satisfied degree of students

Degree of satisfaction	1	2	3	4	5
Number of students	3	1	13	9	4

4.2 Discussion

Because constructing rules depends on the identification of super-contexts and their sub-contexts in Vietnamese, so it causes difficulties in designing the hierarchy of contexts. Consequently, we want to simplify this designing phase according to the process of semantic knowledge acquisition. We built additional scripts as shown in our companion paper [5] to detect noun phrases, question phrases and relation phrases or semantic constraints between them for Vietnamese. Using these scripts, we constructed pattern expressions and got the suitable phrases from response expressions. These phrases actually are keywords which may be used as patterns of rules in the hierarchical contexts.

In addition, our Vietnamese text-based conversational agent is integrated with our ontology-based Vietnamese question answering system [4] to form a general system. Our goal is to retrieve the necessary information from user's utterance to support our Vietnamese question answering system in providing answers to users. We consider the process that the Answer retrieval component similarly measures between elements of the intermediate representation of user's question and the ontology's elements. In case of ambiguity for the similarity among ontology's elements is still present, the system will interacts with the users by presenting different options to get the correct ontology's elements. In this process, we can construct the supplemental scripts to solve ambiguities from ontology knowledge base. Using these scripts, we can retrieve the suitable elements from ontology through the conversational contexts structured based on the given ontology.

5 Conclusion

In this paper, we proposed a Vietnamese text-based conversational agent architecture as backup component integrated with our Vietnamese question answering system to form a general system. We focused on presenting an approach to construct the hierarchical contexts consisting of organized rules over a specific knowledge domain. There are two steps to construct the conversation contexts: the first step to identify the transformations from a context to other contexts, and the second step to organize these contexts into a hierarchy to handle unexpected inputs. Our contribution is to provide the suitable information related to users' statements and to retrieve the necessary knowledge to support our question answering system in providing answers.

The experimental results are promising, with positive evaluation from users for our Vietnamese text-based conversational agent. To the best of our knowledge, this is the first Vietnamese text-based conversational agent to enable users to interact with the system via a natural language interface.

In the future, our text-based conversational agent will be extended not only to communicate with users but also to get the necessary information related to ontology knowledge base from input utterances. We will build scripts to resolve the ambiguity between elements of ontology such as the similarity of string names among classes or instances in the ontology. The constructed scripts would

be utilized to generate options in order to obtain terms from conversational interactions with users.

Acknowledgements. This work is partially supported by the Research Grant from Vietnam National University, Hanoi No. QG.10.23.

The authors would like to acknowledge Vietnam National Foundation for Science and Technology Development (NAFOSTED) for their financial support to present the work at the conference.

References

1. Graesser, A.C., Lu, S., Jackson, G.T., Mitchell, H.H., Ventura, M., Olney, A., Louwerse, M.M.: AutoTutor: A Tutor with Dialogue in Natural Language. Behavioral Research Methods, Instruments, and Computers 36, 180–192 (2004)
2. Leuski, A., Patel, R., Traum, D., Kennedy, B.: Building Effective Question Answering Characters. In: Proc. of the 7th SIGdial Workshop on Discourse and Dialogue, pp. 18–27 (2006)
3. McGill, M., Sammut, C., Westendorp, J., Kadous, W.: Framescript: A Multi-modal Scripting Language. In: The School of Computer Science and Engineering, UNSW (Copyright © 2003-2008)
4. Nguyen, D.Q., Nguyen, D.Q., Pham, S.B.: A Vietnamese Question Answering System. In: Proc. of the 2009 International Conference on Knowledge and Systems Engineering, pp. 26–32 (2009)
5. Nguyen, D.Q., Nguyen, D.Q., Pham, S.B.: A Semantic Approach for Question Analysis. In: Jiang, H., et al. (eds.) IEA/AIE 2012. LNCS (LNAI), vol. 7345, pp. 156–165. Springer, Heidelberg (2012)
6. O'Shea, K., Bandar, Z., Crockett, K.: A Conversational Agent Framework using Semantic Analysis. International Journal of Intelligent Computing Research (IJICR) 1, 23–32 (2010)
7. Sammut, C.: Managing Context in A Conversational Agent. Electronic Transactions on Artificial Intelligence 5, 189–202 (2001)
8. Traum, D.R.: Talking to Virtual Humans: Dialogue Models and Methodologies for Embodied Conversational Agents. In: ZiF Workshop, pp. 296–309 (2006)
9. Wallace, R.S.: A.L.I.C.E. Artificial Intelligence Foundation, Inc. (2001), `http://www.alicebot.org/about.html`
10. Weizenbaum, J.: ELIZA - A Computer Program for The Study of Natural Language Communication between Man and Machine. Commun. ACM pp. 23–28 (1983)

Using DRSA and Fuzzy Measure to Enlighten Policy Making for Enhancing National Competitiveness by WCY 2011

Yu-Chien Ko[1], Hamido Fujita[2], and Gwo-Hshiung Tzeng[3,4]

[1] Department of Information Management, Chung Hua University, Hsinchu 300, Taiwan
eugene@chu.edu.tw
[2] Software and Information Science, Iwate Prefectural University, Takizawa, Japan
issam@iwate-pu.ac.jp
[3] Graduate Institute of Project Management, Kainan University, Taoyuan 338, Taiwan
[4] Institute of Management of Technology, National Chiao Tung University, Hsinchu 300, Taiwan,
ghtzeng@mail.knu.edu.tw, ghtzeng@cc.nctu.edu.tw

Abstract. The fuzzy measure of competitiveness criteria can be used to enlighten policy making for enhancing national competitiveness. However, fuzzy densities and interactions among criteria are usually unknown or uncertain for implications thus making analysis complicated and hard. This research proposes an extended fuzzy measure to non-additively (or called super-additively) aggregate preferences and implication possibilities into utilities or values, and then implies competitiveness features, patterns, and trends based on the utilities or values. Technically, the dominance-based rough set approach (DRSA) is used to transform '*if…then...*' implications into fuzzy densities. For illustration, the extended fuzzy measure is applied on World Competitiveness Yearbook 2011 for analyzing Greece, Italy, Portugal, and Spain, then how making policy for avoiding debt crisis and enhancing national competitiveness.

Keywords: fuzzy measure, national competitiveness, dominance-based rough set approach (DRSA), World Competitiveness Yearbook (WCY).

1 Introduction

The fuzzy measure can highlight component information in analyzing features [1, 2], patterns [3, 4], and multi-criteria decision making (MCDM) [5, 6, 7, 8]. However, applying its advantages on national competitiveness has two challenges. Firstly, the fuzzy densities are usually unknown and identifying fuzzy densities becomes uncertain when facing a big number of densities. Secondly, the mixed interaction effects as shown in Table 1 might cause ambiguity, thus making analysis complicated and hard [9, 10, 11, 12]. The three typical interaction types are additive, sub-additive, and super-additive effects. The additive type represents that interaction effect matches the expectation of assuming the components independent. The sub-additive interaction,

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 709–719, 2012.

however, yields some substituted effects replacing the expected effect, while the super-additive interaction yields an extended effect in addition to the expected.

Table 1. The mixed interaction effects

Interaction types	Interaction effects
Super-additive effects	Complicated with additional effects to the expected
Additive effects	The expectation of assuming components to be independent
Sub-additive effects	Complicated with substituted effects replacing the expected

Our strategies to the challenges above are described in the following points.

- Replacing the independent densities by implication densities called the conditional fuzzy densities (CFD) which can be indifferently measured. It can make fuzzy measure identification easier when facing a huge number of criteria and endow '*if…then…*' implications into densities.
- Integrating preferences and CFD to fuzzily measure utilities or values. The measuring function is called the extended fuzzy measure which can transform the aggregation into utility or values. The utilities or values thus can verify boundaries for the interaction effect.

To fulfill these two strategies a methodology is designed in Figure 1 initially with a dominance class, unknown CFD, unknown λ-fuzzy measure, and two proposed models. The first model uses the dominance-based rough set approach (DRSA) to solve CFD and the second model is the extended fuzzy measure.

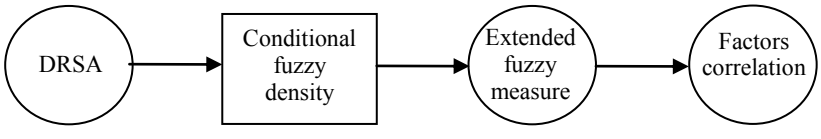


Fig. 1. The proposed methodology

This paper has two main parts. The first is the development of the proposed models. The second is a case study about competitiveness features, patterns, and trends for Greece, Italy, Portugal, and Spain. The remainder of this paper is organized as follows: Section 2 reviews techniques of fuzzy measure and DRSA, Section 3 proposes the extended fuzzy measure for solving non-additive problems in real world, Section 4 lists the results, Section 5 presents discussions and implications, and finally concluding remarks are offered to close the paper.

2 Literature Review

Sugeno presented the theory of fuzzy measures as a means of expressing fuzzy systems in 1974 [5, 9]. The reviewing of the Sugeno's definitions is described in Table 2 with a set, Q, and a set function $g(\cdot)$ called a fuzzy measure for the subsets, $\beta(Q)$, of Q.

Table 2. The definition of fuzzy measure by Sugeno

	Properties	Description
1	$g: \beta(Q) \to [0,1]$, $g(\varnothing) = 0$, and $g(Q) = 1$	Boundaries of the fuzzy measure.
2	$\forall A, B \in \beta(Q)$, if $A \subseteq B$, then $g(A) \le g(B)$	Monotonicity holds for g
3	If $A_i \in \beta(Q)$, $1 \le i < \infty$, and the sequence $\{A_i\}$ is monotonic, then $\lim_{i\to+\infty} g(A_i) = g(\lim_{i\to+\infty} A_i)$.	Continuity holds if the sequence is infinite.
4	If $\forall A, B \in \beta(Q)$, $A \cap B = \varnothing$, and $\lambda \in (-1, +\infty)$ then $g(A \cup B) = g(A) + g(B) + \lambda g(A) g(B)$	λ-fuzzy measure is used to illustrate interaction for disjoint sets.

λ-fuzzy measure has three types of interactions: (i) supper-additive with, $\lambda > 0$; (ii) additive with $\lambda = 0$; (iii) sub-additive with $\lambda < 0$. Let $Q = \{q_1, q_2, ..., q_m\}$ where q_j represents a criterion and Q is finite. The fuzzy measure can be identified as the following formula:

$$g_\lambda(\{q_1, q_2, \cdots, q_m\}) = \sum_{j=1}^{m} g_j + \lambda \sum_{j1=1}^{m-1} \sum_{j2=j1+1}^{m} g_{j1} g_{j2} + \cdots + \lambda^{n-1} g_1 g_2 \cdots g_m = \frac{1}{\lambda}\left[\prod_{j=1}^{m}(1+\lambda g_j) - 1\right]$$

where $-1 < \lambda < \infty$ and $g_j = g_\lambda(\{q_j\})$ is defined as the fuzzy density with respect to q_j by g_λ where $j = 1, \cdots, m$.

DRSA [14, 15, 16,17, 18, 19] can be used to induce entities assigned to $Cl_t^{\ge}$ (the upward union of classes which have entities ranked at least t^{th}) or to $Cl_t^{<}$ (the downward union of classes which has entities ranked less than t^{th}, where Cl is a cluster set containing ordered classes Cl_t, $t \in T$ and $T = \{1, 2, ..., n\}$). For all $s, t \in T$ and $s \ge t$, every entity in Cl_s is preferred to be at least as high as any of entity in Cl_t, which is constructed as $Cl_t^{\ge} = \bigcup_{s \ge t} Cl_s$; inversely, $Cl_t^{<} = \bigcup_{s<t} Cl_s$ for $s < t$. Thus, *P*-dominating or *P*-dominated sets by taking entity x as a boundary through *P*, $P \subseteq C$ can be express as *P*-dominating set: $D_P^+(x) = \{y \in X, yD_P x\}$, *P*-dominated set: $D_P^-(x) = \{y \in X, xD_P y\}$ where $x, y \in Cl$, $y \succsim_q x$ for $D_P^+(x)$, $x \succsim_q y$ for $D_P^-(x)$, and all $q \in P$. Explaining the unions of decision classes by *P*-dominance sets conceptually has been implemented as follows:

$$\underline{P}(Cl_t^{\ge}) = \{x, \in Cl_t^{\ge}, D_P^+(x) \subseteq Cl_t^{\ge}\},\ \overline{P}(Cl_t^{\ge}) = U - \underline{P}(Cl_{t-1}^{<}), \boldsymbol{Bnp}(Cl_t^{\ge}) = \overline{P}(Cl_t^{\ge}) - \underline{P}(Cl_t^{\ge})$$

$$\underline{P}(Cl_t^{<}) = \{x, \in Cl_t^{<}, D_P^-(x) \subseteq Cl_t^{<}\},\ \overline{P}(Cl_t^{<}) = U - \underline{P}(Cl_{t+1}^{<}), \boldsymbol{Bnp}(Cl_t^{<}) = \overline{P}(Cl_t^{<}) - \underline{P}(Cl_t^{<})$$

where $t = 1, ..., n$, $\boldsymbol{Bnp}(Cl_t^{\ge})$ and $\boldsymbol{Bnp}(Cl_t^{<})$ are *P*-doubtful regions. According to Pawlak [20, 21] and Greco et al. [14], two typical coverage rates, the upward unions $Cl_t^{\ge}$ and the downward union $Cl_t^{<}$, are as $CR(Cl_t^{\ge}) = \frac{|\underline{P}(Cl_t^{\ge})|}{|Cl_t^{\ge}|}$ and $CR(Cl_t^{<}) = \frac{|\underline{P}(Cl_t^{<})|}{|Cl_t^{<}|}$.

Another two typical accuracy rates are enlisted as $\alpha(Cl_t^{\geq}) = \frac{|\underline{P}(Cl_t^{\geq})|}{|\overline{P}(Cl_t^{\geq})|} = \frac{|\underline{P}(Cl_t^{\geq})|}{|U| - |\underline{P}(Cl_{t-1}^{<})|}$ and $\alpha(Cl_t^{<}) = \frac{|\underline{P}(Cl_t^{<})|}{|\overline{P}(Cl_t^{<})|} = \frac{|\underline{P}(Cl_t^{<})|}{|U| - |\underline{P}(Cl_{t+1}^{\geq})|}$.

3 The Extended Fuzzy Measure

The research adopts WCY 2011 as the dataset which has 58 nations. There are four factors and twenty criteria presented in Table 3. The proposed extended fuzzy measure is designed and applied on WCY 2011 as illustrations. (Note: the source of WCY 2011 can be reached on https://www.worldcompetitiveness.com/OnLine/App/Index.htm)

Table 3. Four factors and twenty criteria of WCY 2011

	Economic Performance		Business Efficiency
q_1	Domestic Economy	q_{11}	Productivity and Efficiency
q_2	International Trade	q_{12}	Labor Market
q_3	International Investment	q_{13}	Finance
q_4	Employment	q_{14}	Management Practices
q_5	Prices	q_{15}	Attitudes and Values
	Government Efficiency		Infrastructure
q_6	Public Finance	q_{16}	Basic Infrastructure
q_7	Fiscal Policy	q_{17}	Technological Infrastructure
q_8	Institutional Framework	q_{18}	Scientific Infrastructure
q_9	Business Legislation	q_{19}	Health and Environment
q_{10}	Societal Framework	q_{20}	Education

Due to WCY does not provide competitiveness classes this research assumes a dominance class for a case study. According to the dominance class, the extended fuzzy measure treats the product of the conditional fuzzy density and preference as utility in Proposition (1).

Proposition (1): The extended fuzzy measure of competitiveness for nation k is formulated as (1).

$$g_\lambda^k(\{q_1, q_2, \cdots, q_m\}) = \frac{1}{\lambda}\left[\prod_{j=1}^{m}(1+\lambda g_j' r_{kj}) - 1\right] \tag{1}$$

where g_j' represents CFD of the criterion, r_{kj} is the preference value of q_j for nation k, and λ is the interaction degree.

3.1 The Properties of the Extended Fuzzy Measure

The properties of the extended fuzzy measure are presented in Table 4. Property 1 states an extended fuzzy measure function. Property 2 states the range of the extended

fuzzy measure extends from [0, 1] to [0, 100]. Property 3 states that the extended fuzzy measure has interaction effects, too.

Table 4. The properties of the extended fuzzy measure

	Property	Descriptions
1	$g_\lambda^k(\{q_j\}) : g'_j r_{kj} \to [0,100],\ 0 \le g'_j \le 1,\ 0 \le r_{kj} \le 100,$	g'_j represents the conditional fuzzy density for q_j.
2	$g_\lambda^k(\varnothing) = 0,\ g_\lambda^k(Q) \le 100$	Boundaries of the extended fuzzy measure
3	If $\forall q_j, q_i \in Q$ and $q_j \cap q_i = \varnothing$, then $g_\lambda^k(\{q_j\} \cup \{q_i\}) = g_\lambda^k(\{q_j\}) + g_\lambda^k(\{q_i\}) + \lambda g_\lambda^k(\{q_j\}) g_\lambda^k(\{q_i\})$	λ is restricted within the boundaries of the extended fuzzy measure. $\lambda \in R$

Two points cause the extended fuzzy measure different from Sugeno's. Firstly, summation of CFD is not restricted to one. Secondly, $\min\_\lambda \le \lambda \le \max\_\lambda,\ \lambda \in R$ makes boundaries for the interaction effect where

$$\max\_\lambda = \text{MAX}\ \lambda\ \ \textbf{s.t.}\ g_\lambda^k(\{q_1, q_2, \cdots, q_m\}) \le 100,\ k \in \{\text{all nations}\},$$

$\min\_\lambda = \text{MIN}\ \lambda$ **s.t.** $g_\lambda^k(\{q_1, q_2, \cdots, q_m\}) \ge 32.25,\ k \in \{\text{all nations}\}$, 100 being the top score from US, and 32.25 being the bottom score from Venezuela based on WCY 2011.

3.2 The Conditional Fuzzy Densities (CFD)

The properties of DRSA used to derive CFD are listed in Table 5. $q_{j,t}^{\ge}$ is an approximation ($q_{j,t}^{\ge} = \bigcup_{s \ge t} q_{j,s}$), which has nations ranked in at least t with respect to criterion q_j.

Table 5. The properties of DRSA for the conditional fuzzy densities

	properties	Descriptions
1	$q_{j,t}^{\ge} \to Cl_t^{\ge}$	A rule about q_j supports nations to compete the top t positions.
2	$CR(q_{j,t}^{\ge}, Cl_t^{\ge}) = \dfrac{\lvert \underline{P}(q_{j,t}^{\ge}, Cl_t^{\ge}) \rvert}{\lvert Cl_t^{\ge} \rvert},\ \alpha(q_{j,t}^{\ge}, Cl_t^{\ge}) = \dfrac{\lvert \underline{P}(q_{j,t}^{\ge}, Cl_t^{\ge}) \rvert}{\lvert \overline{P}(q_{j,t}^{\ge}, Cl_t^{\ge}) \rvert}$	The coverage and accuracy rates for $q_{j,t}^{\ge} \to Cl_t^{\ge}$.
3	$g'_j = g(q_{j,t}^{\ge} \to Cl_t^{\ge}) = \max_{x,y \in q_{j,t}^{\ge}} \left\{ CR(q_{j,t}^{\ge}, Cl_t^{\ge}) \times \alpha(q_{j,t}^{\ge}, Cl_t^{\ge}) \right\}$ **s.t.** $\underline{P}(q_{j,t}^{\ge}, Cl_t^{\ge}) = \{x \in Cl_t^{\ge} \cap q_{j,t}^{\ge}, D_j^+(x) \subseteq q_{j,t}^{\ge}\}$, $\overline{P}(q_{j,t}^{\ge}, Cl_t^{\ge}) = D_j^+(y),\ x \in Cl_t^{\ge} \cap q_{j,t}^{\ge},\ y \in q_{j,t}^{\ge},\ x \le y \le t$	y is the upper approximation boundary for $q_{j,t}^{\ge}$. x is the lower approximation boundary for $Cl_t^{\ge} \cap q_{j,t}^{\ge}$. $D_j^+(y)$ is a dominance set at least as good as y.

$Cl_t^{\geq}$ is a union of classes which has the nations ranked at least *t* with respect to the competitiveness scoring of WCY 2011. t is set 29th, i.e. CFD represents how strong a criterion supports nations to achieve the upper half positions.

3.3 Solving λ Interval

The maximum and minimum λ can be backwardly solved in **Model I** and **II** by optimally approaching the top and bottom boundaries of competitiveness scores.

Model I MAX λ

$$g_\lambda^k(Q) \leq 100, \ \max\_\lambda = \lambda$$

Model II MIN λ

$$g_\lambda^k(Q) \geq 32.25, \ \min\_\lambda = \lambda$$

4 The Results

The resulted CFD is listed in the left part of Table 6. Institutional framework (q_8) plays a leading criteria in supporting nations to achieve the upper half positions with the conditional fuzzy measure 0.87.

Table 6. The resulted CFD and λ of the extended fuzzy measure for WCY 2011

Fuzzy density for competitiveness criteria								Factors correlation				
E		G		B		I			E	G	B	I
g_1'	0.65	g_6'	0.49	g_{11}'	0.74	g_{16}'	0.77	E	1	0.67	0.74	0.49
g_2'	0.51	g_7'	0.31	g_{12}'	0.52	g_{17}'	0.77	G		1	**0.85**	0.73
g_3'	0.57	g_8'	**0.87**	g_{13}'	0.77	g_{18}'	0.74	B			1	0.62
g_4'	0.52	g_9'	0.74	g_{14}'	0.83	g_{19}'	0.65	I				1
g_5'	0.34	g_{10}'	0.68	g_{15}'	0.63	g_{20}'	0.59	$\lambda = -0.01$				

Note: E: Economic, G: Government, B: Business, I: Infrastructure

The resulted λ interval, $-0.028 \leq \lambda \leq -0.010$, is presented as the slashed area between $\min\_\lambda = -0.028$ and $\max\_\lambda = -0.010$ in Table 7. Obviously, only a single type of interaction exists for aggregation in this case.

Table 7. The range of interaction effect of the extended fuzzy measure

Interaction types	Interaction interval ($-0.028 \leq \lambda \leq -0.010$)
Super-additive effects, $\lambda > 0$	
Additive effects, $\lambda = 0$	
Sub-additive effects, $\lambda < 0$	

The resulted correlations in Table 6 choose $\lambda = -0.01$ to fuzzily measure factors $g_\lambda^k(E)$, $g_\lambda^k(G)$, $g_\lambda^k(B)$, and $g_\lambda^k(I)$ where $E = \{q_1, q_2, q_3, q_4, q_5\}$, $G = \{q_6, q_7, q_8, q_9, q_{10}\}$, $B = \{q_{11}, q_{12}, q_{13}, q_{14}, q_{15}\}$, and $I = \{q_{16}, q_{17}, q_{18}, q_{19}, q_{20}\}$. The choice $\lambda = -0.01$ makes the smallest interaction within the factor criteria and leaves the biggest interaction between factors. A map of factor correlations is shown in Figure 2.

In a summary, this case has achieved three points. Firstly, the conditional fuzzy densities are derived to be implication densities. Secondly, verifying interaction effect is fulfilled. Thirdly, the correlations of aggregation utilities are obtained. These findings are further discussed for Greece, Italy, Portugal, and Spain.

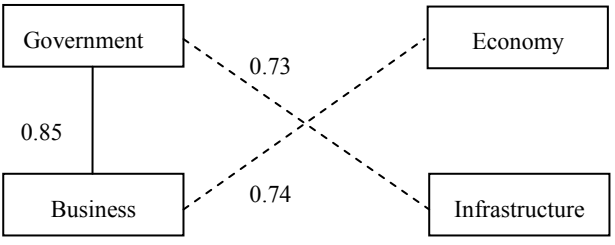


Fig. 2. The factor correlations by the extended fuzzy measure for WCY 2011

5 Discussion and Implications

The correlation feature between the government and business efficiencies of WCY 2011 is presented in Figure 3 in which the vertical axis is scaled by $g_\lambda^k(G)$, the horizontal axis is scaled by $g_\lambda^k(B)$, and the interaction degree within a factor chooses $\lambda = -0.01$. The feature shows that the government efficiency goes up with the increase of the business efficiency. v is a vector directing to better competitiveness. The values of v is $(0.70, 0.71)$ by **Model 3** for 58 nations. Currently, Greece, Italy, Portugal, and Spain have the same problem in debts. They stand close to the solid line which means their government and business have similar relationship. If a proposal can lead a nation to move along the solid line then it can help these four nations.

Figure 3 has two dominance patterns which are formed by a dashed line. The upper half nations belong to the dominating (right) side and the bottom half nations belong to the dominated (left) side. Only two nations violate this rule. The scale of this pattern reveals Spain has a big potential to approach the dominating positions while Greece stays away from the dashed line. Spain very possibly plays a leading role to help the other three nations in the future.

Model 3:

$\text{MIN} = (w_g g_\lambda^k(G) + w_b g_\lambda^k(B) + h)^2$ $k \in$ all nations

s.t. $w_g + w_b = 1$ where w_g and w_b represent weights for $g_\lambda^k(G)$ and $g_\lambda^k(B)$, respectively.

$v \bullet (w_g, w_b) = 0$ where $v = (v_g, v_b)$ and $w = (w_g, w_b)$

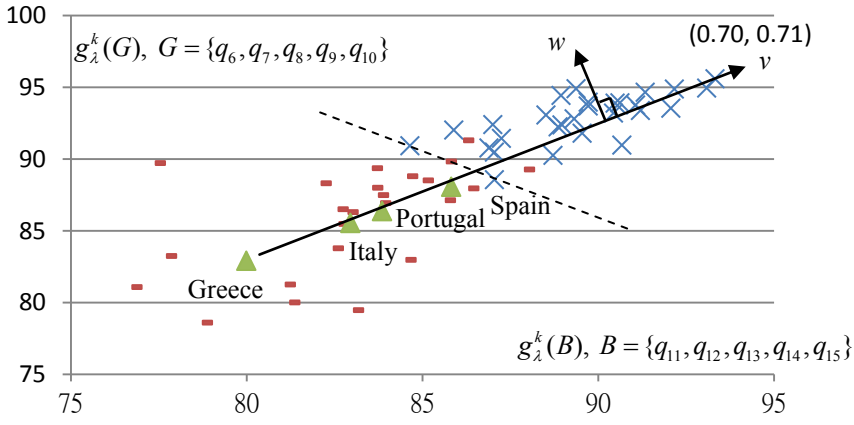


Fig. 3. The dominance pattern and correlation feature of government and business factors by the extended fuzzy measure for WCY 2011

The trends of government and business efficiencies in WCY 1997~2011 are presented as Figure 4 with correlation coefficients of Greece (0.76), Ireland (0.91), Italy (0.82), Portugal (0.61), and Spain (0.94). The vertical axis is the performance scale from 0 to 100. The horizontal axis extends years from 1997 to 2011. The red lines represent the trends of government efficiency and the blue lines for the business efficiency. In vision, Greece, Italy, Ireland, Portugal, and Spain were different from Sweden which has been getting better in government and business efficiencies.

Usually people assume government should lead business development. Greece, Italy, Portugal, and Spain have been keeping their government efficiency under 60. In order to get details inside of these four nations, a vector of the government efficiency in 2011 is solved in Model 4 by approximating the minimum square distance to a hyper plane. The values of the vector include public finance (q_6), fiscal policy (q_7), institutional framework (q_8), business legislation (q_9), and societal framework (q_{10}).

Model 4:

$$\text{MIN} = (w_6 r_{k6} + w_7 r_{k7} + w_8 r_{k8} + w_9 r_{k9} + w_{10} r_{k10} + h)^2 \quad k \in \text{Greece, Italy, Portugal, Spain}$$

$$s.t.\ w_6 + w_7 + w_8 + w_9 + w_{10} = 1,\ w_j \text{ represents a weight for } q_j,\ j = 6,...,10$$

$$v \bullet (w_6, w_7, w_8, w_9, w_{10}) = 0 \text{ where } v = (v_6,\ v_7, v_8,\ v_9, v_{10})$$

The result of Model 4 presents the government characteristic of these four nations with $(0.27,\ 0.28,\ 0.26,\ 0.86,\ -0.22)$. It implies that the improvement of societal framework might scarify public finance, fiscal policy, institutional framework, and business legislation, or vice versa. Especially, the business legislation will be impacted significantly from the change of the societal framework. If an improving proposal in societal framework need not scarify public finance, fiscal policy, institutional framework, and business legislation then these four nations will have a chance to improve stably.

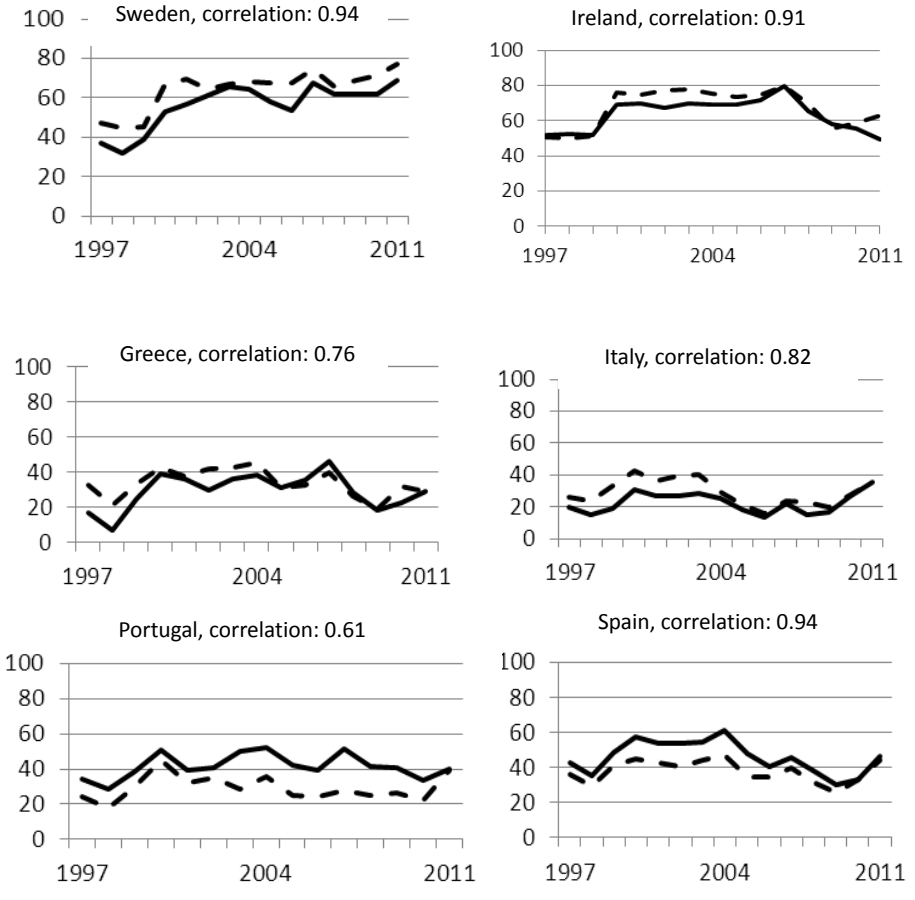


Fig. 4. The trends of government and business efficiency, WCY 1997~2011

In the future work, the multiple objective programming might play an important role to discover a MCDM proposal for Greece, Italy, Portugal, and Spain. Alternatively, national happiness might give another perspective to enhance people's life. Applying the happiness and competitiveness together to overcome debt crisis is a good issue in the future, too.

6 Concluding Remarks

This research discovers the conditional fuzzy densities for criteria, extends the fuzzy measure to aggregate component utilities, implies competitiveness features, patterns, and trends, and finally solves regressive characteristics of government efficiency for Greece, Italy, Portugal, and Spain. The case study on WCY achieves some merits. A competitiveness feature shows that government and business efficiency are highly correlated. A dominance pattern shows Greece, Italy, Portugal, and Spain belong to a

less competitiveness class while Spain has a big potential to achieve the upper half positions. The historical trends during 1997~2011 reveal Sweden successfully overcame the global financial crisis in 2008 due to government and business efficiencies getting better. Conversely, Greece, Italy, Portugal, and Spain have been keeping their government efficiencies in the lower performance. In the future a proposal improving societal framework without scarifying public finance, fiscal policy, institutional framework, and business legislation will help these four nations to grow stably.

References

1. Grabisch, M.: The representation of importance and interaction of features by fuzzy measures. Pattern Recognition Letters 17(6), 567–575 (1996)
2. Mikenina, L., Zimmermann, H.-J.: Improved feature selection and classification by the 2-additive fuzzy measure. Fuzzy Sets and Systems 107(2), 197–218 (1999)
3. Grabisch, M., Sugeno, M.: Multi-attribute classification using fuzzy integral. In: IEEE International Conference in Fuzzy Systems 1992, pp. 47–54 (1992)
4. Keller, J.M., Gader, P., Tahani, H., Chiang, J.-H., Mohamed, M.: Advances in fuzzy integration for pattern recognition. Fuzzy Sets and Systems 65(2-3), 273–283 (1994)
5. Grabisch, M.: Fuzzy integral in multicriteria decision making. Fuzzy Sets and Systems 69(3), 279–298 (1995)
6. Grabisch, M.: The application of fuzzy integrals in multicriteria decision making. European Journal of Operational Research 89(3), 445–456 (1996)
7. Yang, J.L., Chiu, H.N., Tzeng, G.H.: Vendor selection by integrated fuzzy MCDM techniques with independent and interdependent relationships. Information Sciences 178(21), 4166–4183 (2008)
8. Larbani, M., Huang, C.Y., Tzeng, G.H.: A novel method for fuzzy measure identification. International Journal of Fuzzy Systems 13(1), 24–34 (2011)
9. Sugeno, M.: Theory of fuzzy integrals and its applications. Ph.D. thesis, Tokyo Institute of Technology, Tokyo, Japan (1974)
10. Tahani, H., Keller, J.: Information Fusion in Computer Vision Using the Fuzzy Integral. IEEE Transactions on Systems, Man, Cybernetics 20(3), 733–741 (1990)
11. Lee, K.-Y., Leekwang, H.: Identification of λ-fuzzy measure by genetic algorithms. Fuzzy Sets and Systems 75(3), 301–309 (1995)
12. Larbani, M., Huang, C.Y., Tzeng, G.H.: A Novel Method for Fuzzy Measure Identification. International Journal of Fuzzy Systems 13(1), 24–34 (2011)
13. Greco, S., Matarazzo, B., Slowinski, R.: Rough approximation of a preference relation by dominance relations. European Journal of Operational Research 117(1), 63–83 (1999)
14. Greco, S., Matarazzo, B., Slowinski, R.: Extension of the rough set approach to multicriteria decision support. INFOR 38(3), 161–193 (2000)
15. Greco, S., Matarazzo, B., Slowinski, R.: Rough set theory for multicriteria decision analysis. European Journal of Operational Research 129(1), 1–47 (2001)
16. Greco, S., Matarazzo, B., Slowinski, R.: Rough approximation by dominance relations. International Journal of Intelligent Systems 17(2), 153–171 (2002)
17. Fan, T.F., Liu, D.R., Tzeng, G.H.: Rough set-based logics for multicriteria decision analysis. European Journal of Operational Research 82(1), 340–355 (2007)
18. Liou, J.J.H., Tzeng, G.H.: A dominance-based rough set approach to customer behavior in the airline market. Information Sciences 180(11), 2230–2238 (2010)

19. Shyng, J.Y., Shieh, H.M., Tzeng, G.H.: Compactness rate as a rule selection index based on rough set theory to improve data analysis for personal investment portfolios. Applied Soft Computing 11(4), 3671–3679 (2011)
20. Pawlak, Z.: Rough set approach to knowledge-based decision support. European Journal of Operational Research 99, 48–57 (1997)
21. Pawlak, Z.: Rough set, decision algorithm, and Bayes' theorem. European Journal of Operational Research 136(1), 181–189 (2002)

Evaluating Top Information Technology Firms in Standard and Poor's 500 Index by Using a Multiple Objective Programming Based Data Envelopment Analysis

Chi-Yo Huang[1], Po-Yen Wang[1], and Gwo-Hshiung Tzeng[2,3]

[1] Department of Industrial Education, National Taiwan Normal University
No. 162, Hoping East Road I, Taipei 106, Taiwan
cyhuang66@ntnu.edu.tw
[2] Department of Business and Entrepreneurial Administration, Kainan University
No. 1, Kainan Road, Luchu, Taoyuan County 338, Taiwan
[3] Institute of Management of Technology, National Chiao Tung University
Ta-Hsuch Road, Hsinchu 300, Taiwan
ghtzeng@mail.knu.edu.tw

Abstract. Information technology (IT) is defined as the obtainment, procedure, storage and propagation of sounding, drawing, and textual information by combining microelectronics-based computing and telecommunications. Nowadays, IT is starting to spread further from the conventional personal computer and network technologies to integrations of other fields of technology such as the use of cell phones, televisions, automobiles, etc. In other words, IT has penetrated in daily life of human beings and become one part of the whole society. The importance of IT has become momentous. Therefore, to understand the performance of efficiency and productivity of the IT firms is critical for managers as well as for personal investors. Until now, there are very few researches tried to analyze final performance of the IT firms. As a result, this research intends to use traditional Data Envelopment Analysis (DEA) CCR or BCC models to evaluate the performance of IT firms. The Decision Making Units (DMUs) on this research are chosen from IT firms in S&P 500. However, the traditional DEA models are not fair models from the aspect of improper weight derivations. Thus, this paper intends to analyze the efficiency of IT firms in S&P 500 efficiencies by using multiple objective programming (MOP) based Data Envelopment Analysis (DEA). In a MOP based DEA approach, DMUs will be evaluated based on an equal standard and the results will be evaluated more fairly. The world's leading IT firms in S&P 500 will be evaluated based on publicly available financial reports of the fiscal year 2010. In addition, the newly developed MOP can improve the traditional DEA's unfair weights problems and benchmark the efficiency of IT firms in S&P 500 correctly. In the empirical study, the MOP based DEA demonstrated that F5 Networks should be the communications equipment companies of IT worthwhile to be invested. In the future, performance evaluation results can be served as foundations for investment strategies definition.

Keywords: Information Technology (IT), Standard and Poor's 500 index, Performance Evaluation, Data Envelopment Analysis (DEA), Multiple Objective Programming (MOP).

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 720–730, 2012.

1 Introduction

In the twenty-first century, information technology (IT) has penetrated in daily life of human beings and become one part of the whole society. IT was first appeared by Harvard Business Review in 1958 since this new technology does not have a unique established name. Nowadays, IT is starting to spread further from the conventional personal computer and network technologies to integrations of other fields of technology such as the use of cell phones, televisions, automobiles, etc. In other words, IT has penetrated in daily life of human beings and become one part of the whole society. The importance of IT has become momentous.

The field of IT industry is broad. By using Standard Industrial Classification (SIC) code which is a United States government system for classifying industries by a four-digit code to choose IT firms, it shows that every company have different major products in its field. The author selects the communications equipment to evaluate the performance. The reasons for selecting these two classifications of IT firms are mentioned in the following. Jorgenson [1] purposed that communications equipment is an important market for semiconductors owing to the switching and terminal equipments rely deeply on semiconductor's technology.

The main purpose of evaluating the productivity of IT has resulted from the increasing use of IT. There have been some researches concerning the performance of IT by using different methods. Therefore, performance evaluation plays an essential role in management process. It not only provides critical information for decision-making, but also gives a foreseeable advantage for following operations [2]. Hence, how to use performance evaluation to measure organizational performance in a multidimensional construct is important to determine a whole management procedure [3].The data envelopment analysis (DEA) is a flexible tool and can be molded with other analytical methods. Within the techniques of evaluating organization performance, the DEA proposed by Charnes et al. [4] may be the most suitable role in proceeding performance evaluation. Thus, DEA methods have become popular tools, which were widely adopted on national, industrial as well as firm level performance evaluations.

Although DEA approaches were widely adopted on performance evaluations of nations, industries as well as firms, such performance evaluation results were derived based on different bases of comparisons of DMUs. Nevertheless, traditional DEA approaches which were usually leveraged on firms' performances are based on the unfair weights' problems as mentioned by Fare and Hunsaker [5]. Apparently, to resolve the disadvantages being introduced by traditional DEA approaches, an appropriate measure of the IT by a suitable method is required. To resolve the above mentioned disadvantages being introduced by traditional DEA approaches, this research aims to introduce a Multiple Objectives Programming (MOP) based DEA method being developed by Prof. Gwo-Hshiung Tzeng [6] to evaluate the performance of the top IT firms in S&P 500. In the novel MOP based DEA approach, DMUs will be evaluated based on an equal standard [6 ; 7]. By this approach, the efficiency rating of each DMU can be evaluated more fairly than the traditional CCR approach being proposed by Charnes et al. [4].

In this research, to evaluate the performance of top IT firms in S&P 500 and distinguish the differences between the results being derived by either traditional DEA approaches and or the outcome derived by the MOP based DEA, the traditional CCR, BCC

based DEA models will be first introduced as research methods, then the MOP based DEA model will be derived to compare the differences of both methods. The top IT firms in S&P 500 will serve as DMUs. The Efficiency Achievement Measure (EAM) will be introduced by the MOP based DEA method. For each IT companies, the traditional DEA models and the MOP based on DEA model will be used to evaluate the performance of two different models. Finally, by comparing the results of difference models, the evaluation outcome can serve as a basis for investment decisions in the future.

The remainder of this paper is organized as follows. The related literature regarding to performance evaluation and efficiency and productivity of IT firms will be reviewed in Chapter 2. The analytic framework based on CCR, BCC, and MOP based DEA methods will be introduced in Chapter 3. Then, in Chapter 4, the results of evaluating by CCR and MOP based DEA will be shown as an empirical study. Managerial implications as well as discussion will be presented in Chapter 5. Finally, the whole article will be concluded in Chapter 6.

2 Literature Review

Performance evaluation can be considered as a primary element in the human social behavior. Performance evaluation plays an essential role in management process. Performance evaluation not only provides critical information for decision-making, but also provides a foreseeable advantage for following operations [2]. Hence, how to use performance evaluation to measure organizational performance in a multidimensional construct is important to determine a whole management procedure) [3].

2.1 Productivity and Efficiency

Productivity is one of our most basic and intuitive measures of performance. At the firm level, productivity is a component of profit growth along with price changes, and at the aggregate level. Productivity is a fundamental part of economic growth and welfare. To determine the productivity of any organization, it is required to have a tool to measure it. Over the time period, measuring productivity would be helpful for the organization to compare the performance towards the industry of similar firms or similar service providers, and to compare the productivity of a certain department) [8].

The definition of efficiency is that the minimum resource level which is abstractly needed to conduct the desired operations in a given system when contrasted to the actual resource has been used [9; 10]. Based on Coelli et al. [11], the normal measurement of efficiency is used to measure a firm as the ratio of the outputs, and measure produces as the ration of the inputs. The existence of multiple inputs and outputs in regard to distinct resources, actions and environmental factors [12] cause the formula is usually insufficient [11].

2.2 Performance Measurement

Measurement comprises solving the size, sum, or degree of something [13]. Measurement was defined as “the way to judge something” by Collin [14]. The purpose of measurement is not used to demonstrate how well or how badly about the organization will be in the latest reporting period. On the contrary, measurement is used to present the

required data of organization which can later feedback in the expectation of making decisions about the change in demand based on the organization capability [15].

Performance measurement attempts to measure both outcomes of performance or the mean which affects results of performance [13]. Harry [16] defined a simple meaning of performance measurement in government "how efficiency and effectively of services are being delivered to a community in a systematic assessment". In addition, Collin [14] indicated that performance measurement is to accomplish a benchmark program. During conducting any type of performance measurement, it is significant to indicate whether there is any important thing about products or services.

2.3 Performance Evaluation

Performance evaluation is an important part of the procedure of management. Performance evaluation not only provides required information for decision-making, but also gives a rival advantage for following operations [2]. As a result, it is critical to decide how to measure performance more organized in a multidimensional construct [3]. Among the methods of evaluating organizational performance, the technique of data envelopment analysis (DEA) proposed by Charnes et al. [4] will be the most representative method of performance evaluation.

The performance evaluation based on DEA method has been widely used in a variety of fields, including airlines, banking, insurance, life insurance companies, telecommunications, transportation companies, textile companies, hotel industry, supplier selection, and high-tech companies.

Grinstein et al. [17] used 24 measurements belonging to 19 characteristics to evaluate and interview the performance of high-technology firms. Grinstein et al. [17] discovered the connection to improve several dimensions that can be used by other fields to define technology firms and classify firms based on their technology level, including R&D activities, market conditions, product strategy, and corporate culture. Park et al. [18] used Total Quality Management (TQM) to evaluate the relationship between the implementation of quality management and performance evaluation in high-tech manufacturing firms. There are some suitable approaches to evaluate the performance in high technology firms, which involve internal or external environments [19]. Bowonder and Yadav [20] evaluated high technology firms by measuring R&D expenses which have been committed to improve technology activity. Deeds et al. [21] discussed the purpose of innovation through R&D activity to evaluate the performance of high technology firms.

3 Research Methods

DEA has been developed for 30 years since Charnes et al. [4] developed the CCR model. Up to now the DEA methods have been used in various applications including education, health care (hospitals, clinics), agricultural production, banking, armed forces, sports, etc. [12]. Further, at the same time, a lot of advanced models of DEA have been developed (e.g., cross-period data [22] and the multiple objective programming approach [23 ; 24]. In this research, the traditional CCR, BCC DEA models as well as the MOP based on DEA model will be applied to aggregate the

efficiency scores of the top information technology (IT) companies in S&P 500 based on the values as weights versus each input and output.

3.1 DEA

DEA is a non-parametric approach and doesn't need assumptions about the inputs and outputs. In 1957, Farrell first introduced how to deal with the problem of measuring the productive efficiency to both the economic theorist and the economic policy maker [25]. The first DEA model, a mathematical programming model by Charnes, Cooper, and Rhodes in 1978, was built to discuss the efficiency frontier by Farrell [4]. The CCR model assumes that production exhibits constant returns to scale. Then, in 1984, Banker, Charnes, and Cooper, extended the CCR model by assuming variable returns to scale and named the new model as the BCC model [26].

For company managers, controlling the range of inputs and decreasing inputs is easier than increasing the total sales. The CCR and BCC models of DEA are often used the input-oriented.

3.1.1 CCR

CCR–DEA model computes relative efficiency scores (h_i) based on selected s outputs $(r=1,...,s)$ and m inputs $(i=1,...,m)$ using the following linear programming expression [4 ; 24 ; 27]:

$$\begin{aligned} &\text{Max } h_i = \sum_{r=1}^{s} u_r Y_{rj} \Big/ \sum_{i=1}^{m} v_i X_{ij} \\ &\text{s.t. } \sum_{r=1}^{s} u_r Y_{rj} \Big/ \sum_{i=1}^{m} v_i X_{ij} \le 1 \\ &u_r, v_i \ge \varepsilon > 0;\ r=1,...s;\ i=1,...,m;\ j=1,...,n. \end{aligned} \tag{1}$$

In Eq. (1), it assumes the DMU has s outputs and m inputs, and there are n DMUs. The definition of X_{ij} is the ith input (i =1,2,…,m) and the jth DMU (j=1,2,…,m). The definition of Y_{rj} is the rth output (r =1,2,…,s) and the jth DMU (j=1,2,…,m). The u_r and v_i are not zero, calculating as $u_r, v_i \ge \varepsilon > 0$, ε is non-Archimedean number and is 10^{-6} in this paper.

3.1.2 BCC

Input-oriented BCC has a variable u_0 (returns to scale). The mathematical programming shows as follows [26]:

$$\begin{aligned} &\text{Max } h_i = \Big(\sum_{r=1}^{s} u_r Y_{rj} - u_0\Big) \Big/ \Big(\sum_{i=1}^{m} v_i X_{ij}\Big) \\ &\text{s.t. } \Big(\sum_{r=1}^{s} u_r Y_{rj} - u_0\Big) \Big/ \Big(\sum_{i=1}^{m} v_i X_{ij}\Big) \le 1, \\ &u_r, v_i \ge \varepsilon > 0;\ r=1,...s;\ i=1,...,m;\ j=1,...,n. \end{aligned} \tag{2}$$

Equation (2) was changed to (3) for solving formula as follows :

Assuming $v_i = (v_i / t),\ u_r = (u_r / t),\ t^{-1} = \sum_{i=1}^{m} v_i X_{ij}$, then

$$\begin{aligned}
&\text{Max } g_j = (\sum_{r=1}^{s} u_r Y_{rj} - u_0) \\
&\text{s.t. } \sum_{i=1}^{m} v_i X_{ij} = 1 \\
&(\sum_{r=1}^{s} u_r Y_{rj} - u_0) - (\sum_{i=1}^{m} v_i X_{ij}) - u_0 \le 0 \\
&u_r, v_i \ge \varepsilon > 0;\ r = 1,...s;\ i = 1,...,m;\ j = 1,...,n.
\end{aligned} \tag{3}$$

The dual formula:

$$\begin{aligned}
&\text{Min } z_j = \theta - \varepsilon(\sum_{i=1}^{m} s_i^- + \sum_{r=1}^{s} s_r^+) \\
&\text{s.t. } \sum_{j=1}^{n} \lambda_j X_{ij} - \theta X_{ij} + s_i^- = 0,\ i = 1,...,m; \\
&\sum_{j=1}^{n} \lambda_j Y_{rj} - s_i^- = Y_{rj}, r = 1,...,s; \\
&\sum_{j=1}^{n} \lambda_j = 1; \\
&\lambda_j, s_i^-, s_r^+ \ge 0;\ r = 1,...,m;\ j = 1,...,n.
\end{aligned} \tag{4}$$

3.2 MOP Based DEA

MOP based DEA method provides a unitary weight (u^*, v^*) for all DMUs, which are evaluated by an equal standard [23 ; 24 ; 28]. By this approach, this research can obtain the efficiency rating of each DMU more fairly. Moreover, all DMUs can be treated simultaneously, which makes it effective in handling large numbers of DMU.

Model 1

$$\begin{aligned}
&\text{Max } z_1 = \sum_{r=1}^{s} u_r y_{r1} / \sum_{i=1}^{m} v_i x_{i1} \\
&\text{Max } z_2 = \sum_{r=1}^{s} u_r y_{r2} / \sum_{i=1}^{m} v_i x_{i2} \\
&\quad \vdots
\end{aligned} \tag{5}$$

$$\text{Max } z_n = \sum_{r=1}^{s} u_r y_{rn} / \sum_{i=1}^{m} v_i x_{in}$$

$$s.t \sum_{r=1}^{s} u_r y_{rj} / \sum_{i=1}^{m} v_i x_{ij} \le 1, k = 1,2,...,n$$

$$u_r, v_i \ge \varepsilon > 0; 0 < \alpha \le 1$$

The definition of y_{rj} is the observed amount of output of rth (r =1,2,…,s) type for the jth DMU (j = 1,…, n). The x_{ij} is the observed amount of input of rth (r = 1,2,…,s) type for the jth DMU (j = 1,…, n). The v_i is the multiplier or weight of the ith input and the u_r is the multiplier or weight of the rth output. The ε is non-Archimedean quantity.

Then multiply the numerators and denominators in CCR model [4] was established the multiple objectives programming model and shown as Eq. (5). It was considerd by the efficiencies of all DMU and established a Multiple Objective Linear Fractional Programming (MOLFP) model, as shown in Model 1 (Eq. 5). According to the research of Sakawa and Yumine [29], Sakawa and Yano [30], and Ohta and Yamaguchi [31], the MOLFP problem can be solved by the Multiple Objective Linear Programming (MOLP) approach, as proposed by Zimmermann [32]. MOLP with DEA approach adopts to obtain common weights, which can maximize all DMU's efficiencies.

The concept of MOLP utilizes membership function transfers of multiple objective functions into one objective function. The membership function is as follows:

$$\mu(z_k) = \begin{cases} 0 & ; \quad z_k \le z_k^L \\ \dfrac{z_k - z_k^L}{z_k^R - z_k^L}; & z_k^L \le z_k \le z_k^R \\ 1 & ; \quad z_k \ge z_k^R \end{cases}$$

Where z_j^L and z_j^R are the negative ideal solution and the positive ideal solution, respectively, for the value of the objective function Zj, such that the degree of membership function is [0, 1]. The geometric view of the linear membership function is shown in Figure 1.

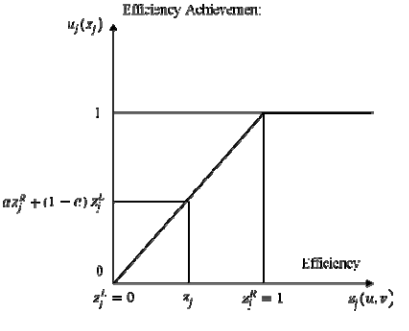


Fig. 1. Linear Membership Function of Zj Source: [24]

The degree of membership function of z_j^L in $\mu(Zj)$refers to the achievement level of the efficiency ratio for DMUk .The problem of obtaining the maximum decision is to choose (μ^*, v^*), such that

Model 2

$$\begin{aligned}
&\underset{\mu,v}{\text{Max}}\ \underset{j}{\text{Min}}\ u(z_j) \geq \alpha;\ j = 1,...,n \\
&\text{s.t. } (\sum_{r=1}^{s} u_r y_{rj}) / (\sum_{i=1}^{m} v_i x_{ij}) \leq 1;\ k = 1,2,...,n; \\
&u(z_j) \geq \alpha;\ j = 1,...,n.
\end{aligned} \tag{6}$$

Then, let the achievement level of the objective functions for Model 1 to be at a larger level, such as:

$$\alpha = (z_k - z_k^L) / (z_k^R - z_k^L) \tag{7}$$

Eq. (7) via variable transformation, has transformed $z_j = \alpha \cdot z_j^R + (1-\alpha) \cdot z_j^L$ where zj is a convex combination of z_j^L and z_j^R ; Eq. (6) can be rewritten as Eq. (8). According to the concept of multiple objective linear programming, we can determine a weight that satisfies all DMU restrictions. The weight (μ^*, v^*), is the common weight of all DMU, which are evaluated on a consistent standard of ranking.

$$\begin{aligned}
&\underset{\mu,v}{\text{Max}}\ \underset{j}{\text{Min}} \left\{ Z_j = (\sum_{r=1}^{s} \mu_r y_{rj}) / (\sum_{i=1}^{m} v_i x_{ij}) \right\} \\
&s.t. (\sum_{r=1}^{s} \mu_r y_{rk}) / (\sum_{i=1}^{m} v_i x_{ik}) \leq 1, \\
&(\sum_{r=1}^{s} \mu_r y_{rj}) / (\sum_{i=1}^{m} v_i x_{ij}) \geq \alpha \cdot z_j^R + (1-\alpha) \cdot z_j^L \\
&0 \leq \alpha \leq 1;\ u_r \geq \varepsilon > 0, r = 1,...,s;\ v_i \geq \varepsilon > 0, i = 1,...,m.
\end{aligned} \tag{8}$$

The efficiency approach measure (EAM) is Eq. (9).

$$\alpha_k = (\sum_{r=1}^{s} u_r^* \times y_{rk}) / (\sum_{i=1}^{m} v_i^* \times x_{ik}) \tag{9}$$

4 Empirical Study

By assuming the similarity of the firms may bias the results. When using the same standard to evaluate the performance of companies; nevertheless, the difference including products, life cycle etc. of the companies being as DMUs is so significant, the equal standard seems to be a big problem on this research. Thus, the author tried to select the

firms with equal character and field being as DMUs. The author chooses Communications Equipment companies in S&P 500 index as DMUs to evaluate the performance. These companies include Cisco Systems, F5 Networks, Juniper Networks, Motorola Mobility Holdings Inc., Motorola Solutions Inc., and QUALCOMM Inc..

Cost of revenue and R&D expense were selected as inputs while total revenue, Return on Investment (ROI) as well as net income growth were selected as outputs. Based on the CCR DEA, Cisco Systems, F5 Networks as well as Motorola Mobility Holdings Inc. are 100% efficient. On the other hand, by using the novel MOP approach, the F5 Networks still has achieved the optimum efficiency of 100%. (Please refer to Table 1 for the empirical study results.) This empirical study demonstrates that performance evaluation results based on CCR DEA or the novel MOP are totally different.

Table 1. Evaluation of Communications Equipment companies in S&P 500 index by CCR and MOP Based DEA

No.	DMU	CCR	Rank	MOP	Rank
1	Cisco	1.000	1	0.845	2
2	F5	1.000	1	1.000	1
3	Juniper	0.849	5	0.593	4
4	Motorola Mobility	1.000	1	0.592	5
5	Motorola Solutions	0.991	4	0.637	3
6	QUALCOMM	0.649	6	0.592	5

5 Discussion

In the following section, both theoretical advances of the MOP based DEA approach as well as managerial implications will be discussed. At first, the novel MOP based DEA approach, can really provide a totally different result based on the assumptions of fair weights versus each input and out. Furthermore, the successful introduction of the min-max approach resolved the DEA model successfully. Apparently, this novel MOP model can be considered as a better alternative to the traditional CCR DEA.

In the past, people seldom developed an evaluation model especially by using DEA for the communications equipment companies. Thus, this research defined a feasible framework for evaluating the communications equipment companies which can be leveraged as the basis for government policy and firm level investment, R&D and competitive strategy definitions. Further, the novel MOP based DEA approach demonstrated totally different results to the traditional CCR DEA. In the empirical study, the MOP based DEA demonstrated that F5 Networks should be the communications equipment companies to invest. Meanwhile, based on the MOP based DEA results, firms including Juniper Networks, Motorola Mobility Holdings Inc., Motorola Solutions Inc., and QUALCOMM Inc. are still far less than efficient. Appropriate adjustments of both inputs and outputs are required to enhance their competitiveness.

6 Conclusions

The information technology industry is one of the popular industries at the moment due to IT is starting to spread further from the conventional personal computer and network technologies to integrations of other fields of technology such as the use of cell phones,

televisions, automobiles, etc. Albeit important, very few researches tried to define an evaluation framework for IT especially in communications equipment companies. This research bridged the gap and introduced a novel MOP based DEA approach which overcame the shortage of the traditional CCR DEA model. Based on the evaluation results, F5 Networks is the most efficient communications equipment companies. The evaluation results can serve as the basis for investment strategy definitions.

References

1. Jorgenson, D.W.: Information Technology and the U.S. Economy. The American Economic Review 91(1), 1–32 (2001)
2. Phillips, P.A.: Performance measurement systems and hotels: a new conceptual framework. International Journal of Hospitality Management 18(2), 171–182 (1999)
3. Lewin, A.Y., Minton, J.W.: Determining organizational effectiveness: another look, and an agenda for research. Management Science 32(5), 514–538 (1986)
4. Charnes, A., Cooper, W., Rhodes, E.: Measuring the efficiency of decision making units. European Journal of Operational Research 2(6), 429–444 (1978)
5. Fare, R., Hunsaker, W.: Notions of efficiency and their reference sets. Management Science 32(2), 237–243 (1986)
6. Chiang, C.I., Tzeng, G.H.: A new efficiency measure for DEA: efficiency achievement measure established on fuzzy multiple objectives programming. Journal of Management 17(2), 369–388 (2003)
7. Chiang, C.I., Tzeng, G.H.: A multiple objective programming approach to data envelopment analysis. In: Shi, Y. (ed.) New Frontiers of Decision Making for the Information Technology Era. World Science, Hong Kong (2000)
8. Helms, M.M.: Encyclopaedia of Management, 5th edn. Thomson Gale (2006)
9. Sumanth, D.J.: Productivity engineering and management. McGraw-Hill, New York (1994)
10. Tangen, S.: Demystifying productivity and performance. International Journal of Productivity and Performance Management 54(1), 34–46 (2005)
11. Coelli, T., Prasada Rao, D.S., O'Donnell, C.J., Battese, G.E.: An introduction to efficiency and productivity analysis. Springer, New York (2005)
12. Emrouznejad, A.: Data Envelopment Analysis Homepage (1995), http://www.DEAzone.com
13. Harbour, J.L.: The performance Paradox: understanding the real drivers that critically affect outcomes. Productivity Press Taylor & Francis Group (2009)
14. Collin, S.M.H.: Dictionary of ICT, 4th edn. Bloomsbury Publishing Plc. (2004)
15. Bruton, N.: Managing the IT services process, p. 95. Butterworth Heinemann (2004)
16. Harry, P.: Performance Measurement Principles and Techniques: An Overview for Local Government. Public Productivity Review 4(4), 312–339 (1980)
17. Grinstein, A., Goldman, A.: Characterizing the technology firm: An exploratory study. Research Policy 35, 121–143 (2006)
18. Park, S., Hartley, J., Wilson, D.: Quality management practices and their relationship to buyers' supplier ratings: A study in the Korean automotive industry. Journal of Operations Management 268, 1–18 (2001)
19. Cardy, R.L., Dobbins, R.H.: Human resources, high technology, and a quality organizational environment: Research agendas. The Journal of High Technology Management Research 6(2), 261–279 (1995)

20. Bowonder, B.Y., Yadav, S.: R&D spending patterns of global firms. Research Technology Management 42(6), 44–55 (1999)
21. Deeds, D.L.: The role of R&D intensity, technical development and absorptive capacity in creating entrepreneurial wealth in high technology startups. Journal of Engineering and Technology Management 18(1), 29–47 (2001)
22. Fare, R., Grabowaski, R., Grosskopf, S.: Technical efficiency of Philippine agriculture. Applied Economics 17(2), 205–214 (1985)
23. Chiang, C.I., Tzeng, G.H.: A New Efficiency Measure for DEA: Efficiency Achievement Measure Established on Fuzzy Multiple Objectives Programming. Journal of Management 17(2), 369–388 (2000a)
24. Chiang, C.I., Tzeng, G.H.: A multiple objective programming approach to data envelopment analysis. In: Shi, Y., Zeleny, M. (eds.) New Frontiers of Decision Making for the Information Technology Era, pp. 270–285. World Science Publishing Company, Singapore (2000b)
25. Farrell, M.J.: The measurement of productive efficiency. Journal of the Royal Statistical Society 120(3), 253–291 (1957)
26. Banker, R.D., Charnes, A., Cooper, W.W.: Some models for estimating technical and scale inefficiencies in data envelopment analysis. Management Science 30(9), 1078–1092 (1984)
27. Charnes, A., Clark, T., Cooper, W.W., Golany, B.: A developmental study of data envelopment analysis in measuring the efficiency of maintenance units in the US Air Force. Annals of Operations Research 2(1), 95–112 (1985)
28. Yu, J.R., Tzeng, Y.C., Tzeng, G.H., Yu, Z.Y., Sheu, H.J.: A fuzzy multiple objective programming to DEA with imprecise data. International Journal of Uncertainty. Fuzziness & Knowledge-Based Systems 12(5), 591–600 (2004)
29. Sakawa, M., Yumine, T.: Interactive fuzzy decision-making for multi-objective linear fractional programming problems. Large Scale Systems 5, 105–114 (1983)
30. Sakawa, M., Yano, H.: Interactive decision making for multi- objective linear fractional programming problems with parameters. Cybernetics and Systems: An International Journal 16, 377–394 (1985)
31. Ohta, H., Yamaguchi, T.: Multi-goal programming including fractional goal in consideration of fuzzy solutions. Journal of Japan Society for Fuzzy Theory and System 7, 1221–1228 (1995)
32. Zimmermann, H.J.: Fuzzy programming and linear programming with several objective functions. Fuzzy. Sets and Systems 1(1), 45–55 (1978)

Using the DEMATEL Based Network Process and Structural Equation Modeling Methods for Deriving Factors Influencing the Acceptance of Smart Phone Operation Systems

Chi-Yo Huang[1], Ming-Jenn Wu[1], Yu-Wei Liu[1], and Gwo-Hshiung Tzeng[2,3]

[1] Department of Industrial Education, National Taiwan Normal University
No. 162, Hoping East Road I, Taipei 106, Taiwan
cyhuang66@ntnu.edu.tw
[2] Department of Business and Entrepreneurial Administration, Kainan University
No. 1, Kainan Road, Luchu, Taoyuan County 338, Taiwan
[3] Institute of Management of Technology, National Chiao Tung University
Ta-Hsuch Road, Hsinchu 300, Taiwan
ghtzeng@mail.knu.edu.tw

Abstract. Smartphones have emerged as one of the most important consumer-electronic products during the past years. Therefore, researches on factors influencing consumers' behavior and thus, repurchase intention toward smartphones become the most critical issue for smartphone marketers. The smartphone operation system (OS) is one of the major factors influencing consumers' purchase decisions toward purchasing smartphones. However, the analysis and predictions of consumer behaviors toward the smartphone OSs are not easy due to due to the fast emerging technology and highly competitive market situation. To resolve this problem, this research aims to propose a novel multiple criteria decision making (MCDM) based approach for discovering the factors influencing the technology acceptances of the smartphone OSs based on industry experts' opinions. The opinions of mass users will also be summarized by using the Structural Equation Modeling (SEM) based Technology Acceptance Model (TAM) for comparisons. Differences of the analytic results being derived by the two analytic frameworks will be compared. Both the analytic framework and results can serve as the basis for future smartphone marketers' uses for strategy definitions.

Keywords: Decision Making Trial and Evaluation Laboratory (DEMATEL), Technology Acceptance Model (TAM), Structural Equation Modeling (SEM), Lead User Theory.

1 Introduction

Smartphones meet the consumers' need for powerful cell phones as the use of cell phone grows worldwide [1]. A smartphone can be seen as a ubiquitous computing platform [2; 3], as well as a hybrid of mobile phones and PDAs (Personal Digital Assistant) [3; 4]. They are either proprietary for special devices (e.g. Blackberry and

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 731–741, 2012.

Mac OS X) or open to all independent software providers (e.g. Symbian, Windows Mobile, Palm OS and Linux). Due to the daily increased computation requirements being caused by technology push or market drive applications (APPs), the operation systems (OSs) being used in smart phones should be improved for fast changing mobile computation architectures [5]. Such demands for better smartphone OSs have changed the nature of the marketplace. And that marketplace is healthy, despite the global economic problems, largely because smart phone sales are increasing [5].

Recently, various embedded OSs, e.g., Apple iPhone OSX, Google Android, Nokia Symbian, Microsoft Window Mobile, Blackberry OS, Palm OS, etc., are competing for modern mobile applications [6]. However, it is a difficult task for designer and marketers to design and promote the OSs when concerning the complexity of human behavior [5]. Besides, high-technology products like the embedded OSs are more complicated than traditional ones. Predictions for consumers' preferences and satisfactions for such products are comparatively more difficult.

In order to resolve the above-mentioned prediction problems, this research applied the Decision Making Trial and Evaluation Laboratory (DEMATEL) based network process (DNP) with the Lead User Method (LUM) to collect information for consumer needs from the leading edge of the market [7]. The LUM is more suitable for evaluating the disruptive or radical innovations [7] when the availability of experts is very limited. Meanwhile, mass users usually follow the lead users of a novel product during the mature period of a market in general [8]. The Technology Acceptance Model (TAM) which can explain users' usage intentions and behaviors towards to the high technology products with high accuracy [9] will also be introduced as the theoretic background of the analytic framework. To demonstrate the differences between results being derived by the multiple criteria decision making (MCDM) based LUM framework, the results being derived from questionnaires being surveyed from consumers by using the statistical structural equation modeling (SEM) will also be demonstrated. Finally, the analytic result based on experts' and mass users' perspectives will be compared and discussed. An empirical study based on Taiwanese embedded OS experts as well as smartphone consumers from the northern part of Taiwan will be introduced for demonstrating the feasibility of the proposed framework.

Following of the research will be organized as follows. The related literature regarding to technology acceptance theories and TAM model will be reviewed in Section 2. The analytic framework based on the DNP and SEM methods will be introduced in Section 3. Then, in Section 4, an empirical study follows, demonstrating the feasibility of the analytic framework by using the proposed DNP and SEM based TAM frameworks. Managerial implications as well as discussion will be presented in Section 5. Finally, the whole article will be concluded in Section 6.

2 The Concept and Theories of Human Behavior

In this section, those well-known consumer behavior analysis models such as TRA, TAM and Lead-user theory which have been applied in many fields will be brief reviewed.

2.1 Theory of Reasoned Action (TRA)

TRA is a prediction model, which attempts to figure out the requirements of intended behaviors regarding to the acceptance of users. According to Fishbein and Ajzen (1980), user's specified behavior will be determined and influenced by users' behavioral intentions. The behavioral intention is decided by the person's attitude and subjective norm [10; 11; 12].

According to TRA theory, users' attitude toward the acceptance of a product is decided by user's prominent beliefs about results of accept the product and the evaluation of the acceptance. Beliefs are the uses' subjective probability of product acceptance and usage [10; 12].

2.2 Technology Acceptance Model (TAM)

The Technology Acceptance Model (TAM) was proposed by Davis especially for predicting the individual adoption and use of new information technologies [11]. To better predict, explain, and increase user acceptance, Davis bends his effort to understand why people accept or reject computers [11; 14]. The research of TAM addresses the ability to predict peoples' computer acceptance from a measure of their intentions, and the ability to explain their intentions in terms of their attitudes, subjective norms, perceived usefulness, perceived ease of use, and related variables [11; 14].

TAM posits that two particular beliefs, perceived usefulness and perceived ease of use, are of primary relevance for computer acceptance behaviors, which is shown in figure 2. Perceived usefulness (U) is defined as the prospective user's subjective probability that using a specific application system will increase his or her job performance within an organizational context [14]. Perceived ease of use (PEU) refers to the degree to which the prospective user expects the target system to be free of effort [11].

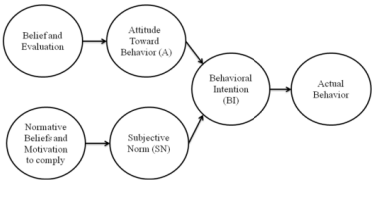


Fig. 1. TRA (Source: [13])

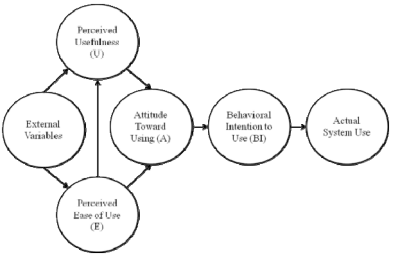


Fig. 2. TAM (Source: [14])

The direct effect of TAM isn't as same as the TRA theory, but the TAM model provides theoretical justification and empirical evidence of direct effect towards to intention [12; 15; 16; 17].

2.3 Lead User Theory

The lead user theory is an approach which was originally proposed to selectively identify commercially attractive innovations developed by users [10]. Lead users are defined

as members of a user population who (1) anticipate obtaining relatively high benefits from obtaining a solution to their needs and so may innovate and (2) are at the leading edge of important trends in a marketplace under study and so are currently experiencing needs that will later be experienced by many users in that marketplace [11; 17].

In a previous paper, von Hippel (1986) has proposed that analysis of need and solution data from "lead users" can improve the productivity of new product development in fields characterized by rapid change. Consequently, the LUM, a market analysis technique, is applied to the development of new products and services [18]. The methodology is composed of four major steps based on the work of Urban and Hippel (1988): (1) specify lead user indicators, (2) identify lead user group, (3) generate concept (product) with lead users and (4) test lead user concept (product). Further details can be found in the earlier work by Urban and Hippel (1988).

3 Research Methods

In order to build the analytical framework for comparing the smart phone operation system acceptance versus lead users and mass users are initiated. This research applies SEM to evaluate the determinants of mass users' acceptance. Furthermore, the correlation coefficient versus each criterion is derived by SEM from mass users' perspectives. Meanwhile, the determinants and causal relationship of lead users' acceptance are identified by using DNP method. Finally, the result from mass users and lead users will be compared.

3.1 Structural Equation Modeling (SEM)

Jöreskog and Sörbom [19] developed the analysis skill of the matrix so as to deal the analyzing problems of covariance structure. Because LISREL is very similar with covariance structure models, early scholar named covariance structure models as LISREL model. Henceforth, scholars proposed some software one after another, which can be divided as two main types. One is based on components such as PLSPATH while another is based on covariance such as LISREL, EQS, AMOS, MPLUS, CALLS and RAMONA.

SEM technique deals with relations of multiple criteria constructs simultaneously and fits in proving positive research. The primary aim of SEM technique is the analysis of latent variables and the analysis of causal relations between latent constructs to verify theory so would be called causal model technique.

The Structural Equation Modeling (SEM) Methodology is a confirmatory modeling for data analysis; therefore, researchers must have a theoretical foundation for their proposed research models which are guided by theories. No matter it is to prove any causal relationships or confirm the internal structure, both depend on clarifying the contents and the properties of prior research variables, and a clear description of hypothetical relations.

Moreover, researchers advance the concrete structural hypothetical relations and seek for statistical confirmation. The investigation of the variable structural relations

in the areas of sociological and behavioral science mainly consists of a group of indirectly observed, measured abstractly latent constructs. Precise statistical data is required to prove the existence of the construct, which is one of the major advantages of SEM methodology [20].

3.2 Decision Making Trial and Evaluation Laboratory (DEMATEL) Based Network Process

The DNP, a decision making process which consists of the DEMATEL as well as the network process for deriving the impact of each criterion on others as the weight, was developed by Prof. Tzeng [21].

The DEMATEL technique was developed by the Battelle Geneva Institute: (1) to analyze complex "real world problems" dealing mainly with interactive map-model techniques [22]; and (2) to evaluate qualitative and factor-linked aspects of societal problems [21].

DEMATEL technique was developed with the belief that the pioneering and proper use of scientific research methods could help to illuminate specific and intertwined phenomena and contribute to the recognition of practical solutions through a hierarchical structure [21]. DEMATEL has been successfully applied in many situations such as marketing strategies, safety problems and environment watershed plans [23; 24; 26]. The ANP is a general form of the analytic hierarchy process (AHP) [27] which has been used in multi criteria decision making (MCDM) to can release the restriction of hierarchical structure [21].

Combing the DEMATEL and ANP method, the process and equations for the DNP method will be demonstrated below based on the work by Chiu, Tzeng and Li [21], the steps of the DNP method can be summarized as follows,

Step 1: Calculate the direct-influence matrix by scores. Based on experts' opinions, evaluations are made of the relationships among elements (or variables/ attributes) of mutual influence using a scale ranging from 0 to 4, with scores representing "no influence" (0), "low influence" (1), "medium influence" (2), "high influence" (3), and "very high influence" (4). They are asked to indicate the direct effect they believe a factor i will have on factor, as indicated by d_{ij}. The matrix D of direct relations can be obtained.

Step 2: Normalize the direct-influence matrix based on the direct-influence matrix D, the normalized direct relation matrix X is acquired by using Eq. (1)

$$N = vD; v = \min\{1/\max_i \sum_{j=1}^{n} d_{ij}, 1/\max_j \sum_{i=1}^{n} d_{ij}\}, i, j \in \{1,2,...,n\} \tag{1}$$

Step 3: Attaining the total-influence matrix $\boldsymbol{T}$. Once the normaliz j ed direct-influence matrix $\boldsymbol{N}$ is obtained, the total-influence matrix $\boldsymbol{T}$ of NRM can be obtained.

$$\boldsymbol{T} = \boldsymbol{N} + \boldsymbol{N}^2 + ... + \boldsymbol{N}^k = \boldsymbol{N}(\boldsymbol{I} - \boldsymbol{N})^{-1} \tag{2}$$

where $k \to \infty$ and $\boldsymbol{T}$ is a total influence-related matrix; $\boldsymbol{N}$ is a direct influence matrix and $\boldsymbol{N} = [x_{ij}]_{n \times n}$; $\lim_{k \to \infty} \left(\boldsymbol{N}^2 + \cdots + \boldsymbol{N}^k \right)$ stands for a indirect influence matrix

and $0 \le \sum_{j=1}^{n} x_{ij} < 1$ or $0 \le \sum_{i=1}^{n} x_{ij} < 1$, and only one $\sum_{j=1}^{n} x_{ij}$ or $\sum_{i=1}^{n} x_{ij}$ equal to 1 for $\forall i, j$.

So $\lim_{k\to\infty} \boldsymbol{N}^k = [\boldsymbol{0}]_{n\times n}$. The (i, j) element t_{ij} of matrix $\boldsymbol{T}$ denotes the direct and indirect influences of factor i on factor j.

Step 4: Analyze the result. In this stage, the row and column sums are separately denoted as $\boldsymbol{r}$ and $\boldsymbol{c}$ within the total-relation matrix $\boldsymbol{T}$ through Equations (3), (4), and (5).

$$\boldsymbol{T} = [t_{ij}], \quad i, j \in \{1,2,...,n\} \tag{3}$$

$$\boldsymbol{r} = [r_i]_{n\times 1} = \left[\sum_{j=1}^{n} t_{ij}\right]_{n\times 1} \tag{4}$$

$$\boldsymbol{c} = [c_j]_{1\times n} = \left[\sum_{i=1}^{n} t_{ij}\right]_{1\times n} \tag{5}$$

where the $\boldsymbol{r}$ and $\boldsymbol{c}$ vectors denote the sums of the rows and columns, respectively. Suppose r_i denotes the row sum of the ith row of matrix $\boldsymbol{T}$. Then, r_i is the sum of the influences dispatching from factor i to the other factors, both directly and indirectly. Suppose that c_j denotes the column sum of the jth column of matrix $\boldsymbol{T}$. Then, c_j is the sum of the influences that factor i is receiving from the other factors. Furthermore, when $i = j$ (i.e., the sum of the row sum and the column sum) $(r_i + c_i)$ represents the index representing the strength of the influence, both dispatching and receiving), $(r_i + c_i)$ is the degree of the central role that factor i plays in the problem. If $(r_i - c_i)$ is positive, then factor i primarily is dispatching influence upon the strength of other factors; and if $(r_i - c_i)$ is negative, then factor i primarily is receiving influence from other factors [23; 25]. Therefore, a causal graph can be achieved by mapping the dataset of $(r_i + s_i, r_i - s_i)$ providing a valuable approach for decision making [20].

Now the total-influence matrix is called as $\boldsymbol{T}_C = \left[t_{ij}\right]_{nxn}$, which is obtained by criteria and $\boldsymbol{T}_D = \left[t_{ij}^{D}\right]_{nxn}$ obtained by dimensions (clusters) from $\boldsymbol{T}_C$. Then we normalize the ANP weights of dimensions (clusters) by using influence matrix $\boldsymbol{T}_D$.

$$\boldsymbol{T}_D = \begin{bmatrix} t_{11}^{D_{11}} & \cdots & t_{1j}^{D_{1j}} & \cdots & t_{1m}^{D_{1m}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{i1}^{D_{i1}} & \cdots & t_{ij}^{D_{ij}} & \cdots & t_{im}^{D_{im}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{m1}^{D_{m1}} & \cdots & t_{mj}^{D_{mj}} & \cdots & t_{mm}^{D_{mm}} \end{bmatrix} \begin{matrix} \\ \\ \longrightarrow d_i = \sum_{j=1}^{m} t_{ij}^{D_{ij}} \\ \\ \\ \longrightarrow d_m = \sum_{j=1}^{m} t_{mj}^{D_{mj}} \end{matrix} \quad d_1 = \sum_{j=1}^{m} t_{1j}^{D_{1j}}, \; d_i = \sum_{j=1}^{m} t_{ij}^{D_{ij}}, \; i = 1,...,m$$

Step 5: The original supermatrix of eigenvectors is obtained from the total-influence matrix $\boldsymbol{T} = [t_{ij}]$. For example, $\boldsymbol{D}$ values of the clusters in matrix $\boldsymbol{T}_D$, as Eq.(6). Where if $t_{ij} < \boldsymbol{D}$, then $t_{ij}^{D} = 0$ else $t_{ij}^{D} = t_{ij}$, and t_{ij} is in the total-influence matrix $\boldsymbol{T}$. The total-influence matrix $\boldsymbol{T}_D$ needs to be normalized by dividing by the following formula. There, we could normalize the total-influence matrix and represent it as $\boldsymbol{T}_D$.

$$\boldsymbol{T}_D = \begin{bmatrix} t_{11}^{D_{11}}/d_1 & \cdots & t_{1j}^{D_{1j}}/d_1 & \cdots & t_{1m}^{D_{1m}}/d_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{i1}^{D_{i1}}/d_i & \cdots & t_{ij}^{D_{ij}}/d_i & \cdots & t_{im}^{D_{im}}/d_i \\ \vdots & \vdots & & \vdots & \vdots \\ t_{m1}^{D_{m1}}/d_m & \cdots & t_{mj}^{D_{mj}}/d_m & \cdots & t_{mm}^{D_{mm}}/d_m \end{bmatrix} = \begin{bmatrix} \alpha_{11}^{D_{11}} & \cdots & \alpha_{1j}^{D_{1j}} & \cdots & \alpha_{1m}^{D_{1m}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_{i1}^{D_{i1}} & \cdots & \alpha_{ij}^{D_{ij}} & \cdots & \alpha_{im}^{D_{im}} \\ \vdots & \vdots & & \vdots & \vdots \\ \alpha_{m1}^{D_{m1}} & \cdots & \alpha_{mj}^{D_{mj}} & \cdots & \alpha_{mm}^{D_{mm}} \end{bmatrix} \quad \text{where}$$

$\alpha_{ij}^{D_{ij}} = t_{ij}^{D_{ij}}/d_i$.

This research adopts the normalized total-influence matrix $\boldsymbol{T}_D$ (here after abbreviated to "the normalized matrix") and the unweighted supermatrix $\boldsymbol{W}$ using Eq. (7) shows theses influence level values as the basis of the normalization for determining the weighted supermatrix.

$$\boldsymbol{W}^{*} = \begin{bmatrix} \alpha_{11}^{D_{11}} \times \boldsymbol{W}_{11} & \alpha_{21}^{D_{21}} \times \boldsymbol{W}_{12} & \cdots & \cdots & \alpha_{m1}^{D_{m1}} \times \boldsymbol{W}_{1m} \\ \alpha_{12}^{D_{12}} \times \boldsymbol{W}_{21} & \alpha_{22}^{D_{22}} \times \boldsymbol{W}_{22} & \cdots & \cdots & \vdots \\ \vdots & \cdots & \alpha_{ji}^{D_{ji}} \times \boldsymbol{W}_{ij} & \cdots & \alpha_{mi}^{D_{mi}} \times \boldsymbol{W}_{im} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_{1m}^{D_{1m}} \times \boldsymbol{W}_{m1} & \alpha_{2m}^{D_{2m}} \times \boldsymbol{W}_{m2} & \cdots & \cdots & \alpha_{mm}^{D_{mm}} \times \boldsymbol{W}_{mm} \end{bmatrix} \tag{7}$$

Step 6: Limit the weighted supermatrix by raising it to a sufficiently large power k, as Eq. (8), until the supermatrix has converged and become a long-term stable supermatrix to get the global priority vectors or called ANP weights.

$$\lim_{k \to \infty} (\boldsymbol{W}^{*})^{k} \tag{8}$$

4 Empirical Study

The element of users' acceptance will be summarized in the sub-section 4-1. Then, an empirical study on users' acceptance will be introduced for verifying the feasibility of the proposed analytic framework.

4.1 Background of Smartphone OSs

The key feature of Smart Phone is the operation system that resides on the devices, which can connect with hardware and applications [28]. According to Hill [28], Motion (RIM), with its Blackberry phones; Apple with iPhone; Nokia, which owns the

Symbian operating system for Smart Phones; Microsoft, with its Windows mobile phones; and Google, which owns the Android operating system for smart phone are the main competitors in smartphone OSs' market. Microsoft and Google make just the operating system and partner with various hardware manufactures (e.g. HTC) to sell the phone to end users while RIM, Apple and Nokia integrate the phone and operating system together and sell the integrated bundle. The market share of operating system is demonstrated in Table 1, it's clearly that the market share of Android and iOS system are increasing [28].

Table 1. Smart Phone Operation System Market Share (Source: [29])

Year	iOS	Android	RIM	Symbian	Microsoft	Others
2011Q1	16.8%	36.0%	12.9%	27.4%	3.6%	3.3%
2010	15.7%	22.7%	16.0%	37.6%	4.2%	3.8%
2009	14.4%	3.9%	19.9%	46.9%	8.7%	6.1%
2008	8.2%	5.0%	16.6%	52.4%	11.8%	10.5%
2007	2.7%	N/A	9.6%	63.5%	12.0%	12.1%

4.2 Empirical Study of Lead User Based on DNP Method

This research invited 7 Taiwanese lead users and 154 mass users to fill the questionnaire first. Then, the DNP method was applied to evaluating the acceptance of lead users; the mass users' acceptance of mass users will be evaluated by SEM method. At first, seven smart phone lead users were invited to evaluate the questionnaire and the casual relationship was derived by DEMATEL method is demonstrated in Figure 3. The causal relationships can be derived by using the DEMATEL method by setting the threshold value as 1.286. Further, the weights of each criterion calculated by ANP method are demonstrated in Table 2. According to the result, PU, PEU and B are important criteria for lead users.

Table 2. The Weight of Each Criteria

Criteria	PU	PEU	ATT	BI	SN	PBC	B
Weight	0.149	0.152	0.142	0.145	0.114	0.138	0.158
Rank	3	2	5	4	7	6	1

4.3 Empirical Study of Mass User Based on SEM Method

This research regards subject norms and perceived behavioral controls as external variables in TAM theories. In a good fit model, the p-value should be greater than 0.5, the RMSEA should be smaller than 0.05. Further, this research attempts to delete subject norms and perceived behavioral controls in order to have a good fit model. The path coefficients between the criteria with SEM method are shown in Table 3 and Figure 4. The empirical study figured out that the attitudes and behavioral intention significantly influence the actual behavior.

Table 3. The Path Coefficient between Each Criterion

Path	PEU→PU	PEU→ATT	PU→ATT	ATT→BI	BI→B
Coefficient	0.51	0.23	0.47	0.77	1.51

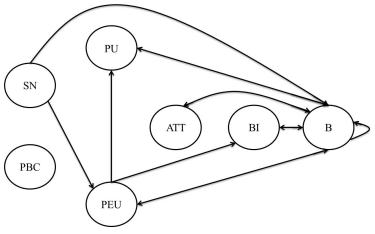


Fig. 3. The Casual Relationship

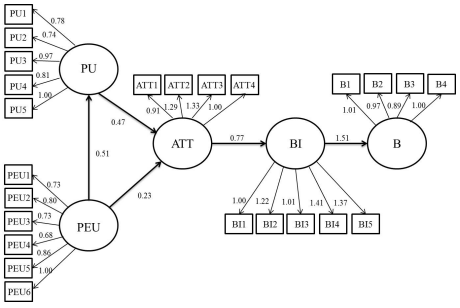


Fig. 4. The Path Coefficients of the TAM

5 Discussion

In order to establish a feasible analytic procedure for factors influencing technology acceptance of smartphone OSs by consumers and compare the differences between the influential factors on consumer behaviors of lead users and those of mass customers, an analytic framework was defined and verified. From the perspectives of mass users, PEU will influence PU; PU and PEU will influence ATT; ATT will influence BI; finally, BI will influence B. From the perspectives of lead users, PU will influence B; PEU will influence PU, BI and B; ATT will be influenced by B; BI will influence B; SN will influence PEU and B; finally, B will also influence itself. It is obviously that the casual relationship of criteria is different between lead users and mass users. Besides, subjective norm and perceived behavioral control are not suitable to predict the acceptance of smart phone operation system.

Regarding to the importance of acceptance factor, attitudes and behavioral intention influence other factors mostly for mass users. However, actual behavior and perceived ease of use are significant factors for lead users. Consequently, behavioral intention will influence the actual acceptance for both mass users and lead users.

6 Conclusions

The smartphone OS has become a key factor for influencing smartphone consumer behaviors. However, the prediction of smartphone OS is a difficult due to various factors influencing the fast emergence of technology and highly competitive market situation. This research proposed a DNP based analytic framework for predicting the factors influencing technology acceptance of smartphone OS based on lead users' opinions. Mass customers' opinions were surveyed by the traditional SEM while analytic results were compared. Based on the research results, the attitude and behavioral intention are critical factors for influencing technology acceptance of mass users. On the other hand, actual behavior and perceived ease of use are critical influential factors for lead users. Finally, the analytical framework being derived and verified in this research can be applied to the embedded OS development for other high technology product developing strategies.

References

1. Lin, F., Ye, W.: Operating System Battle in the Ecosystem of Smartphone Industry. In: International Symposium on Information Engineering and Electronic Commerce, pp. 617–621 (2009)
2. Abowd, G.D., Iftode, L., Mitchell, H., Tech, G.: The Smart Phone: A First Platform for Pervasive Computing. Pervasive Computing 4(2), 18–19 (2005)
3. Marcus, A., Chen, E.: Designing the PDA of the future. Interactions 9(1), 34–44 (2002)
4. Anderson, P., Blackwood, A.: Mobile and PDA technologies and their future use in education. JISC Technology and Standards Watch, Bristol, UK (2004)
5. Ajzen, I.: Attitudes, personality, and behavior. Dorsey Press, Chicago (1988)
6. Chang, Y.F., Chen, C.S., Zhou, H.: Smart phone for mobile commerce. Computer Standards & Interfaces 31(4), 740–747 (2009)
7. Mohr, J., Sengupta, S., Slater, S.: Marketing of High-Technology Products and Innovations. Prentice-Hall, Upper Saddle River (2010)
8. Rogers, E.M., Shoemaker, E.F.: Communication of Innovations: A Cross-Cultural Approach. Free Press, New York (1971)
9. Legris, P., Ingham, J., Collerette, P.: Why do people use information technology? A critical review of the acceptance model. Information and Management 40(3), 191–204 (2003)
10. Ajzen, I., Fishbein, M.: Understanding attitudes and predicting social behavior. Prentice-Hall, Englewood Cliffs (1980)
11. Hippel, E.V.: Lead User: A source of novel product concepts. Management Science 32(7), 791–805 (1986)
12. Tamura, H., Akazawa, K., Nagata, H.: Structural modeling of uneasy factors for creating safe, secure and reliable society. In: SICE System Integration Division Annual Conference, pp. 330–340 (2002)
13. Ajzen, I.: From intentions to actions: A theory of planned behavior, United State (1985)
14. Davis, F.D.: A Technology Acceptance Model for Empirically Testing New End-User Information Systems: Theory and Results. Doctoral dissertation, Sloan School of Management, Massachusetts Institute of Technology (1986)
15. Brinberg, D.: An Examination of the Determinants of Intention and Behavior: A Comparison of Two Models. Social Psychology 9(6), 560–575 (1979)
16. Triandis, H.C.: Interpersonal behavior. Brooks/Cole, Monterey (1977)
17. Franke, N., von Hippel, E., Schreier, M.: Finding Commercially Attractive User Innovations: A Test of Lead-User Theory. Journal of Product Innovation Management 23(4), 301–315 (2006)
18. Urban, G., von Hippel, E.: Lead User Analyses for the Development of New Industrial Products. Management Science 35(5), 569–582 (1988)
19. Joreskog, K.G., Sorbom, D.: LISREL VI: A general computer program for estimation of a linear structural equation system by maximum likelihood methods. National Educational Resources, Chicago (1981)
20. Lin, Y.F.: Prediction of the Acceptance Model of Technology Product by Using the DEMATEL based Network Process, Unpublished Master Thesis, National Taiwan Normal University, Taipei, Taiwan (2011)
21. Chiu, W.Y., Tzeng, G.H., Li, H.L.: Advances in Intelligent Decision Technologies: Proceedings of the Second KES International Symposium IDT 2010. Scientific Publishing Services Pvt. Ltd., Chennai (2010)
22. Gabus, A., Fontela, E.: World problems, an invitation to further thought within the framework of DEMATEL, Battelle Institute, Geneva, Switzerland (1972)

23. Liou, J.J.H., Tzeng, G.H., Chang, H.C.: Airline safety measurement using a hybrid model. Journal of Air Transport Management 13(4), 243–249 (2007)
24. Lin, C.L., Hsieh, M.S., Tzeng, G.H.: Evaluating vehicle telematics system by using a novel MCDM techniques with dependence and feedback. Expert System with Application 37(10), 6723–6736 (2010)
25. Huang, C.Y., Tzeng, G.H.: Reconfiguring the Innovation Policy Portfolios for Taiwan's SIP Mall Industry. Technovation 27(12), 744–765 (2007)
26. Chen, Y.C., Lien, H.P., Tzeng, G.H.: Measures and Evaluation foe environment watershed plans using a novel hybrid MCDM model. Expert System with Applications 37(2), 926–938 (2010)
27. Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)
28. Hill, J.: Theory of Strategic Management. Cengage Learning, USA (2010)
29. Reisinger, D.: Gartner: Android leads, Windows Phone lags in Q1. Gartner, Inc., http://news.cnet.com/8301-13506_3-20064223-17.html (retrieved)

Determine the Service Value Position for Package Tour Plan Based on a Novel MCDM Approach

Chia-Li Lin

Department of Resort and Leisure Management,
Taiwan Hospitality & Tourism College, No. 268, Chung-Hsing St. Feng-Shan Village,
Shou-Feng Township, Hualien County, 974, Taiwan, ROC
linchiali0704@yahoo.com.tw

Abstract. In early years, customers' preferences were not taken into account for planning of package tours, and instead the price was considered as the top priority. Customers could have limited alternatives based on only few packages plans and prices. However, the enterprises of package tour have faced the difficult challenge of what are the package tours the tourist wants. This research provides a service evaluation model for package tours. The model could be applied to satisfy tourists' needs and to provide mass customized services by value oriented operation instead of traditional price ones. In addition, this research proposes six package tours which applies the guideline of travel mode choices and divides the regional package tours into three transportation tools (bus tour, railroad tour, and private vehicle tour) and four tour modes (Shopping and cuisine, nature experience, culture exploring, and specific events) in the Nantou area of Taiwan.

Keywords: Package tour plan, DEMATEL, Principal component analysis (PCA), Analytic network process (ANP), VIKOR.

1 Introduction

Leisure is emphasized by people in modern society, and the policy of weekly 2-day break provides more chances for people to visit the nature. Therefore, recreation areas or spectacular scenery areas are always crowded during weekends and holidays. Without raising the service quality for the increasing amount of tourism demands, the recreation spots are threatened by the over-crowded problem. In addition, tourism operators commonly apply the low-price strategies. Those make the homogenous package tours are over supplied on the market. The study considers two approaches to solve this issue. Firstly, package contents shall be differentiated to improve service quality by diversifying package plans. Secondly, the large amount of customized products shall be introduced to decrease costs. In the research, the Decision-making trial and evaluation laboratory (DEMATEL) is used to construct the criteria relation structure, the Principal Component Analysis (PCA) is applied to categorize criteria based on their properties, the Analytic Network Process (ANP) is applied to define

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 742–753, 2012.

the magnitude relation among the criteria, and the VIKOR (Vlsekriterijumska Optimizacija I Kompromisno Resenje) is adopted to evaluate and resolve the problem of mutual competition among criteria and issues of alternative prioritization. Nantou area in Taiwan is nominated for case study. This research applies the guideline of travel mode choices and divides the regional package tours into 3 types, such as bus tour, railroad tour, and private vehicle tour. Mixing the tour mode with shopping and cuisine, nature experience, culture exploring, and event participation, and reorganizes 6 package tours. VSI and PSI are applied to evaluate those package tours, and preferences of different categories of tourists. The railroad tour has the highest VSI while the traditional culture experiencing tour has the highest PSI. The traditional culture experiencing tour with lower costs is also favored by general people and lovers of traditional arts. The railroad exploration tour focusing on railway history and culture is unique under market segmentation and promoted by the cooperation between government and the private agent. However, packages of natural scenery and enjoyment of hot spring, such as Lake and Sky Tour in Sun Moon Lake, Sea of Clouds Tour in Cing Jing Farm, and Spring enjoying tour in Dong Pu Hot Spring are declining in competitiveness due to price and uniqueness factors. The Ha Kone Tour focusing on museum visiting is the least favored one. Therefore, affording customized combination of various types of tours and keep concentrating on major customers will be the valid strategy in the future.

2 The Discussion of Package Tours Plan and Service Selection

Therefore, the providers of package tours shall clarify whether the current services can satisfy existing and potential customers. In order to help the providers be aware of the competitiveness of their packages, the research assists them to review package tours with a service evaluation system screening the value satisfaction and price satisfaction index (**Table 1**). A study of price and non-price decision making in the UK package tour industry based on the comparisons of small-scale travel agents and tour operators. They considered that researcher need to re-emphasize the behavior nature of oligopoly, because the characteristic of tourism industry is widely different from manufacturing ones [1]. Tourists' perceptions of online and offline information sources will influence the attitude of tour choices. Most females are risk averter , so they prefer for staying in four- or five-star hotels [2]. Tourists often compare the contents of package tours based on brochures in European countries. These tourists come from United Kingdom, Denmark, and Israel, where are all situated on the fringes of Europe. The distance of actual and perceived between originating country and destination would be associated with the numbers and variety of package tours from destination country [3]. Besides, some studies also pointed out that most consumers emphasized facility, service quality, price, reputation, outlook, and safety after deciding the location for staying [4].

Table 1. The descriptions of aspects/criteria for package tours

Aspects / Criteria	Descriptions
Transportation planning (T)	
Transportation shuttle (T1)	More convenient transportation shuttle service can reduce the uncertainty of tourist's transportation arrangement.
Selection of transportation tools(T2)	More choices of transportation tools can help tourists to plan their itinerary flexibly.
System of Tour route (T3)	Thorough itinerary routes can help consumers to realize the content of touring products.
Number of scenic spots along the way(T4)	More famous scenic spots along the way can attract the customer's view and the willingness to participate this tour..
Planning of itinerary along the way (T5)	The scenery along the route will influence the consumer's desire to travel.
Accommodation planning (H)	
Location environment (H1)	The environment of accommodation will influence the customer's decision.
Pricing of accommodation (H2)	Different pricing of the accommodation can fit different kind of customers.
Accommodation service (H3)	The level of service will influence the customer's willingness to accomadate.
Affiliated facilities (H4)	Better affiliated facilities of the accommodation make customers feeling better.
Arrangement of recreation activity (H5)	More recreation activities means more leisure time.
Schedule arrangement (S)	
Theme tour (S1)	Theme can fit different kind of customer for different travel demand.
Shopping & delicacies experience(S2)	Various selections of shopping and delicacies can fit different kind of shoppers and customers of different tastes.
Ecological & scenic experience (S3)	Diverse ecological & scenic experience can satisfy the needs for ecological & scenic hobbyists.
Local culture experience (S4)	Local culture travel in depth can satisfy the needs for native culture hobbyists.
T Cultural event (S5)	Cultural can satisfy the needs of tourism & festival hobbyists.
Food service planning (F)	
Local delicacies (F1)	The choice of delicacies can further satisfy consumer demands for catering.
Hygiene condition of food and drinks (F2)	Good hygiene condition assure the traveler that there was no danger in food.
Dining environment (F3)	Good dining environment make dinners being delight when dining,
Reservation service of catering (F4)	The reservation service of restaurant can contribute diners to save their time for waiting.
Booking service for souvenirs (F5)	The booking service of souvenirs can enhance the purchase willingness of consumers for tour products.
Package marketing (M)	
Promotion of local activity (M1)	Integrated local activities would be a useful reference for tour planning or purchasing.
Journey information (M2)	Journey information would be a useful reference for tour planning or purchasing.
Assessment for itinerary (M3)	The grade of sssessment for itinerary would be a useful reference for tour planning or purchasing.
Introduction by special magazine column (M4)	The special introduction in magazine column would be a useful reference for tour planning or purchasing.
Suggestion by tourism program (M5)	The recommendation of tourism program would be a useful reference for tour planning or purchasing.
Expenditure (E)	
Sightseeing expenses for itinerary (E1)	Various combination and selection of sightseeing expenditure could contribute consumers to spend money within their budget.
Accommodation expenses for itinerary (E2)	Various combination and selection of accommodation expenditure could contribute consumers to spend money within their budget.
Catering expenses for itinerary (E3)	Various combination and selection of catering expenditure could contribute consumers to spend money within their budget.
Transportation expenses for itinerary (E4)	Various combination and selection of transportation expenditure could contribute consumers to spend money within their budget.
Promotional and discount package (E5)	Package discount could contribute consumers to spend money within their budget.

3 The Service Value Position for Package Tour Plan

In the real world, problems are mostly related. Therefore, criteria relation structure for solving problems shall be established when dealing with aspect and criteria relation problems. Fig.1 shows the relation between aspects and criteria. e_{ij} means the influence of aspect/criterion *i* on aspect/criterion *j* while e_{ii} refers to the influence of aspect/criterion *i* on itself. The traditional method of independent decision making is not applicable when trying to solve problems related with each other or themselves. When we finished the aspect/criterion interrelated figure, we can understand the structural relation among aspects / criteria and decide how to analyze the importance of aspect/ criterion relation or the influence among aspect/criterion. There are 3 kinds of possibilities for general criteria structure, AHP is employed to calculate the weights

of aspects/criteria under independent condition, the dependence and feedback ANP is applied to analyzing the dependence and feedback, and Fuzzy Integral is used to clarify inter-dependent/ relation.

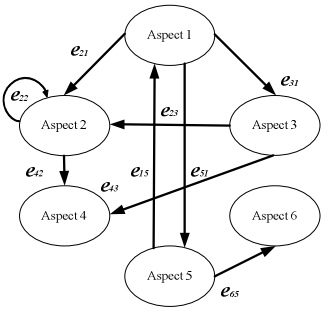


Fig. 1. The decision map of relation map

3.1 Decision-Making Trial and Evaluation Laboratory (DEMATEL)

DEMATEL is used to construct the structure of network relation map (NRM) of the package tours service. When users are making decisions in using packages tour services, there are many criteria to consider. The most common problem they face is that those criteria have impacts on one another. Therefore, before making improvements on criteria, it is necessary to clarify the key criteria and make effective improvements on them to enhance overall satisfaction. When a decision-maker needs to improve a lot of criteria, the best way to handle this is to determine and to improve the key criteria which impact others most. DEMATEL has been widely adopted for solving complicated problems. In recent years, DEMATEL has drawn lots of attention on decision and management domains. Some recent studies considered that the DEMATEL techniques are applied for solving complex studies, such as developing global managers' competencies [5], evaluating performance in e-learning programs [6], causal analytic method for group decision making [7], value-created system of science (technology) park [8], and the evaluation model of vehicle telematics system[9]. This study divides DEMATEL technique into five steps: (1) calculate the original average matrix; (2) calculate the direct influence matrix; (3) calculate the indirect influence matrix; (4) calculate the full direct/indirect influence matrix; and (5) analyze the structure of NRM.

Step1: Calculate the original average matrix

Respondents were asked to score the influence that each aspect exerts on each of the others, according to scoring scales ranging from 0 to 4. "0" means no influence and "4" means "extremely strong influence, between aspect/criterion; "1", "2", and "3" mean "low influence", "medium influence" and "high influence" respectively.

Step 2: Calculate direct influence matrix

We processed the "original influence matrix $\boldsymbol{A}$" by using Eqs. (1) and (2) and got the "direct influence matrix $\boldsymbol{X}$".

$$\boldsymbol{X} = s\boldsymbol{A}, \quad s > 0 \tag{1}$$

where

$$s = \min_{i,j} [1/\max_{1\le i\le n} \sum_{j=1}^{n} a_{ij}, 1/\max_{1\le j\le n} \sum_{i=1}^{n} a_{ij}], \quad i, j = 1,2,...,n \tag{2}$$

and $\lim_{m\to\infty} \boldsymbol{X}^m = [0]_{n\times n}$, where $\boldsymbol{X} = [x_{ij}]_{n\times n}$, when $0 < \sum_{j=1}^{n} x_{ij} \le 1$ or $0 < \sum_{i=1}^{n} x_{ij} \le 1$, and at least one $\sum_{j=1}^{n} x_{ij}$ or $\sum_{i=1}^{n} x_{ij}$ equal one, but not all. So we can guarantee $\lim_{m\to\infty} \boldsymbol{X}^m = [0]_{n\times n}$.

Step 3: Calculate Indirect Influence Matrix

The indirect influence matrix can be derived from Eq. (3).

$$\boldsymbol{IT} = \sum_{i=2}^{\infty} \boldsymbol{X}^i = \boldsymbol{X}^2(\boldsymbol{I} - \boldsymbol{X})^{-1} \tag{3}$$

Step 4: Calculate full influence matrix

Full influence matrix $\boldsymbol{T}$ can be derived from Eqs. (4) or (5). The full influence matrix $\boldsymbol{T}$, consists of multiple elements, indicated as Eq. (6). The sum vector of the row value is $\{ d_i \}$, and the sum vector of the column value $\{ r_j \}$; then, let $i = j$, the sum vector of row value plus column value is $\{ d_i + r_i \}$, which means the full influence of the matrix $\boldsymbol{T}$. As the sum of the row value plus the column value $\{ d_i + r_i \}$ is higher, the relationship of the dimension or criterion is stronger. The sum of the row value minus the column value is $\{ d_i - r_i \}$, which means the net influence relationship. If $d_i - r_i > 0$, it means the degree of influencing others is stronger than the degree to be influenced; otherwise, $d_i - r_i < 0$.

$$\boldsymbol{T} = \boldsymbol{X} + \boldsymbol{IT} = \sum_{i=1}^{\infty} \boldsymbol{D}^i \tag{4}$$

$$\boldsymbol{T} = \sum_{i=1}^{\infty} \boldsymbol{D}^i = \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{D})^{-1} \tag{5}$$

$$\boldsymbol{T} = [t_{ij}], \quad i, j \in \{1,2,...,n\} \tag{6}$$

$$\boldsymbol{d} = \boldsymbol{d}_{n\times 1} = [\sum_{j=1}^{n} t_{ij}]_{n\times 1} = (d_1,...,d_i,...,d_n) \tag{7}$$

$$\boldsymbol{r} = \boldsymbol{r}_{n\times 1} = [\sum_{i=1}^{n} t_{ij}]'_{1\times n} = (r_1,...,r_j,...,r_n) \tag{8}$$

Step 5: Analyze the network relationship map

According to the aspects/criteria defined in Table 1, some experts were asked to discuss the relation and influence levels of criteria under the same aspects/ criteria and to score the relation and influence among criteria based on the DEMATEL technique. Aspects/criteria are divided into different types, so the experts could answer the

questionnaire in areas/fields where they were familiar. The net full influence matrix, T_{net}, is determined by the Eq. (9) .

$$\boldsymbol{T_{net}} = [t_{ij} - t_{ji}], \qquad i, j \in \{1, 2, ..., n\} \tag{9}$$

The diagonal items of the matrix are all 0. In other words, the matrix contains a strictly upper triangular matrix and a strictly lower triangular matrix. Moreover, while values of strictly upper triangular matrix and strictly lower triangular matrix are the same, their symbols are opposite. This property helps us only to choose one of strictly triangular matrix.

3.2 Principal Component Analysis (PCA)

This study uses PCA to analyze the original data of importance degree. It can be used to simplify the large number of criteria and it also can satisfy the hypothesis of AHP/ANP on the independence/dependence of criteria included in system aspect. However, the founder of ANP, Professor Saaty, didn't explicitly define it [10]. From the paper analysis of AHP/ANP, it can be figured out that the hypothesis is that criteria in aspects are independent/dependent. That's why we use this technique in this study. The research conducts the Principal components analysis (PCA) for the original data of importance degree (the maximum value is 10 and minimum value is 0) collected by the questionnaire survey.

3.3 Analytic Network Procedure (ANP)

The ANP method is expressed by a unidirectional hierarchical relation among decision levels [10-12]. The top element of the hierarchy is the overall goal for the decision model. The hierarchy decomposes to a more specific criterion, until a level of manageable decision criteria is met [13]. Under each criterion, sub-criteria elements relative to the criterion can be constructed. The ANP separates complex decision problems into elements within a simplified hierarchical system [14-16]. The steps of the ANP method are described as follows: (1) clarify the questions and constructs the framework, (2) design the questionnaires and the survey, (3) determine the relative importance of factors by pair-wise comparison to calculate factor weights by dependence and feedback, and checks the consistency of logical judgment, (4) calculate the supermatrix, and (5) determine the factor weights[14, 16].

3.4 Vlse kriterijumska Optimizacija I Kompromisno Resenje (VIKOR)

The VIKOR method is used to evaluate and rank the performance of benchmarked alternatives. The VIKOR method is one of MCDM methods, and applies to solve a discrete decision problem with non-commensurable and conflicting criteria. This method focuses on ranking and selecting the solution from a set of alternatives, and determines compromised solutions for a problem with conflicting criteria. This helps the decision makers to get a final decision. The compromised solution is a feasible one closest to the optimal scenario, and a compromise means an agreement established through mutual concessions. The basic concept of VIKOR is to identify

the positive-ideal solution and the negative-ideal solution. The positive solution is the best solution that satisfies the most required criterion whereas the opposite is the negative-ideal solution. The VIKOR method can rank and determine the difference of negative and positive ideal solutions between services of the existing service systems of tourist package tours. When calculating the distance between the ideal solution and the proposed service systems of package tours, the score of every criterion should be summarized. The gap between the consumers' most satisfied one and most dissatisfied one is also analyzed regarding services of the existing service systems of package tours. The VIKOR method starts with the form of the L_P –metric , which is used as an aggregating function in a compromised alternative method and it develops the multi-criteria measure for compromised ranking [17, 18]. The VIKOR provides a maximum group utility of the "majority" and a minimum individual regret of the "opponent". The compromised solutions could be the base for negotiation, involving the decision makers' preferences by criteria weights (Fig.2).

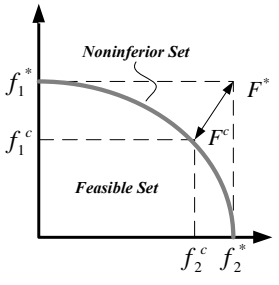


Fig. 2. Ideal and compromised solutions

Where: F^* is the ideal solution. f_1^* represents the ideal value (or called the aspired/desired level) of criterion 1. f_2^* shows the ideal value (the aspired/desired level) of criterion 2. When both criteria are conflicting, the first criterion can be reached for the ideal value, but the second criterion has to be neglected for its performance, and vice versa. For simultaneously fulfilling these conflicting criteria, compromise is necessary. The grey arc in Fig. 2 means the combination of non-inferior solutions. The compromise solution, F^c, is a feasible solution that is "closest" to the ideal F^*. A compromise means an agreement established by mutual concessions. The VIKOR method is presented with the following steps:

Step 1: Determine the best f_k^* value and the worst f_k^- value in criterion i

$$f_i^+ = \left\{ \left(\max_k f_{ik} \mid k \in I_1 \right), \left(\min_k f_{ik} \mid k \in I_2 \right); \text{or setting the aspired level for } i \text{ criterion} \right\}, \ \forall k \tag{10}$$

$$f_i^- = \left\{ \left(\min_k f_{ik} \mid k \in I_1 \right), \left(\max_k f_{ik} \mid k \in I_2 \right); \text{or setting the worst level for } i \text{ criterion} \right\}, \ \forall k \tag{11}$$

where k is the kth alternative; i is the criterion; f_{ik} is the performance value of the ith criterion of the kth alternative; I_1 is the cluster of utility-oriented criteria; I_2 is the

cluster of cost-oriented criteria; f_i^* is the positive-ideal solution (or setting the aspired level); and f_i^- is the positive-ideal solution (or setting the worst level).

Step 2: Calculate the values S_k and Q_k, $k = 1,2,...,m$, using the relations

Let r_{ik} be $r_{ik} = (|f_i^* - f_{ik}|)/(|f_i^* - f_i^-|)$. Before we formally introduce the basic concept of the solutions, let us define a class of distance functions by Yu (1973).

$$d_k^p = \left\{ \sum_{i=1}^{n} [w_i (|f_i^* - f_{ik}|)/(|f_i^* - f_i^-|)]^p \right\}^{1/p} = \left\{ \sum_{i=1}^{n} [w_i r_{ik}]^p \right\}^{1/p}, p \geq 1 \tag{12}$$

$$S_k = d_k^{p=1} = \sum_{i=1}^{n} w_i r_{ik}, \tag{13}$$

$$Q_k = d_k^{p=\infty} = \max_k \{ r_{ik} \mid i = 1,2,...,n \}, \tag{14}$$

where S_k shows the average gap for achieving the aspired/desired level; Q_k shows the maximal degree of regret for prior improvement of gap criterion. w_i is the weight of the criterion i and $i = 1,2,...,n$, expressing the relative importance value of the criteria gained via the application of the ANP method, based on NRM.

Step 3: Calculate the index values R_k, $k = 1,2,\cdots,m$, using the relation

$$R_k = v(S_k - S^*)/(S^- - S^*) + (1-v)(Q_k - Q^*)/(Q^- - Q^*) \tag{15}$$

$$S^* = \min_k S_k, \quad S^- = \max_k S_k$$

$$Q^* = \min_k Q_k, \quad Q^- = \max_k Q_k$$

where $S^* = \min_k S_k$ (showing the minimal average gap is the best, we also can set $S^* = 0$), $S^- = \max_k S_k$ (we can set $S^- = 1$), $Q^* = \min_k Q_k$ (showing the minimal degree of regret is the best, we also can set $Q^* = 0$), $Q^- = \max_k Q_k$ (we can set $Q^- = 1$). We also can re-write Eq. (15), $R_k = vS_k + (1-v)Q_k$.

Step 4: Rank the alternatives

In addition, $0 \leq v \leq 1$ when $v > 0.5$, it indicates that S is emphasized more than Q in Eq. (15), whereas when $v < 0.5$ it indicates that Q is emphasized more than S in Eq. (15). More specifically, when $v = 1$, it represents a decision-making process that could use the strategy of maximum group utility; whereas when $v = 0$, it represents a decision-making process that can use the strategy of minimum individual regret, which is obtained among maximum individual regrets/gaps of lower level dimensions of each alternative (or aspects/objectives). The v affects the ranking order of the dimensions /aspects/criteria and is usually determined by the experts or decision making. In this paper, R_k (here, v =0.5) is applied to determine the customer satisfaction index (VSI). R_k can also consider the index of the maximum group utility

and the minimum individual regret of the "opponent", where R_k smaller is better and $0 \leqq R_k \leqq 1$. The study ranks the alternatives by establishing the value satisfaction index (VSI) and price satisfaction index (PSI) which take R_{vk} and R_{pk} value when v value equals to 0.5 and constitutes the satisfaction index table for a maximum group utility and a minimum individual regret/gap. When R_{vk} and R_{pk} is the smaller-the-better index and ranges from 0 to 1, the study transfers it to a bigger-the-better index by calculating '$1-R_{vk}$' and '$1-R_{pk}$'. Therefore, when both v values of value satisfaction and price satisfaction equal to 0.5, V and P are R_{vk} and R_{pk}. Here, both VSI and PSI come from $1-R_{vk}$ and $1-R_{pk}$, and the VSI and PSI for different alternatives can be calculated.

4 The Empirical Study of Package Tour in Taiwan's Nantou Area

This research plans six package tours (Sea of Clouds Tour (Tour 1), Traditional culture experiencing tour (Tour 2) , Lake and sky tour (Tour 3), Enjoying spring tour (Tour 4), Railroad exploration tour (Tour 5), and Museum visiting tour (Tour 6)) for the guideline of travel mode choices and divides the regional package tours into three transportation tools (bus tour, railroad tour, and private vehicle tour) and four theme modes (shopping and cuisine, nature approaching, culture exploring, and event participation) in Taiwan's Nantou area (Fig.3).

Fig. 3. The relative location of Nantou country

R_{vk} and R_{pk} (here v =0.5) is applied to determine the (customer) value satisfaction index (VSI) and (customer) price satisfaction index (PSI). R_{vk} and R_{pk} could also consider the index of the maximum group utility and the minimum individual regret of the "opponent", where R_{vk} and R_{pk} means smaller is better and $0 \leqq R_{vk}, R_{pk} \leqq 1$. However, this research prefers $1-R_{vk}$ and $1-R_{pk}$ for evaluation, which means $1-R_{vk}$ and $1-R_{pk}$ bigger are better. When the v value of VSI is 0.5, then $V = R_{vk}$ and VSI=

$1-R_{vk}$ and the v value of PSI is 0.5, then $P=R_{pk}$ and PSI=$1-R_{pk}$. Therefore, VSI and PSI of different alternatives could be gained. Under v=0.5, R_{vk} =0.295, VSI=0.705 of Tour 2 (Traditional culture experiencing tour) and R_{vk} =0.376, VSI=0.624 of Tour 6 (Museum visiting tour). The VSI rank of package tours are $T_2 \succ T_5 \succ T_4 \succ T_1 \succ T_3 \succ T_6$. As shown in Table 2, under v=0.5, R_{pk} =0.281, PSI=0.719 of Tour 2 (Traditional culture experiencing tour) and R_{pk} =0.347, PSI=0.653 of Tour 6 (Museum visiting tour). The PSI rank of package tours are $T_2 \succ T_5 \succ T_4 \succ T_3 \succ T_1 \succ T_6$.

Table 2. The VSI and PSI ranks for package tours

v=0.5	Tour 1 Sea of clouds Tour	Tour 2 Traditional culture experiencing tour	Tour 3 Lake and sky tour	Tour 4 Spring enjoying tour	Tour 5 Railroad exploration tour	Tour 6 Museum visiting tour
V = Q_{vk}	0.343	0.295	0.363	0.333	0.301	0.376
VSI = $1-Q_{vk}$	**0.657**	**0.705**	**0.637**	**0.667**	**0.699**	**0.624**
P = Q_{pk}	0.332	0.281	0.331	0.327	0.305	0.347
PSI = $1-Q_{pk}$	**0.668**	**0.719**	**0.669**	**0.673**	**0.695**	**0.653**

From Fig. 4 and Table 2, the Traditional Culture Experiencing Tour (Tour 2) has the highest aggregated satisfaction index (0.705, 0.719) and the Museum Visiting Tour (Tour 6) demonstrates the lowest aggregated satisfaction index (0.624, 0.653), followed by the Railroad Exploration Tour (Tour 5) (0.699, 0.695), the Spring Enjoying Tour (Tour 4) (0.667, 0.673), the Sea of Clouds Tour (Tour 1) (0.657, 0.668), and the Lake and Sky Tour (Tour 3) (0.637, 0.669). In addition, the 6 package tours present 3 competition strategies proposed by operators as follows: both high VSI and PSI (the Traditional Culture Experiencing Tour (Tour 2) (0.705, 0.719) and the Railroad Exploration Tour (Tour 5) (0.699, 0.695)), both medium VSI and PSI (the Spring Enjoying Tour (Tour 4) (0.667, 0.673), the Sea of Clouds Tour (Tour 1) (0.657, 0.668), and the Lake and Sky Tour (Tour3) (0.637, 0.669)), and both low VSI and PSI (the Museum Visiting Tour (Tour 6) (0.624, 0.653)).

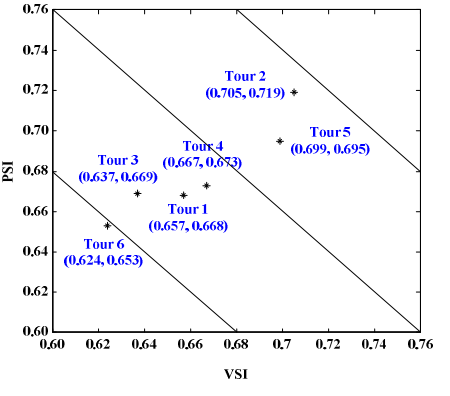


Fig. 4. The satisfaction location of package tours based on VSI and PSI

5 Conclusions

Price is the crucial factor for products on tourism market. It seems that high-price strategy cannot attract consumers. Instead of that, it is proved by the Traditional Culture Experiencing Tour (Tour 2) that low-cost tour is a good alternative for customers. The Tour emphasizes traditional crafts and local culture, and participants are able to experience cultural stuffs. Free combination of diversified packages of different activities helps cost down the service price with the increasing demands. In addition, the Railroad Exploration Tour focuses on the villages along the railroad, which allures railroads fans and tourists interested in regional development history, local cultural. The rail station represents the origin of regional development and plays the role of bridging local economic activities with external development. Furthermore, hot spring and natural experience tours are characterized with scarcity of the natural resources. Boosting demands may push up the travel price and therefore, the segmentation of customers based on time is applicable. Tourism coupons can help separate the tourism market when retired tourists with flexible time may be attracted by the bonus price to the tours in off-peak time. During holidays, the long-term cooperation or the aggregated efforts of operators can be applied to lower prices by large quantity. Diversification of tour contents can also relieve the problem of rising price caused by the resource scarcity. The museum tour needs to be enhanced by working with schools and institutions and by developing new customers through diversified pricing and marketing strategies.

References

1. Davies, B., Downward, P.: Exploring price and non-price decision making in the UK package tour industry: Insights from small-scale travel agents and tour operators. Tourism Management 28(5), 1236–1261 (2007)
2. Alvarez, M., Asugman, G.D.: Explorers versus planners: A Study of Turkish Tourists. Annals of Tourism Research 33(2), 319–338 (2006)
3. Enoch, Y.: Contents of Tour Packages: A cross-cultural comparison. Annals of Tourism Research 23(3), 599–616 (1996)
4. Hsieh, L.F., Lin, L.H., Lin, Y.Y.: A service quality measurement architecture for hot spring hotels in Taiwan. Tourism Management 29(3), 429–438 (2008)
5. Wu, W.W., Lee, Y.T.: Developing global managers' competencies using the fuzzy DEMATEL method. Expert Systems with Applications 32(2), 499–507 (2007)
6. Tzeng, G.H., Chiang, C.H., Li, C.W.: Evaluating intertwined effects in e-learning programs: A novel hybrid MCDM model based on factor analysis and DEMATEL. Expert Systems with Applications 32(4), 1028–1044 (2007)
7. Lin, C.J., Wu, W.W.: A causal analytical method for group decision-making under fuzzy environment. Expert Systems with Applications 34(1), 205–213 (2008)
8. Lin, C.L., Tzeng, G.H.: A value-created system of science (technology) park by using DEMATEL. Expert Systems with Applications 36(6), 9683–9697 (2009)
9. Lin, C.L., Hsieh, M.S., Tzeng, G.H.: Evaluating vehicle telematics system by using a novel MCDM techniques with dependence and feedback. Expert Systems with Applications 37(10), 6723–6736 (2010)

10. Niemira, M.P., Saaty, T.L.: An Analytic Network Process model for financial-crisis forecasting. International Journal of Forecasting 20(4), 573–587 (2004)
11. Blair, A.R., Nachtmann, R., Saaty, T.L., Whitaker, R.: Forecasting the resurgence of the US economy in 2001: an expert judgment approach. Socio-Economic Planning Sciences 36(2), 77–91 (2002)
12. Saaty, T.L.: Rank from comparisons and from ratings in the analytic hierarchy/network processes. European Journal of Operational Research 168(2), 557–570 (2006)
13. Meade, L.M., Presley, A.: R&D project selection using the analytic network process. IEEE Transactions on Engineering Management 49(1), 59–66 (2002)
14. Shyur, H.J.: COTS evaluation using modified TOPSIS and ANP. Applied Mathematics and Computation 177(1), 251–259 (2006)
15. Yu, R., Tzeng, G.H.: A soft computing method for multi-criteria decision making with dependence and feedback. Applied Mathematics and Computation 180(1), 63–75 (2006)
16. Shyur, H.J., Shih, H.S.: A hybrid MCDM model for strategic vendor selection. Mathematical and Computer Modelling 44(7-8), 749–761 (2006)
17. Yu, P.L.: A class of solutions for group decision problems. Management Science 19(8), 936–946 (1973)
18. Zeleny, M.: Multiple Criteria Decision Making. McGraw-Hill, New York (1982)

Incremental 3D Reconstruction Using Bayesian Learning

Ze-Huan Yuan[1], Lu Tong[1,2,*], Hao-Yi Zhou[1], Chen Bin[1], and Jia-Ning Li[1]

[1] State Key Laboratory of Software Novel Technology, Nanjing University, China 210093
[2] Jiangyin Institute of Information Technology of Nanjing University, China
zhyuan001@gmail.com, lutong@nju.edu.cn

Abstract. We present a novel algorithm for 3D reconstruction in this paper, converting incremental 3D reconstruction to an optimization problem by combining two feature-enhancing geometric priors and one photometric consistency constraint under the Bayesian learning framework. Our method first reconstructs an initial 3D model by selecting uniformly distributed key images using a view sphere. Then once a new image is added, we search its correlated reconstructed patches and incrementally update the result model by optimizing the geometric and photometric energy terms. The experimental results illustrate our method is effective for incremental 3D reconstruction and can be further applied for large-scale datasets or to real-time reconstruction.

Keywords: Incremental reconstruction, Bayesian model, PMVS.

1 Introduction

In computer vision, 3D reconstruction has been one of the widely researched areas in the recent decades, and automatic geometric reconstruction plays a key role in automated intelligent systems. With the decreasing costs of video equipments, we now have the opportunity and an urgent need to run automated and accurate 3D reconstruction algorithms directly on multiple photographs or video clips. Indeed, the most important technological ingredients towards this goal are already in place. We have known that feature matching algorithms [6] can provide accurate correspondences, structure-from-motion (SFM) algorithms use these correspondences to evaluate accurate camera pose, and multi-view-stereo (MVS) methods finally reconstruct dense and accurate surface models of complex objects from a moderate number of calibrated images. Actually, the existing MVS algorithms has nearly achieved surface coverage of about 95% and depth accuracy of about 0.5 mm from a set of low resolution (640x480) images as reported [1, 18].

MVS plays an important role in automatic acquisition of geometric objects. Existing state-of-the-art MVS algorithms can be roughly categorized into four classes: *voxel*, *mesh*, *depth maps* and *patch* based methods. *Voxel-based* MVS methods (VMVS) [2], [3], [4], [5] represent geometry on a regularly sampled 3D grid (volume), either as a discrete occupancy function or a function encoding distance to the closest surface.

* Corresponding author.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 754–763, 2012.

Algorithms based on *deformable polygonal meshes* [7] [8] represent a surface as a set of connected planar facets and operate by iteratively evolving a surface to decrease or minimize a cost function. Approaches based on *multiple depth maps* [9], [10] model a scene as a set of depth maps and fuse individual depth maps into a single 3D model. Finally, *patch-based* MVS (PMVS) [1] algorithms output a dense collection of small oriented rectangular patches covering the observed surface obtained from pixel-level correspondences. Recently, CMVS [17] is approved effective in reconstructing from images of crowed scenes without any initialization process.

However, the mentioned methods still face the following difficulty. They cannot well handle incremental reconstruction tasks. In another word, the input images should be well sequenced before reconstruction. Once a geometric object is obtained, it cannot be incrementally updated when facing a new input view image.

Thus in this paper, we propose a novel algorithm aiming at incrementally reconstructing a 3D model using the Bayesian framework. We first select a group of key views uniformly distributed on our view sphere to create an initial 3D surface modeled by PMVS as stated above. Then when a new calibrated image is input, we 1) map it into a triangle on our view sphere, 2) search the correlated patches with the new input view, and 3) automatically update the initial 3D model using the photometric consistency constraint and geometric smoothness priors under the Bayesian inference framework. Note that once a new image is added, more geometric details can be extracted and integrated to incrementally optimize the final 3D model.

Our method has two main contributions. First, we propose a novel incremental 3D reconstruction framework, which makes full use of new views to incrementally update an existing 3D model. As a result, the reconstruction process is more efficient and convenient, especially useful for automatic 3D reconstruction from a large number of real-life images or videos and real-time reconstruction. Second, to our knowledge, no previous work has attempted to reconstruct 3D models using the Bayesian learning framework, where pixel-level information and geometric level constraints are well integrated to optimize the final model. As a result, the reconstruction accuracy can be effectively improved.

2 Our Method

In this section, we give our incremental reconstruction algorithm in details. Our method can be briefly summarized as the following three steps:

1. Map the given multi-view images set I_{source} to a view sphere $S_{initial}$ and select uniformly distributed key views to initialize a 3D model;
2. For each new input image i_{new}, map it to $S_{initial}$ and search its related patches set P_{update} on the 3D model;
3. Re-calculate the patches of P_{update} using the Bayesian learning framework to incrementally refine the 3D model.

Step 2 and 3 are repeated until there are no new input images. Note that in Step 2, only a subset P_{update} (named *seed patches set*) on the previous 3D model is chosen to be updated for any new input image rather than all the patches on the model. It is based on the following fact that in each incremental recursion step, the existing

patches on the previous 3D model may have different *correlations* to i_{new} and we need not update those patches having low *correlations*. For example, there is no (or too low) *correlation* between i_{new} and another patch that is completely invisible to it. This helps reduce the computational cost, simultaneously without losing accuracy in our incremental reconstruction.

2.1 Initialize a 3D Model

Given a calibrated image set I_{source}, we need firstly to select an image subset uniformly distributed in different viewpoints to reconstruct an initial 3D model. The initial key views are selected as follows: 1) map each view image in I_{source} to a view sphere $S_{initial}$ (see Fig. 1(a)), with its coordinate determined by the corresponding image plane, namely the normalized principal axis vector obtained from its projection matrix, and 2) sample the key views uniformly across the sphere.

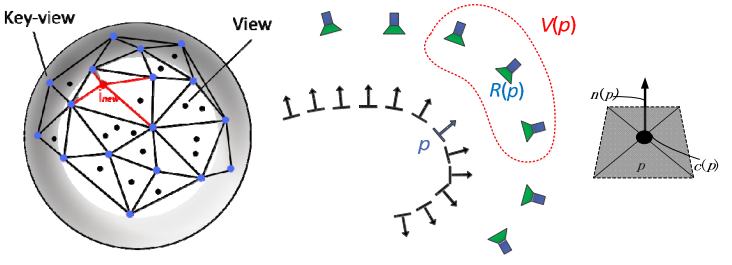


Fig. 1. (a) The view sphere. (b) The patch model.

Next, we triangulate $S_{initial}$ by grouping the neighboring key views on it into triangles using the Delaunay Triangulation algorithm [11]. 3D initial geometric model $\overline{S}$ can be simultaneously reconstructed using [12] from key views. Note that the geometric contour is reconstructed using the patch-based approach [1], where a 3D surface is covered by a plenty of patches, and a patch p is essentially a local tangent plane approximation of the surface. A patch p here has three geometric attributes (see Fig. 1(b)): $c(p)$, $n(p)$ and $R(p)$, where $c(p)$ denotes the geometric center, $n(p)$ is the unit normal vector oriented toward the camera observing it, while a reference image $R(p)$ is an image chosen from $V(p)$ where p is *truly* visible on the condition that the retinal plane of $R(p)$ is nearly parallel to p within a tiny distortion.

As a result, a triangulated view sphere and a 3D patch model are obtained as the initializations of our incremental updating system.

2.2 Search Related Patches for a New Input Image

In our incremental reconstruction step, we first search a corresponding patch subset from the previous 3D model for any new input calibrated image, and then extend the subset to make the model more uniform and well-sampled.

2.2.1 Search Seed Patches for Any Input Image

To search the seed patches P_{update} for any incrementally input image i_{new}, we first search a proper triangle T on $S_{initial}$, where i_{new} can be mapped into using SIFT [6] as follows:

$$T \leftarrow \arg\max_{T} \sum\nolimits_{v \in T} | x_{i_{new}}^{v} | \tag{1}$$

where $x_{i_{new}}^{v}$ is a set of matches between i_{new} and the key view v corresponding to a vertex in triangle T. Then we search the correlated patch subset P_{update} from the reconstructed 3D model by

$$P_{update} = \bigcup_{v \in T} \{ p \mid p \in \overline{S}, v \text{ is } R(p) \} \tag{2}$$

Obviously, i_{new} provides more useful reconstruction details for the patches in P_{update} than those outside it. Then we update $S_{initial}$ as follows: 1) add a new vertex representing the new image; 2) add a pyramid of triangles by connecting the new image to the three vertices of T, and 3) delete T with i_{new} located in. As a result, we can simultaneously obtain an updated view sphere (see i_{new} in Fig. 1(a)).

2.2.2 Extend the Seed Patches

Next, we extend the patch model to obtain a relatively uniform patch density along different viewpoints over the surface. The extension is associated with the orientation of the new view and the average density of the existing global surface. Note that during this process, we may create new patches under the local geometric constraints to improve patch density where patches are too sparse. Our extension has the following steps:

- Estimate local density D_p for every patch p in 3D model. We count its neighbors $N(p)$ to evaluate the local density equivalently as follows:

$$N(p) = \{ p' \mid p' \in \overline{S}, | (c(p) - c(p')) \cdot n(p) | + | (c(p) - c(p')) \cdot n(p') | < \rho \} \tag{3}$$

$$D_p = | N(p) | \tag{4}$$

  where ρ can be computed relating to the distance at the depth of the center of $c(p)$ and $c(p')$ corresponding to an image displacement of u pixels in $R(p)$(u=2 in our experiment);
- Compute the global average density D_g by averaging all estimated local densities;
- For every seed patch in P_{update} with its local density less than 0.5* D_g, use the SMOTE [13] to oversample new ones whose initialization can be seen in Table 1 between the seed patch and its neighbors (see Fig. 2). As a result, the original geometric constraints can be well maintained;
- Add the new patches into P_{update}.

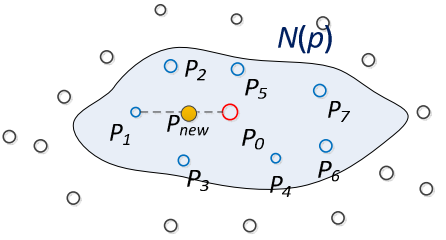


Fig. 2. Seed patches extension, where P_{new} is generated along the line combining a seed patch P_0 and one of its neighbors P_1

2.3 Incremental Surface Reconstruction Using Bayesian Learning

This section introduces the Bayesian model used in our incremental reconstruction. We aim at discovering the photometric consistency and geometric smoothness constraints to obtain high-quality incremental reconstruction results.

Suppose i_{new} is a *measurement* to our camera from the real scene modeled by PMVS in our method. Let S be the real scene to be modeled, we need reconstruct the most likely surface S_{MAP} given the *measurement* i_{new}. This can be achieved by maximizing the Bayesian posterior probability $p(S|i_{new})$ in the solution space Ω

$$p(S \mid i_{new}) = \frac{1}{Z} p(i_{new} \mid S) p(S), S \in \Omega \tag{5}$$

$$S_{MAP} = \arg\min \left(-\log p(i_{new} \mid S) - \log p(S) \right) \tag{6}$$

in order to reduce the parameter dimensions, we constraint Ω to the expanded patches subset P_{update} as mentioned in Section 2. Note that the constant related to Z is ignored in (6). $p(i_{new} \mid S)$ specifies the likelihood of the *measurement* i_{new} agreeing with S. In other words, it measures how well the normal and coordinate of a patch match the real surface according to the information hidden in i_{new} and the other correlated images. It can be defined by the use of photometric discrepancy function [1], which we choose to express the photometric consistency:

$$p(i_{new} \mid S) \propto \exp(-\eta E_p) \tag{7}$$

$$E_p = \frac{1}{|S|} \sum_{p \in S} \frac{1}{|V(p)| - 1} \sum_{i \in V(p)/i_{new}} h(p, i_{new}, i) \tag{8}$$

where η is a control coefficient, and $h(p, i_{new}, i)$ is equal to one minus the pair-wise normalized cross correlation concerning to the patch projection into images i_{new} and i.

We use two constraints to define the prior $p(S)$:

$$p(S) \propto \exp(-\{\lambda E_1 + \zeta E_2\}) \tag{9}$$

where E_1 and E_2 are two geometric smoothness energy terms, and λ, ζ are weighted coefficients. E_1 is used to assure the smoothness of the reconstructed surface. For a natural 3D object, we can model its surface smoothness by accumulating sub-linear potentials of surface curvature similar to [14]. Concretely, we define E_1 as follows:

$$E_1 = \frac{1}{|S|}\sum_{p\in S}\frac{1}{|N(p)|}\sum_{v\in N(p)} f(p,v) \tag{10}$$

$$f(p,v) = \sqrt{(n(p)-n(v))^T (n(p)-n(v))} \tag{11}$$

where $N(p)$ is the neighboring patches set of p defined in (3), $f(p,v)$ is the square-root potential with $f(p,v)=0$ if $n(p)=n(v)$ and positive otherwise.

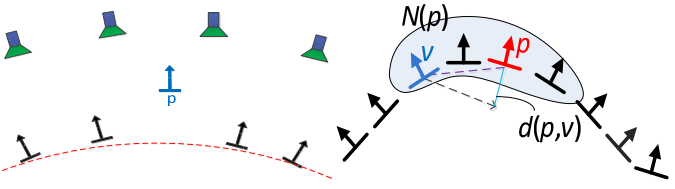


Fig. 3. Geometric smoothness terms. (a) The blue patch p is an outlier; however it has a continuous normal with its neighboring patches. (b) $d(p,v)$ is the absolute distance between two patches p and v along $n(p)$.

However, there still may exist exceptions even (10) is met. For example, in Fig. 3(a), the patch p is an outlier while having well sub-linear continuous relations with normals of its neighbors in $N(p)$. Considering although such a patch has a continuous normal, its geometric location is far away from the real surface, we use another geometric smoothness energy term E_2 to minimize such errors as follows:

$$E_2 = \frac{1}{|S|}\sum_{p\in S}\frac{1}{|N(p)|}\sum_{v\in N(p)} d(p,v) \tag{12}$$

$$d(p,v) = |\, n(p)\cdot(c(v)-c(p))\,| \tag{13}$$

where $d(p,v)$ is the distance between two patches p and v along $n(p)$(see Fig. 3(b)).

This minimization problem requires us to adjust $c(p)$ and $n(p)$ for any patch in S from the initial value to the final convergent solution. It is actually a sparse energy minimization optimization problem. To simplify the complexity and reduce the dimension of variables, we constrain $c(p)$ lie on a ray to assure the projection into $R(p)$ is not changed. Simultaneously, we model $n(p)$ with Euler angles. Thus for every patch, only three parameters participate in the optimization problem, greatly reducing the dimension of the solution space and improve stability in the search process. We use the conjugate gradient descent to solve the global optimization. In this process,

the derivatives for geometric smoothness prior can be directly computed and those for the photometric consistency term are currently estimated numerically.

As a summary, our incremental updating algorithm is shown in Table 1.

Table 1. The incremental algorithm

Input : $S_{initial}$ and 3D patch model $\overline{S}$ reconstructed by PMVS

Output : an improved well-sample, high-resolution and more accurate patch model

While Input an image i_{new}

Locate i_{new} in $S_{initial}$ and find a corresponding triangle T using SIFT

For any p in the 3D patch model

$N_p \leftarrow \{p' \mid p' \in \overline{S}, \mid (c(p)-c(p'))\cdot n(p) \mid + \mid (c(p)-c(p'))\cdot n(p') \mid < \rho\}$

$D_p \leftarrow \mid N(p) \mid$

$P_{update} \leftarrow \bigcup_{v \in T} \{p \mid v \; is \; R(p)\}$

Update $S_{initial}$

Compute D_g by averaging all local density

For any p in P_{update}

If D_p<0.5* D_g

Generate a new patch k

$c(k)$,$n(k) \leftarrow$ oversampling method *smote*(N_p, *sample-rate*,p).

$R(k) \leftarrow R(p)$

$V(k) \leftarrow V(p)$

Add k into P_{update}

For any patch p in P_{update}

$c(p), n(p) \leftarrow \arg\min(\lambda E_1 + \zeta E_2 + \eta E_p), p \in P_{update}$

end while

3 Experiments and Discussions

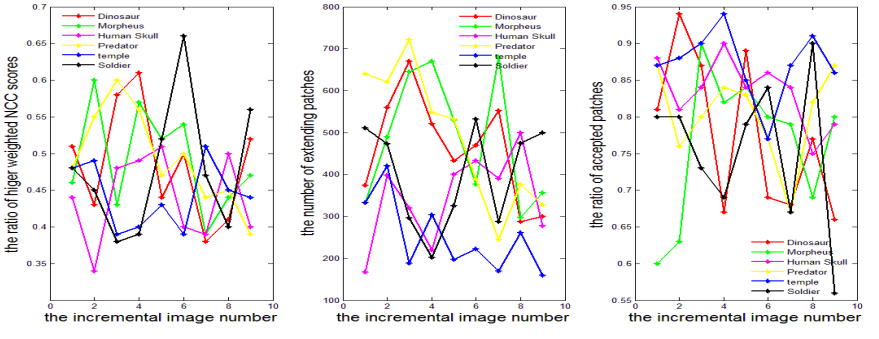
We have implemented our incremental reconstruction algorithm on C++ platform. The datasets [15][16] used in our experiments are shown in Table 2, with the number of the input images, their approximate sizes, the number of the key views we choose and the patch number of the reconstructed initial model using PMVS [12]. In our incremental process, we set λ, ζ andη 0.3, 0.2 and 0.7, respectively.

Fig. 5 gives the incremental reconstruction results of different models, where Column (a) and Column (b) correspond to example 2D images and their initial result models reconstructed from key views, respectively. After gradually adding new images, the result models are incrementally updated, as shown in the rest three columns (c)-(e). It can be seen that the result models can be dynamically optimized and enriched with more details during these processes.

Table 2. The datasets used in our experiments

Name	Images	Image size	Key views	Initial patches
Toy Dinosaur	24	2000*1500	15	27267
Morpheus	24	1400*1200	15	18433
predator	24	1800*1800	15	29620
Human Skull	24	2000*1800	15	45223
temple	312	640 * 480	209	32317

To evaluate our method quantitatively, we adopt the weighted sum of normalized cross correlation (NCC) [1] to model the accuracy of a patch. During each incremental step, we calculate the ratios of those patches with larger weighted NCC scores in P_{update} (see Fig. 4). Fig. 4 is a discrete figure where different points on curves have no relations and can be replaced by tables if enough space available. It can be seen that after adding a new image, the NCC accuracy of nearly 50% of its related patches are improved averagely, illustrating the effectiveness of our method.

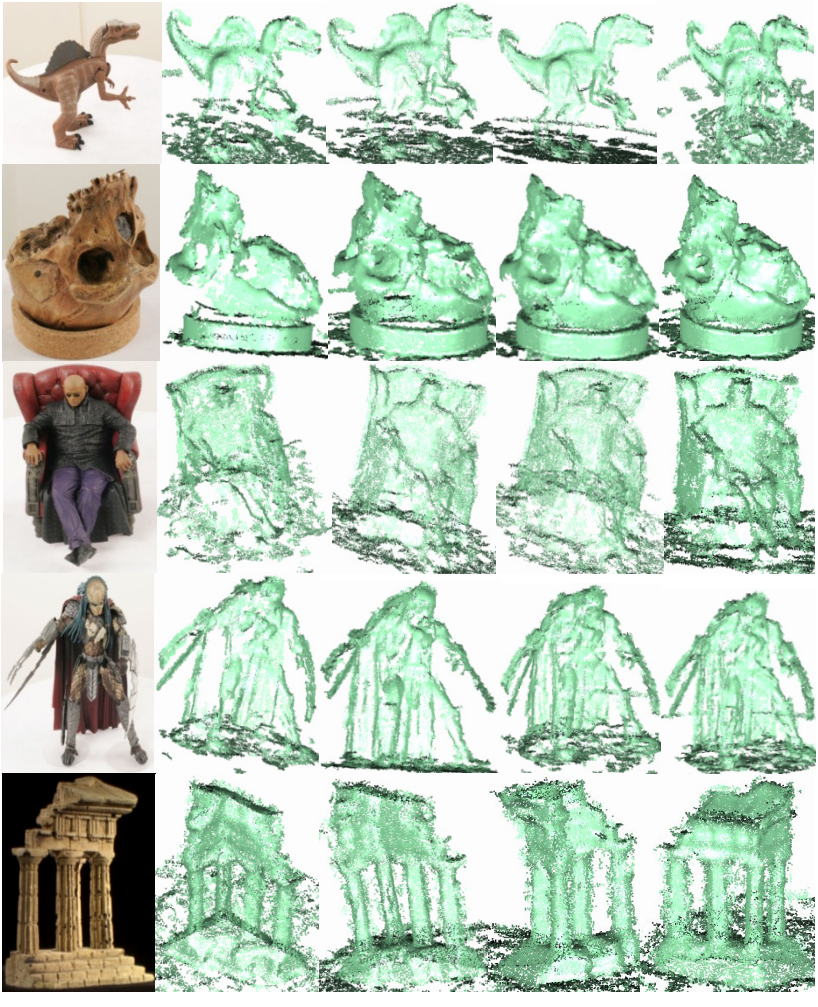


(a) (b) (c)

Fig. 4. The overall statistic analysis. (a) the ratio of patches having higher photometric consistency scores, (b) the number of extended patches, and (c) the ratio of accepted extending patches for different incremental images.

We also find that in Fig. 4(a), the ratio changes with image quality and position on our view sphere during the incremental reconstruction steps. It is due to that for poor-quality images, geometric smoothness term plays an important role in the optimization, and thus the accuracy may be reduced simultaneously because of over-smoothing.

Fig. 4(b) illustrates the number of the extended patches in each incremental reconstruction step with the *sample-rate* as 200% in our experiments. Obviously, the number greatly depends on the viewpoint of 2D images and more patches need to be generated in sparse regions. Note that not all the extended patches are finally added to the result model due to the global geometric constraints and the pixel-level information. Fig. 4(c) gives the accepted patch ratios in our experiments.



(a) 2D images (b) the initial model (c) result 1 (d) result 2 (e) result 3

Fig. 5. Our incremental reconstruction results. (a) 2D sample images, (b) the initial 3D model, (c)-(e) the incremental reconstruction results. From top to bottom, the datasets are *dinosaur*, *human skull cast*, *Morpheus*, *predator* and *temple*.

4 Conclusions

We have developed a novel incremental reconstruction algorithm for calibrated multi-view stereo. Our method first initializes a 3D patch model using the selected key views, and then when inputting a new image interactively, seed patches for which the new image provides useful reconstruction details are searched and then extended to make surface of the 3D target uniform. We end up the incremental learning under Bayesian framework. We focus our future work on directly reconstructing crowed

scene models from real-life videos and online real-time reconstruction. Another improvement may lie on better evaluating 3D model reconstruction methods, especially for incremental reconstruction applications.

Acknowledgements. The work described in this paper was supported by the New Century Excellent Talent of Ministry of Education Project under NCET-11-0232, National Science Foundation of China under Grant No. 61021062, and the 973 Program of China under Grant No. 2010CB327903.

Reference

1. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi view stereopsis. PAMI 32(8), 1362–1376 (2010)
2. Pons, J.-P., Keriven, R., Faugeras, O.D.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. IJCV 72(2), 179–193 (2007)
3. Tran, S., Davis, L.: 3D Surface Reconstruction Using Graph Cuts with Surface Constraints. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 219–231. Springer, Heidelberg (2006)
4. Vogiatzis, G., Torr, P.H., Cipolla, R.: Multi-view stereo via volumetric graph-cuts. In: CVPR, pp. 391–398 (2005)
5. Hornung, A., Kobbelt, L.: Hierarchical volumetric multi-view stereo reconstruction of manifold surfaces based on dual graph embedding. In: CVPR, pp. 503–510 (2006)
6. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
7. Hernández Esteban, C., Schmitt, F.: Silhouette and stereo fusion for 3D object modeling. CVIU 96(3), 367–392 (2004)
8. Furukawa, Y., Ponce, J.: Carved visual hulls for image-based modeling. IJCV 81(1), 53–67 (2009)
9. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. In: CVPR, pp. 2402–2409 (2006)
10. Strecha, C., Fransens, R., Gool, L.V.: Combined depth and outlier estimation in multi-view stereo. In: CVPR, pp. 2394–2401 (2006)
11. `http://www.cse.unsw.edu.au/~lambert/java/3d/delaunay.html`
12. `http://grail.cs.washington.edu/software/pmvs`
13. Chawla, N.V., Bowyer, K.W., Hall, L.O., Philip Kegelmeyer, W.: SMOTE: Synthetic Minority Over-sampling Technique. JAIR 16, 321–357 (2002)
14. Diebel, J.R., Thrun, S.: A Bayesian Method for Probable Surface Reconstruction and Decimation. ACM Transactions on Graphics - TOG 25(1), 39–59 (2006)
15. `http://www.cs.washington.edu/homes/furukawa/research/mview/index.html`
16. `http://vision.middlebury.edu/mview/data/`
17. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards Internet-scale Multi-view Stereo. In: CVPR, pp. 1434–1441 (2010)
18. Furukawa, Y., Ponce, J.: Accurate camera calibration from multi-view stereo and bundle adjustment. IJCV 84(3), 257–268 (2009)

A Stochastic Search on the Line-Based Solution to Discretized Estimation

Anis Yazidi[1], Ole-Christoffer Granmo[1], and B. John Oommen[2,⋆]

[1] Dept. of ICT, University of Agder, Grimstad, Norway
[2] School of Computer Science, Carleton University, Ottawa, Canada

Abstract. Recently, Oommen and Rueda [11] presented a strategy by which the parameters of a binomial/multinomial distribution can be estimated when the underlying distribution is nonstationary. The method has been referred to as the Stochastic Learning Weak Estimator (SLWE), and is based on the principles of *continuous* stochastic Learning Automata (LA). In this paper, we consider a new family of stochastic *discretized* weak estimators pertinent to tracking time-varying binomial distributions. As opposed to the SLWE, our proposed estimator is *discretized*, i.e., the estimate can assume only a finite number of values. It is well known in the field of LA that discretized schemes achieve faster convergence speed than their corresponding continuous counterparts. By virtue of discretization, our estimator realizes extremely fast adjustments of the running estimates by jumps, and it is thus able to robustly, and very quickly, track changes in the parameters of the distribution after a switch has occurred in the environment. The design principle of our strategy is based on a solution, pioneered by Oommen [7], for the Stochastic Search on the Line (SSL) problem. The SSL solution proposed in [7], assumes the existence of an Oracle which informs the LA whether to go "right" or "left". In our application domain, in order to achieve efficient estimation, we have to first *infer* (or rather *simulate*) such an Oracle. In order to overcome this difficulty, we rather intelligently construct an "Artificial Oracle" that suggests whether we are to increase the current estimate or to decrease it. The paper briefly reports conclusive experimental results that demonstrate the ability of the proposed estimator to cope with non-stationary environments with a high adaptation rate, and with an accuracy that depends on its resolution. The results which we present are, to the best of our knowledge, the first reported results that resolve the problem of discretized weak estimation using a SSL-based solution.

Keywords: Weak Estimators, Learning Automata, Non-Stationary Environments, Stochastic Point Location.

⋆ *Chancellor's Professor*; *Fellow: IEEE* and *Fellow: IAPR*. The Author also holds an Adjunct Professorship with the Dept. of ICT, University of Agder, Norway. The third author is grateful for the partial support provided by NSERC, the Natural Sciences and Engineering Research Council of Canada.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 764–773, 2012.

1 Introduction

The problem of estimating the parameters of a distribution, when these parameters change with time, is far from trivial. The reason why the problem is intrinsically difficult is because "traditional" estimation methods rely on the long term properties of the estimation process to conclude on the *quality* of the estimate. Consequently, they attempt to obtain convergence properties that are as "strong" as possible.

Estimators generally fall into various categories including the Maximum Likelihood Estimates (MLE) and the Bayesian family of estimates. Estimates from these families are well-known for having good computational and statistical properties. However, the basic premise for establishing the quality of estimates, including these, is based on the assumption that the parameters being estimated do not change with time, i.e, the distribution is assumed to be stationary. Thus, as mentioned above, it is desirable that the estimate converges to the true underlying parameter with probability 1, as the number of samples increases.

As argued in [11], there are numerous real-life problems in which the environment is non-stationary, and the parameter being estimated changes with time. Once can, for example, be dealing with an ensemble of biased dice, which leads to a multinomially distributed random variable where the parameter (the vector of probabilities for choosing the possible sides) is *switched*, perhaps, periodically, to possibly a new random value. Such a scenario demonstrates the behavior of a non-stationary environment. Thus, in this case, the goal of an estimator scheme would be to estimate the parameter, and to be able to adapt to any changes occurring in the environment. In other words, the algorithm must be able to detect the changes and estimate the new parameter after a *switch* has occurred in the environment. If one uses strong estimators (i.e., estimators that converge w.p. 1), it is impossible for the learned parameter to change rapidly from the value to which it has converged, resulting in poor time-varying estimates.

As opposed to the traditional MLE and Bayesian estimators, Oommen and Rueda, in [11], used the principles of *continuous* stochastic Learning Automata (LA) to propose a Stochastic Learning Weak Estimator (SLWE). Their scheme yielded a *weak* estimator which could learn the parameters of a binomial or multinomial distribution when the underlying distribution is nonstationary. As opposed to their scheme, in this paper, we propose a *discretized* algorithm. However, rather than merely utilize the process of discretization by "uniformly slicing" the parameter space, we propose a novel discretized weak estimator in which the principle of inferring how the updates are to be made, is *itself* based on a meta-learning strategy, explained, in detail, below. Our new estimator can be shown to converge to the true value fairly quickly, and to "*unlearn*" what it has learned so far to adapt to the new, "*switched*" environment. Further, as in the case of the SLWE, the convergence of the estimate is weak, i.e., with regard to the first and second moments.

The principle with which we design the estimator, referred to to as the Stochastic Search on the Line Based Discretized Weak Estimator (SSLDE), is based on the solution to the Stochastic Search on the Line (SSL) problem, pioneered by Oommen in [7].

It is pertinent to mention that the SSL solution presented in [7] is not directly relevant to the problem currently studied. The reason for this is the following: The method proposed in [7] explicitly assumes the existence of an Oracle which informs the learning mechanism whether to go "right" or "left". Such an Oracle is non-existent in the estimation problem. In our application domain, in order to resolve this paradox for estimation, we have to first *infer* (or rather *simulate*) such an Oracle. Thus, to overcome this difficulty, we first intelligently construct an "Artificial Oracle" that suggests whether we are to increase the current estimate or to decrease it.

The analytic results derived and the empirical results obtained demonstrate that the SSLDE estimator is able to cope with non-stationary environments with a high adaptation rate and accuracy, that, as one can expect, is dependent on the choice of the resolution parameter.

With regard to their applicability, apart from the problem being of importance in its own right, weak estimators admit a growing list of applications in various areas such as intrusion detection systems in computer networks [15], spam filtering [19], ubiquitous computing [6], fault tolerant routing [10], adaptive encoding [12], and topic detection and tracking in multilingual online discussions [14]. We ourselves are also considering the application of such estimators in mobile adaptive architectures.

2 State-of-the-Art

Traditionally available methods that cope with non-stationary distributions resort to the so-called *sliding window* approach, which is a limited-time variant of the well-known MLE scheme. The latter model is useful for discounting stale data in the stream of observations. Since the data samples arrive continuously, only the most recent observations are used to compute the current estimates. Thus, the data elements occurring outside the "current window" is forgotten and replaced by the new data. The problem with using sliding windows is the following: If the time window is too small the corresponding estimates tend to be poor. As opposed to this, if time window is too large, the estimates prior to the change of the parameter have too much influence on the new estimates. Moreover, the observations during the entire window width must be maintained and updated during the process of estimation.

Apart from the sliding window approach, many other methods have been proposed, which deal with the problem of detecting change points during estimation. In general, there are two major competitive sequential change-point detection algorithms: Page's Cumulative Sum (CUSUM) [2] detection procedure and the Shiryaev-Roberts-Pollak detection procedure. In [13], Shiryayev used a Bayesian approach to detect changes in the parameters of the distribution, where the change points were assumed to obey a geometric distribution. CUMSUM is motivated by a maximum likelihood ratio test for the hypotheses that a change occurred. Both approaches utilize the log-likelihood ratio for the hypotheses that the change occurred at the point, and that there is no change.

Inherent limitations of CUMSUM and the Shiryaev-Roberts-Pollak approaches for on-line implementation are the demanding computational and memory requirements. In contrast to the CUMSU and the Shiryaev−Roberts−Pollak approaches, our SSLDE avoids the intensive computations of ratios, and does not invoke hypothesis testing.

A description of the state-of-the-art would not be complete without mentioning the SLWE work of Oommen and Rueda [11] (cited above) which is based on the principles of *continuous* stochastic Learning Automata (LA). As opposed to this, our scheme resorts to *discretizing* the probability space [1,5,9,16], and performing a controlled random walk on this discretized space. It is well known in the field of LA that discretized schemes achieve faster convergence speed than continuous schemes [1,8]. By virtue of discretization, our estimator realizes fast adjustments of the running estimates by jumps, and it is thus able to robustly track changes in the parameters of the distribution after a switch has occurred in the environment.

A brief history of the science of discretization in the field of LA is not out of place. The concept of discretizing the probability space was pioneered by Thathachar and Oommen in their study on Reward-Inaction LA [16], and since then that it has catalyzed a significant research in the design of discretized LA [1,5,9,3,4]. Recently, there has been an upsurge of research interest in solving resource allocation problems based on novel discretized LA [3,4]. In [3,4], the authors proposed a solution to the class of *Stochastic Nonlinear Fractional Knapsack* problems where resources had to be allocated based on incomplete and noisy information. The latter solution was applied to resolve the web-polling problem, and to the problem of determining the optimal size required for estimation.

In a previous research work, the authors of this current work also devised a discretized weak estimator that is able to cope with non-stationary binomial and multinomial distributions. That estimate, referred to as the Stochastic Discretized Weak Estimator, has been analyzed and descibed elsewhere [17], and space does not permit us to submit a comprehensive comparison here. In all brevity, we mention that the latter scheme can be seen to be a more "fidel" counterpart of the continuous SLWE [11] since, in both algorithms (SLWE and SDWE), the mean of the final estimate is independent of the scheme's learning parameter.

We now proceed to present the new estimator, i.e., the Stochastic Search on the Line Based Discretized Weak Estimator (SSLDE).

3 The Estimator for Binomial Distributions

We assume that we are estimating the parameters of a binomial distribution. This distribution is characterized by two parameters, namely the number of trials and the parameter characterizing each Bernoulli trial. We assume that the number of observations is the number of trials. We seek to estimate the Bernoulli parameter for each trial.

Let X be a binomially distributed random variable, which takes on the value either "1" or "2". We choose to use these values instead of the more common

used notation "0" or "1" to make the notation consistent when we consider the multinomial case. It is assumed that the distribution of X is characterized by the parameter vector $S = [s_1, s_2]^{\mathrm{T}}$. In other words,

$X =$ "1" with probability s_1

$X =$ "2" with probability s_2, where $s_1 + s_2 = 1$.

Let $x(t)$ be a concrete realization of X at time 't'. We intend to estimate S, i.e, s_i for $i = 1, 2$. We achieve this by maintaining a running estimate of $P(t) = [p_1(t), p_2(t)]^{\mathrm{T}}$ of S where $p_i(t)$ represents the estimate of s_i at time t, for $i = 1, 2$. Our proposed SSLDE works in a discretized manner. In fact, we enforce the condition that $p_i(t)$ takes values from a finite set, i.e, $p_i(t) \in \{0, \frac{1}{N}, \frac{2}{N}, \ldots, \frac{N-1}{N}, 1\}$, where N is a user-defined integer parameter. N is called the "resolution" parameter and determines the stepsize Δ ($\Delta = \frac{1}{N}$) relevant to the updating of the estimates. A larger value of N will ultimately imply a more accurate convergence to the unknown parameter S. However, a smaller value of N will hasten the convergence rate, but will forfeit some accuracy. Initially, we assign $p_1(0) = p_2(0) = \frac{N}{2}$, where N is assumed to be an even integer. Let $r_1(t)$ be the state index of the estimator, implying that $p_1(t) = \frac{r_1(t)}{N}$. Thereafter, the value of $r_1(n)$, is updated as follows, depending on whether $r_1(t)$ is greater or less than $\frac{N}{2}$:

- **Case 1**: $r_1(t) \geq \frac{N}{2}$
  If $x(t) =$ "1" and $rand() \leq \frac{N}{2r_1(t)}$ and $r_1(t) < N$

$$r_1(t) := r_1(t) + 1 \tag{1}$$

  Else If $r_1(t) > 0$

$$r_1(t) := r_1(t) - 1 \tag{2}$$

  Else

$$r_1(t) := r_1(t) \tag{3}$$

- **Case 2**: $r_1(t) < \frac{N}{2}$
  If $x(t) =$ "2" and $rand() \leq \frac{N}{2(N - r_1(t))}$ and $r_1(t) > 0$

$$r_1(t+1) := r_1(t) - 1 \tag{4}$$

  Else If $r_1(t) < N$

$$r_1(t) := r_1(t) + 1 \tag{5}$$

  Else

$$r_1(t) := r_1(t), \tag{6}$$

where $p_2(t+1) = 1 - p_1(t+1)$ and $rand()$ is a uniform random number generator function. In the interest of simplicity, we omit the time index t, whenever there is no confusion, and thus, P implies $P(t)$.

The main theorem that we present concerns the convergence of the vector P which estimates S as per Equations (1 - 6). We claim that the mean of P converges exactly to S as N goes to infinity. In the interest of brevity, this proof is omitted here.

Theorem 1. *Let X be a binomially distributed random variable, and $P(t)$ be the estimate of S at time t obtained by Equations (1 - 6). Then:*

$$Lim_{N\to\infty} E[P(\infty)] \to S.$$

The proof of this result is quite involved and can be found in [18]. □

Remarks: A few remarks regarding our method for updating the estimates are not out of place. Indeed:

- First of all, it is pertinent to mention that although the rationale for updating is similar to that of the SSL algorithm [7], there are some fundamental differences. Unlike the latter, which explicitly assumes the existence of an "Oracle", in this case, our scheme simulates such an entity.
- Secondly, at this juncture, we emphasize that unlike the work of [7], the probability that the Oracle suggests the move in the correct direction, is not constant over the states of the estimator's state space. This is quite a significant difference, which basically reduces our model to a Markov Chain with state-dependent transition probabilities.
- The crucial issue is that by means of a random number generator function, we, hopefully elegantly, construct an *Artificial* Oracle that is **informative** as per the definition of [7], i.e, the Artificial Oracle's suggestions to either increase or decrease the estimate are correct with a probability always larger than 0.5. In that sense, whenever $p_1(t)$ is less than s_1, the Oracle directs the scheme to increase $p_1(t)$ with a probability greater than 0.5, and vice versa. The converse is true whenever $p_1(t)$ is greater than s_1.
- The main difference between our estimator and the SDWE, presented in [17], is that in the latter, the mean of the final estimate is independent of the scheme's learning coefficient, N. Consequently, we can say that the SDWE is a more fidel counterpart version of the SLWE, where the mean does not depend on *its* parameter, λ, as well. Moreover, in the case of the SDWE, at the internal states (i.e. $0 < r_1 < N$), there is a non zero probability that the estimate remains unchanged at the next time instant. As opposed to this, in our present updating scheme, the machine never stays at the same state at the next time instant, except at the end states. Therefore, it can be better characterized to be true to the essence of the SSL problem where the environment directs the LA to move to the right or to the left, and never directs it to stay at the same position.

4 Experimental Results

In this section, we evaluate the new family of discretzied estimators in non-stationary environments as well as in stationary environments. In the interest of brevity, we merely cite a few specific experimental results that highlight the salient properties of our approach.

4.1 Optimality Property

In this set of experiments, we experimentally verify the asymptotic optimality property of the estimator, as stated in Theorem 1. The results obtained have been recorded in Table 1, which summarizes the performance of the estimator, for a wide range of resolution parameters, N, and for two different values of s_1, namely $s_1 = 0.352$ and $s_1 = 0.827$. The resulting performance is reported in terms of the asymptotic true value of $E[p_1(\infty)]$, where $E[p_1(\infty)]$ is obtained using a single run experiment consisting of 10^7 iterations.

Note that an alternative manner to compute the true value of $E[p_1(\infty)]$ requires using its closed form expression. The reader will observe that by using a significant number of iterations (in our case, 10^7 iterations) the computations render the difference between the theoretical and experimental value of $E[p_1(\infty)]$ unobservable.

The experimental results confirm that the optimality property is valid. Indeed, $E[p_1(\infty)]$ consistently approaches s_1 as we increase the resolution. For example, for a resolution N equal to $1,000$, the final terminal value represents an error less than 0.002%.

Table 1. The true value of $E[p_1(\infty)]$ for various values of the resolution parameter, N, and for two different values of s_1

N	$s_1 = 0.352$	$s_1 = 0.827$
6	0.32	0.764
10	0.3371	0.798
60	0.3424	0.830
100	0.3472	0.8311
200	0.3496	0.8291
400	0.3503	0.8279
1000	0.3511	0.8277

4.2 Rate of Convergence

This set of experiments were conducted so that we could better understand the transient behavior of the chain associated with the estimator, and to thus perceive the rate of convergence of p_1 for different resolution configurations. To accomplish this, we fixed s_1 to be 0.9123, while we increased the memory N from $N = 6$ to $N = 40$. The quantity p_1 was then estimated by averaging it over $1,000$ experiments. In order to understand the effect of the resolution on the rate of convergence, we report the number of iterations required to reach a value that is 95% of the terminal value of $E[p_1(\infty)]$.

From Figure 1(a), we see that it took *only* 5 time instants for the algorithm to reach 95% of $E[p_1(\infty)]$ for a resolution $N = 6$. This, we believe, is remarkable. Further, from Figure 1(b), we see that 95% of $E[p_1(\infty)]$ was attained within 17 iterations when we set N to have the value 20. Similarly, for the results depicted in Figure 1(c), we chose N to be 30, where we see that it took *only* 25 time

instants to converge to 95% of $E[p_1(\infty)]$. Finally, in Figure 1(d), when we set N to 40, we record the required number of iterations was as low as 35! We believe that these outstanding results speak for themselves.

Observe that as we increased the memory, the estimator spent more time to converge to the optimal value of $E[p_1(\infty)]$, which, of course, is understandable.

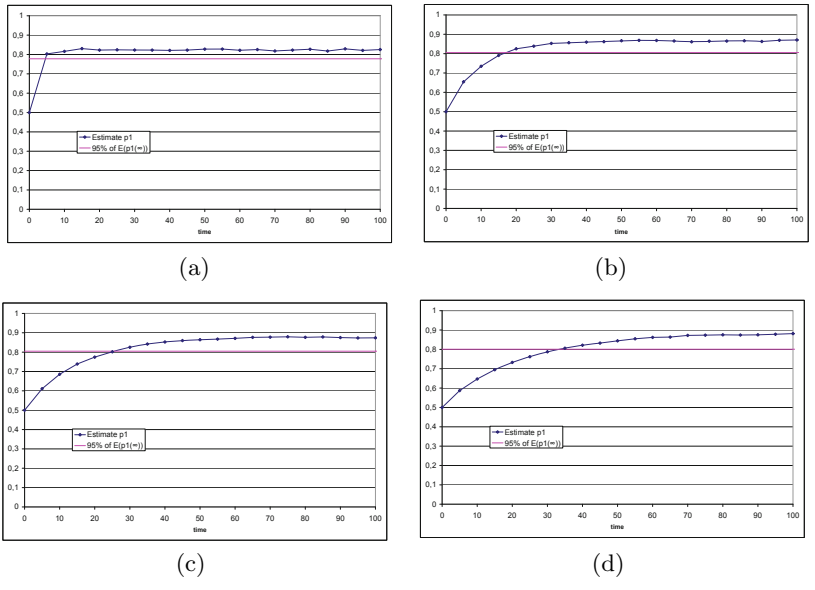
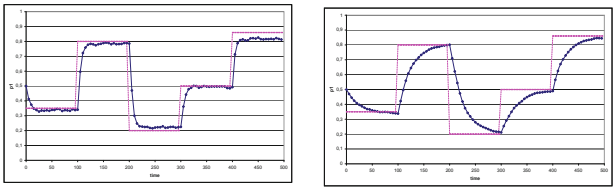


Fig. 1. This figure depicts the transient behavior of the chain as a function of time, where we plot (a) $p_1(t)$ for a memory size $N = 6$, (b) $p_1(t)$ for a memory size $N = 20$, (c) $p_1(t)$ for a memory size $N = 30$, and (d) $p_1(t)$ for a memory size $N = 40$

4.3 Performance in Dynamic Environments

In this experiment, we were interested in understanding the characteristics of the estimator when interacting with a non-stationary environment (i.e., when s_1 changed with time). To do this, we modeled a non-stationary environment by altering the parameter of the binomial distribution, s_1, at every 100th time slot.

In particular, the parameter s_1 that we used in the experiments was drawn sequentially every 100th time instant from the vector $R = [0.35, 0.8, 0.2, 0.5, 0.86]$. To be more specific, between time instants 0 and 100, s_1 was equal to 0.35, between instants 100 and 200, s_1 was equal to 0.8, and so on. In Figure 2(a), we have plotted the average value of p_1 over the set of experiments using a continuous line, and the target parameter s_1 using a discontinuous (dashed) line, when $N = 10$. Similarly, in Figure 2(b), we have reported the results of achieving the same (as in Figure 2(a)), except that, in this case, the resolution parameter was set as $N = 40$. From Figures 2(a) and 2(b), we observe that the instantaneous value of p_1 were well able to track the target distribution (drawn using the



(a) (b)

Fig. 2. Ability of the scheme to track the target distribution with (a) $N = 10$, and with (b) $N = 40$

discontinuous line) in a near-optimal manner, which we believe is quite fascinating! The use of this strategy to achieve time-varying testing, is obvious!

5 Conclusion

In this paper, we have presented a novel estimator, referred to as the Stochastic Search on the Line Based Discretized Weak Estimator (SSLDE), that is suitable for estimation in non-stationary environments. The design and foundations motivating the SSLDE are based on the principles of the pioneering solution of Oommen to the Stochastic Search on the Line problem [7]. By virtue of the SSLDE, we have shown that discretizing the probability space offers a new promising approach for the design of weak estimators. In fact, comprehensive simulation results demonstrate that the new estimator is able to cope with non-stationary environments with both a high adaptation rate and accuracy. To the best of our knowledge, this paper represents the first reported results that resolve *discretized* weak estimation using a SSL-based solution. The generalization of the scheme to handle multinomial distributions is currently being investigated.

References

1. Agache, M., Oommen, B.J.: Generalized pursuit learning schemes: New families of continuous and discretized learning automata. IEEE Transactions on Systems, Man, and Cybernetics 32(6), 738–749 (2002)
2. Basseville, M., Nikiforov, I.V.: Detection of abrupt changes: theory and application. Prentice-Hall, Inc. (1993)
3. Granmo, O.C., Oommen, B.J., Myrer, S.A., Olsen, M.G.: Learning automata-based solutions to the nonlinear fractional knapsack problem with applications to optimal resource allocation. IEEE Transactions on Systems, Man, and Cybernetics 37(1), 166–175 (2007)
4. Granmo, O.C., Oommen, B.J.: Solving stochastic nonlinear resource allocation problems using a hierarchy of twofold resource allocation automata. IEEE Transactions on Computers 59, 545–560 (2009)
5. Lanctôt, J.K., Oommen, B.J.: Discretized estimator learning automata. IEEE Transactions on Systems, Man, and Cybernetics 22(6), 1473–1483 (1992)

6. May, M., Berendt, B., Cornuéjols, A., Gama, J., Giannotti, F., Hotho, A., Malerba, D., Menesalvas, E., Morik, K., Pedersen, R., Saitta, L., Saygin, Y., Schuster, A., Vanhoof, K.: Research challenges in ubiquitous knowledge discovery. In: Next Generation of Data Mining, 1st edn. Chapman Hall Crc Data Mining and Knowledge Discovery Series, Chapman Hall CRC (2008)
7. Oommen, B.J.: Stochastic searching on the line and its applications to parameter learning in nonlinear optimization. IEEE Transactions on Systems, Man and Cybernetics 27B, 733–739 (1997)
8. Oommen, B.J., Agache, M.: Continuous and discretized pursuit learning schemes: Various algorithms and their comparison. IEEE Transactions on Systems, Man, and Cybernetics 31, 277–287 (2001)
9. Oommen, B.J., Hansen, E.: The asymptotic optimality of discretized linear reward-inaction learning automata. IEEE Transactions on Systems, Man, and Cybernetics 14(3) (May/June 1986)
10. Oommen, B.J., Misra, S.: Fault-tolerant routing in adversarial mobile ad hoc networks: an efficient route estimation scheme for non-stationary environments. Telecommunication Systems 44, 159–169 (2010)
11. Oommen, B.J., Rueda, L.: Stochastic learning-based weak estimation of multinomial random variables and its applications to pattern recognition in non-stationary environments. Pattern Recogn. 39(3), 328–341 (2006)
12. Rueda, L., Oommen, B.J.: Stochastic automata-based estimators for adaptively compressing files with nonstationary distributions. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 36(5), 1196–1200 (2006)
13. Shiryayev, A.N.: Optimal Stopping Rules. Springer (1978)
14. Stensby, A., Oommen, B.J., Granmo, O.-C.: Language Detection and Tracking in Multilingual Documents Using Weak Estimators. In: Hancock, E.R., Wilson, R.C., Windeatt, T., Ulusoy, I., Escolano, F. (eds.) SSPR&SPR 2010. LNCS, vol. 6218, pp. 600–609. Springer, Heidelberg (2010)
15. Tartakovsky, A.G., Rozovskii, B.L., Blazek, R.B., Kim, H.: A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods. IEEE Transactions on Signal Processing 54, 3372–3382 (2006)
16. Thathachar, M.A.L., Oommen, B.J.: Discretized reward-inaction learning automata. Journal of Cybernetics and Information Science, 24–29 (Spring 1979)
17. Yazidi, A., Granmo, O.-C., Oommen, B.J: A novel stochastic discretized weak estimator operating in non-stationary environments. To appear in the Proceedings of ICNC 2012, the 2012 International Conference on Computing, Networking and Communications, Hawaii, USA (2012)
18. Yazidi, A., Granmo, O.C., Oommen, B.J.: On the design of novel discretized weak estimators based on the stochastic search on the line solution. Unabridged version of this paper. To be Submitted for Publication
19. Zhan, J., Oommen, B.J., Crisostomo, J.: Anomaly detection in dynamic systems using weak estimators. ACM Trans. Internet Technol. 11, 3:1–3:16 (July 2011)

A Hierarchical Learning Scheme for Solving the Stochastic Point Location Problem

Anis Yazidi[1], Ole-Christoffer Granmo[1],
B. John Oommen[2,⋆], and Morten Goodwin

[1] Dept. of ICT, University of Agder, Grimstad, Norway
[2] School of Computer Science, Carleton University, Ottawa, Canada

Abstract. This paper deals with the Stochastic-Point Location (SPL) problem. It presents a solution which is novel in both philosophy and strategy to all the reported related learning algorithms. The SPL problem concerns the task of a Learning Mechanism attempting to locate a point on a line. The mechanism interacts with a random environment which essentially informs it, possibly erroneously, if the unknown parameter is on the left or the right of a given point which also is the current guess. The first pioneering work [6] on the SPL problem presented a solution which operates a *one-dimensional* controlled Random Walk (RW) in a discretized space to locate the unknown parameter. The primary drawback of the latter scheme is the fact that the steps made are always very conservative. If the step size is decreased the scheme yields a higher accuracy, but the convergence speed is correspondingly decreased.

In this paper we introduce the Hierarchical Stochastic Searching on the Line (HSSL) solution. The HSSL solution is shown to provide *orders of magnitude* faster convergence when compared to the original SPL solution reported in [6]. The heart of the HSSL strategy involves performing a controlled RW on a discretized space, which unlike the traditional RWs, is not structured on the line *per se*, but rather on a binary tree described by intervals on the line. The overall learning scheme is shown to be optimal if the effectiveness of the environment, p, is greater than the *golden ratio conjugate* [4] – which, in itself, is a very intriguing phenomenon. The solution has been both analytically analyzed and simulated, with extremely fascinating results. The strategy presented here can be utilized to determine the best parameter to be used in any optimization problem, and also in any application where the SPL can be applied [6].

Keywords: Stochastic-Point Problem, Discretized Learning, Learning Automata, Controlled Random Walk.

⋆ Chancellor's Professor; *Fellow: IEEE* and *Fellow: IAPR*. The Author also holds an Adjunct Professorship with the Dept. of ICT, University of Agder, Norway The author is grateful for the partial support provided by NSERC, the Natural Sciences and Engineering Research Council of Canada.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 774–783, 2012.

1 Introduction

Learning algorithms which operate within a known domain (for example, within the unit interval), generally work by moving from their current location to a new point within its *immediate* neighborhood. The step size of the migration depends on the learning parameter, where a smaller step size leads to a slower but more accurate convergence, and conversely, a larger step size leads to a faster but less accurate convergence. In this paper, we propose a completely novel strategy regarding how one can move around within the domain of interest. Rather than consider the domain as a *single* interval, we model it is a sequence of increasingly larger intervals mapped onto an underlying Binary Search Tree (BST). Thus, effectively, we are operating a Random Walk (RW) on a BST, where the steps of the RW need not necessarily be in the close proximity of the current point. The most incredible facet of this strategy is that the RW converges to the underlying optimum point – even if the environment providing the information to the learning algorithm is faulty or erroneous.

The problem we investigate is Stochastic Point Location (SPL) [6] problem which can be described as follows. Consider the problem of a robot (algorithm, Learning Mechanism (LM)) moving along the real line attempting to locate a particular point λ^*. To assist the mechanism, we assume that it can communicate with an Environment ("Oracle"), which guides it with information regarding the direction in which it should go. If the Environment is deterministic, the problem is the "deterministic point location problem", which has been studied rather thoroughly. In an initial version, Baeza-Yates *et al.* [1] presented the problem in a setting such that the Environment could charge the robot a cost that is proportional to the distance it is from the point sought for. The question of having multiple communicating robots locate a point on the line has also been studied by Baeza-Yates *et al.* [1,2]. In the stochastic version of this problem pioneered by Oommen [6,8,9], the LM attempts to locate a point in an interval with stochastic (i.e., possibly erroneous), instead of deterministic, responses from the Environment. Thus, when it should really be moving to the "right" it may be advised to move to the "left" and vice versa, with a nonzero probability.

Bentley and Yao [3] solved the deterministic point-location problem of searching in an unbounded space by examining points $f(i)$ and $f(i+1)$ at two successive iterations between which the unknown point lies, and by doing a binary search between them.

Unlike the work of Bentley and Yao [3], the faulty nature of the feedback from the Oracle in the SPL problem would render the latter task extremely intriguing and challenging. In fact, in [3], by virtue of the "correctness" nature of the feedback provided by the Oracle, the LM is able to confidently discard an entire interval from the search space each time it queries the Oracle. This approach would not be directly applicable to the SPL, as wrongly discarding a region that contains λ^* would mislead the LM. In this paper, we aim to design novel hierarchical learning schemes for solving the SPL problem that tolerate faulty feedback.

The main drawback of the state-of-the-art solution to the Stochastic-Point Location (SPL) problem reported in [6] is that the steps are always very conservative. If the step size is increased, the scheme converges faster, but the accuracy is correspondingly decreased. While if the step size is decreased the scheme yields a higher accuracy, but the convergence speed is correspondingly decreased.

In this paper we introduce the Hierarchical Stochastic Searching on the Line (HSSL) solution to the SPL problem. In the HSSL solution, as alluded to above, we structure the space as a binary tree. The algorithm then orchestrates a controlled RW on this space.

To informally clarify this, we preface our discussion by mentioning that for generations, the technique of searching a sorted list using "Binary Search" has been proven to outperform a controlled walk in the deterministic context. In the HSSL, this same principle has been extended to the SPL by discretizing the parameter space with a multilevel hierarchy, and performing the above-mentioned controlled RW on this space. The heart of the strategy involves performing controlled moves on a space structured as a tree, and then intelligently pruning the space using a hierarchical stochastic search. The rationale of the solution is to take advantage of the tree structure of the search space in order to enhance the search speed. This would enable the LM to quickly explore the search space and hopefully, focus its visits on the region that contains λ^*.

Apart from the problem being of importance in its own right, it also has potential applications in solving optimization problems. The details of this are omitted here, but can be found in[6].

2 Related Work

Oommen [6] proposed and analyzed an algorithm that operates on a discretized search space while interacting with an informative Environment ((i.e. $p > 0.5$). This algorithm takes advantage of the limited precision available in practical implementations to restrict the probability of choosing an action to only finitely many values from the interval $[0, 1]$, which also enables the analysis of the scheme.

The solution proposed in [6] for the SPL problem functions as follows. The space in which the search is conducted is first sliced by subdividing the unit interval into N steps $\{0, 1/N, 2/N, \ldots, (N-1)/N, 1\}$, and a larger value of N will ultimately imply a more accurate convergence to the unknown λ^*. The algorithm then invokes a controlled RW on this space. Whenever the LM is told to go to the right (left), it moves to the right (left) by a single step in this space.

The analytic results derived in [6] proved that if the "Oracle" was itself informative, the discretized RW learning was asymptotically optimal. Thus the mechanism would converge to a point arbitrarily close to the true value of λ^* with an arbitrarily high probability.

The primary drawback of the scheme described in [6] is the fact that the steps made are always very conservative. Thus, as stated above, if the step size is increased the scheme converges faster, but with a correspondingly less accuracy.

A novel strategy combining Learning Automata (LA) [5] and pruning was used in [8], which aims to search for the parameter in the continuous space

when interacting with an informative Environment. Utilizing the response from the Oracle, Oommen *et al.* partitioned the interval of search into three disjoint subintervals, eliminating at least one of the subintervals from further search and recursively searching the remaining interval(s) until the search interval is at least as small as the required resolution of estimation.

In a subsequent work [9], Oommen *et al.* introduced the Continuous Point Location with Adaptive d-ARY Search (CPL-AdS) which is a generalization of a portion of the work in [8]. In CPL-AdS, the given search interval is divided into d partitions representing d disjoint subintervals. In each interval, initially, the midpoint of the given interval was considered as the estimate of the unknown λ^*. Each of the d partitions of the interval is independently explored using an ϵ-optimal two-action LA, where the two actions are those of selecting a point from the left or right half of the partition under consideration. Then, the authors eliminated at least one of the subintervals from being searched further, and recursively searched the remaining pruned contiguous interval until the search interval is at least as small as the required resolution of estimation. This elimination process essentially utilizes the ϵ-optimality property of the underlying automata and the monotonicity of the intervals to guarantee the convergence. At each epoch consisting of a certain number, N_∞, of iterations, the algorithm "confidently" discarded regions of the search space.

In [7], Oommen *et al.* reported the first known solution to the stochastic point location (SPL) problem when the environment is non-stationary.

2.1 Contributions

Our paper presents a set of novel contributions summarized below:

- With regard to the design and analysis of discretized parameter schemes, we submit that a fundamental contribution of this paper is the manner in which we have designed the discretized search space, by structuring it as a balanced binary tree. Traditional approaches for discretization work by restricting the corresponding parameter to be one of finite number of values in the interval $[0, 1]$, and then a "one-dimensional" controlled RW is performed on the discretized space, where the transitions only occur between neighbor nodes, i.e to the "left" or to the "right". Instead, we propose a new philosophy for phenomenon of discretization *itself*, where the parameter space is structured as a binary tree. In brief, to each level of the tree, we associate a resolution that becomes finer at higher levels of the tree.
- The paper presents a significant contribution to the families of solutions relevant to the SPL problem.
- Extensive simulations results confirm that our scheme outperforms the state-of-the art discretized scheme [6]. We verify empirically that the proposed HSSL solution provides orders of magnitude faster convergence compared to the work reported in [6]. In addition, simulations results show that our scheme possesses an excellent ability to cope with nonstationary environments, both of which, we believe, are truly impressive!

– We report the first analytical results for HSSL and prove that the HSSL is asymptotically optimal. The analysis of the scheme is a contribution in its own right to the field of Markov Chains and LA.

3 Merging the Fields of Binary Search and SPL in HSSL

The algorithm we propose operates by invoking a controlled RW on a tree. The space of the search is arranged in the form of a binary tree with depth $D = log_2(N)$, where N, (which, for the sake of simplicity, is assumed to be a power of 2) is the resolution of the algorithm. The LM searches for the optimal value λ^* by traversing the tree, moving from one tree node to another. The way by which this is achieved in explained below.

3.1 Definitions

Construction of Hierarchy: Let $\Delta = [\sigma, \gamma)$ be the current search interval containing λ^* whose left and right (smaller and greater) boundaries on the real line are σ and γ, respectively. Without loss of generality, we initially assume that $\sigma = 0$ and $\gamma = 1$. The search space is constructed as follows: First of all, the hierarchy is organized as a balanced binary tree with depth D. To each node in the hierarchy we associate an interval. For convenience, we will index the nodes using their depth and their relative order with respect to the nodes situated at the same the depth.

Root Node: The root of the hierarchy (at depth 0), which we call $S_{\{0,1\}}$, is assigned the interval $\Delta = \Delta_{\{0,1\}} = [0, 1)$.

This interval is partitioned into two disjoint equisized[1] intervals $\Delta_{\{1,1\}}$ and $\Delta_{\{1,2\}}$, such that $\Delta_{1,1} = [0, 1/2)$ and $\Delta_{1,2} = [1/2, 1)$. Note that $1/2 = mid(\Delta_{\{0,1\}})$, where $mid(\Delta_{\{0,1\}})$ denotes the midpoint of $\Delta_{\{0,1\}}$. To avoid confusion, we shall use the notation[2] that refers to the interval $\Delta_{\{1,1\}}$ as the *Left Child* of the root and to $\Delta_{\{1,2\}}$ as its *Right Child*.

Nodes at depth d: The node $j \in \{1, ..., 2^d\}$ at depth d, is referred to as $S_{\{d,j\}}$ for $1 < d < D$. This node is assigned the interval $\Delta_{\{d,j\}} = [\sigma_{\{d,j\}}, \gamma_{\{d,j\}})$, which is associated with two disjoint equisized intervals $\Delta_{\{d+1,2j-1\}}$ and $\Delta_{\{d+1,2j\}}$.

Following the same previously alluded to nomenclature, $\Delta_{\{d+1,2j-1\}}$ is the *Left Child* of $\Delta_{\{d,j\}}$, and $\Delta_{\{d+1,2j\}}$ is its *Right Child*.

Nodes at depth D: At depth D, which represents the maximal depth of the tree, the nodes do not have children. In fact, when the search interval is at least as small as the required resolution of estimation, we do no further partitioning.

By virtue of the equi-partitioning property, for a given node j at depth d that is associated with the respective interval $\Delta_{\{d,j\}}$, we can deduce the values

[1] The equi-partioning is really not a restriction. It can be easily generalized.

[2] Indeed, we shall utilize the notations that *Parent*, *Left Child* and *Right Child* of an interval $\Delta_{\{i,j\}}$ in the binary tree are the intervals associated to the respective *Parent*, *Left Child* and *Right Child* of the node $S_{\{i,j\}}$.

of the left and right boundaries of the interval to be: $\sigma_{\{d,j\}} = (j-1)(\frac{1}{2})^d$ and $\gamma_{d,j} = j(\frac{1}{2})^d$, for $j \in \{1, ..., 2^d\}$ where $1 \leq d \leq D$.

Boundary Value Convention Regarding Notation: Since level "-1" is nonexistent, we use the notation appropriate for the boundary (basis case) condition, and denote the *Parent* of $\Delta_{\{0,1\}}$ to be $\Delta_{\{0,1\}}$ itself. The same comment is also valid for the root node. In other words: $Parent(\Delta_{\{0,1\}})=\Delta_{\{0,1\}}$.

Also, since level $D+1$ is nonexistent, we use the convention that the *Right Child* of a leaf node is the same as the leaf node itself. Similarly, the *Left Child* of a leaf node is the leaf node itself. Thus, formally, we call $Left\ Child(\Delta_{\{D,j\}})=Right\ Child(\Delta_{\{D,j\}})=\Delta_{\{D,j\}}$ for $j \in \{1, ..., 2^D\}$. The same notation applies as well to the leaf nodes $S_{\{D,j\}}$.

Target Node: We define the **Target** node as the leaf node whose associated interval contains λ^*.

Non-Target Nodes: The **Non-Target** nodes are leaf nodes whose corresponding associated intervals do not contain λ^*.

Resolution: Whenever the LM is at a certain node in the tree, we propose to use, as an estimate of the unknown λ^*, the middle point of the interval itself. By virtue of the equi-partitioning of the intervals at each level of the tree, whenever the LM is at node of a certain depth d in the tree, the estimate of the unknown λ^* will take a discretized value, a multiple of $(\frac{1}{2})^{d+1}$.

We call the resolution of the scheme, the number of leaf nodes, i.e $N = 2^D$.

3.2 Structure of the Search Space and Responses from the Environment

We intend to organize the search space in the form of a balanced binary tree, where each node corresponds to an interval range. Initially, we guess the midpoint of the given interval to be our estimate of the unknown λ^*. The LM searches for the optimal value λ^* by operating a RW on the tree, moving from one tree node to another, with the goal of locating the *target leaf node*. Each node in the tree is associated with an interval; e.g., the root is associated with the interval $[0, 1)$. This interval is partitioned into two disjoint equi-sized intervals. Thus, the left child of the root is associated with $[0, 1/2)$, the right child with $[1/2, 1)$, etc.

At any given time instance, let us assume that the LM finds itself at a node $S_{\{d,j\}}$ in the tree, where $j \in \{1, ..., 2^d\}$ and $1 \leq d \leq D$. The LM attempts to infer the next promising search interval that is likely to contain λ^* by making a sequence of "informed" guesses. For each guess, the Environment essentially informs the LM, possibly erroneously (i.e., with probability p), which way it should move to reach the unknown point. Let $\Delta_{\{d,j\}}$ be the interval that is associated with the node where the LM resides at the current time instant. The informed guesses correspond to a sampling at the boundary points of the interval: $\Delta_{\{d,j\}}$, *and* at the midpoint of the interval: $mid(\Delta_{\{d,j\}})$.

We formalize this by saying that the sampled points can be expressed as a vector $\overrightarrow{x} = [x^1, x^2, x^3]$, where $x^1 = \sigma_{\{d,j\}} = (j-1)(\frac{1}{2})^d$, $x^2 = mid(\Delta_{\{d,j\}}) = (2j-1)(\frac{1}{2})^{d+1}$ and $x^3 = \gamma_{\{d,j\}} = j(\frac{1}{2})^d$.

Further, let the corresponding respective responses from the Environment E be formulated as a tuple: $\overrightarrow{\Omega} = [\Omega^1, \Omega^2, \Omega^3]$.

Note that Ω^k, for $k \in \{1, 2, 3\}$, is a random variable that can take either the value Left or Right. Since the environment is assumed faulty, we suppose that it suggests the correct direction with a probability p. Therefore Ω^k, for $k \in \{1, 2, 3\}$ can be formally defined according to whether λ is bigger or smaller than x^k as:

If $\lambda < x^k$

$$\Omega^k = \begin{cases} L & \text{with probability } p \\ R & \text{with probability } (1-p), \text{and} \end{cases}$$

If $\lambda \geq x^k$

$$\Omega^k = \begin{cases} L & \text{with probability } (1-p) \\ R & \text{with probability } p, \end{cases}$$

where for simplicity, we use L to imply a region to the "left" of the sampled point, and R to imply a point to the "right" of the sampled point.

As a consequence of the above, it is easy to see that the overall effect of the Environment E is that it responds with one of the 2^3 possible results:

{[L, L, L], [L, L, R], [L, R, L], [L, R, R], [R, L, L], [R, L, R], [R, R, L], [R, R, L]}.

Based on these responses the LM moves to another node in the tree, either to the current node's parent, or to one of its children (*Left Child*/*Right Child*). The rules for moving in the tree are summarized in Table 1, and the formal result about the algorithm is stated in Theorem 1.

Table 1. Decision table to choose the next search interval based on the response vector $[\Omega^1, \Omega^2, \Omega^3]$, when the current search interval is $\Delta_{\{i,j\}}$

Next Search Interval	**Condition**
$Parent(\Delta_{\{i,j\}})$	[R, R, R] $\bigvee$ [L, R, R] $\bigvee$ [L, L, R] $\bigvee$ [L, L, L]
$LeftChild(\Delta_{\{i,j\}})$	[R, L, R] $\bigvee$ [R, L, L]
$RightChild(\Delta_{\{i,j\}})$	[R, R, L] $\bigvee$ [L, R, L]

Theorem 1: The parameter learning algorithm specified by the rules summarized in Table 1 is asymptotically optimal if p is bigger than the conjugate of the golden ratio[3]. Formally, $Lim_{N\to\infty} Lim_{n\to\infty} E[\lambda(n)] \to \lambda^*$.

In the interest of brevity, we omit the proof of Theorem 1. The proof is quite involved and can be found in [10]. □

[3] The golden ratio conjugate quantity is defined as $\Phi = \frac{\sqrt{5}-1}{2}$.

4 Simulation Results

In this section, we present the results which demonstrate the power of the HSSL for the SPL. In order to confirm the superiority of our scheme, we have conducted extensive simulations results under different parameter settings. However, in the interest of brevity, we merely cite a few specific experimental results.

In Table 4 we have recorded the true value of $E[\lambda(\infty)]$ for various values of p and the tree depth $D = log_2(N)$ (i.e, the resolution is $N = 2^D$) when the value of λ^* is 0.9123. The values of p are 0.7, 0.85, and 0.95.

The optimality property is empirically confirmed through the simulation. Observe that independent of whether the value of p is as low as 0.7 or as high as 0.95, $E[\lambda(\infty)]$ indefinitely approaches the optimal λ^* as we increase the resolution. Note that for a depth D which equals 12, the final terminal value represents an error less than 0.0005% for $p = 0.7$, $p = 0.85$, and $p = 0.95$.

We now report the results of the second set of experiments in which we have tried to catalogue the convergence of $E[\lambda(n)]$ with time, "n".

In order to obtain an understanding as to how the scheme converges with time, various simulations were conducted to evaluate the performance of the algorithm under a variety of constraints. In each simulation, 100 parallel experiments were conducted so that an accurate ensemble average of the results could be obtained. Again, although numerous experiments have been conducted, in the interest of brevity we shall merely report the results obtained for one set of experiments involving the unknown parameter λ^* switching periodically between the values 0.9123 and $1-0.9123$. We compared our results to the algorithm presented in [6]. Also, to be on the same level playing field, since there was no *apriori* information about the value λ^*, at time instant 0, we initialized the LM of the original SPL scheme to the position $\frac{N}{2}$, while the corresponding LM of the HSSL algorithm started from the root node.

In order to understand the effect of the resolution on the rate of convergence, we report the number of iterations required to reach a value that is 95% of the final value of λ^*.

In all brevity, we state that the HSSL algorithm outperformed the original SPL solution and learned the value of λ^* much faster. The experimental results obtained are truly conclusive.

In the first set of experiment, we fixed p to 0.8. The plots of the corresponding results are shown in Fig. 1(a) and Fig. 1(b). In Fig. 1(a), the resolution N was equal to 256 while λ^* switched every 400th iteration. In Fig. 1(b), the resolution N was equal to $1,024$, while λ^* switched every $1,500^{th}$ iterations.

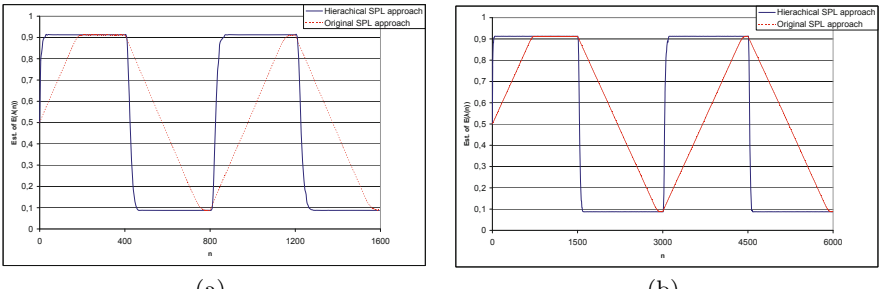
From Fig. 1(a), we see that in the first 400 iterations, it took the HSSL solution only 30 time instants to reach 95% of λ^*, while the original SPL solution required 180 iterations. After the first environment "switch", i.e between time instants 400 and 800, we observe that the convergence speed of both algorithms decreased slightly. In fact, 95% of λ^* was attained within 45 subsequent iterations in the case of the HSSL solution, while the original SPL solution took 350 iterations. Comparing the results of the first 400 iterations with that of the subsequent iterations, we conclude that although the final steady-state probabilities

are independent of the starting state, in reality, the time that the LM took to converge to λ^* is dependent on where one starts.

In Fig. 1(b), we show the results when we increased the resolution N to $1,024$. As in the previous case, from Fig. 1(b), we observe that in the first $1,500$ iterations, it took approximately only 50 iterations for our HSSL solution to reach 95% of the optimal value λ^*, while the original SPL solution spent 680 iterations. After the first environment switch, i.e between time instants 1500 and 3000, we observe that the convergence speed decreased. In fact, it took approximately 75 iterations for our HSSL solution to reach 95% of the optimal value λ^*, while the original SPL solution required $1,380$ iterations. In these settings, the HSSL approach provided an order of magnitude (i.e., 18) faster convergence than the original SPL solution. This, we believe, is impressive. We further remark that, as we increased the resolution N from 256 (see Fig. 1(a)) to 1024 (see Fig. 1(b)) for the same $p = 0.8$, the convergence speed of the original SPL solution was significantly reduced while the speed of the HSSL was less affected by this.

Table 2. True value of $E[\lambda(\infty)]$ for various values of p and various resolutions, when the value of λ^* is 0.9123

$log_2(N)$	$p = 0.7$	$p = 0.85$	$p = 0.95$
2	0.727906125	0.825522	0.866370875
3	0.82995875	0.9204060625	0.9337756875
4	0.86781290625	0.90972528125	0.9076381875
5	0.89240075	0.91925609375	0.921083703125
6	0.9056527421875	0.9150871328125	0.9144455
7	0.9069764765625	0.91111155859375	0.91035579296875
8	0.908265740234375	0.911884287109375	0.912022787109375
9	0.911209423828125	0.9128465126953125	0.9130258076171875
10	0.9116439086914062	0.912655470703125	0.9126211733398437
11	0.91219955078125	0.9124482912597657	0.912371505859375
12	0.9121167028808593	0.9122652081298828	0.9122371402587891



(a) (b)

Fig. 1. The learning characterstics of the HSSL and the original SPL algorithm when λ^* switches between the values 0.9123 and $1.0-0.9123$ every: (a) 400th iterations where $N = 256$ and $p = 0.8$, and (b) 1500th iterations where $N = 1024$ and $p = 0.8$

5 Conclusions

The SPL problem involves a LM that attempts to learn a parameter, for example, λ^*, within a closed interval. For each guess, the environment essentially informs the mechanism with three responses, each possibly erroneous (i.e., with probability p), on which way it should move to reach the unknown point. We have presented a solution that involves discretizing the space, mapping the discretized intervals onto a binary tree and performing a controlled random walk on this space. The solution we have presented has been both analytically analyzed and simulated, with extremely fascinating results. Apart from formally analyzing this algorithm, we have also experimentally demonstrated its superiority over the state-of-the-art. From this perspective, our approach has been shown to provide orders of magnitude faster convergence speed than the traditional SPL solution [6] in non-stationary environments, i.e., where λ^* changes over time.

References

1. Baeza-Yates, R., Culberson, J., Rawlins, G.: Searching with uncertainty. In: Proceedings of Scandinavian Workshop Algorithms and Theory (SWAT), Halmstad, Sweden, pp. 176–189 (1998)
2. Baeza-Yates, R., Schott, R.: Parallel searching in the plane. Comput. Geom. Theory Appl. 5, 143–154 (1995)
3. Bentley, J.L., Yao, A.C.C.: An Almost Optimal Algorithm for Unbounded Searching. Inform. Proc. Lett. 5, 82–87 (1976)
4. Livio, M.: The Golden Ratio: The Story of Phi, the World's Most Astonishing Number. Paw Prints (2008)
5. Narendra, K.S., Thathachar, M.A.L.: Learning Automata: An Introduction. Prentice-Hall, Inc. (1989), http://portal.acm.org/citation.cfm?id=64802
6. Oommen, B.J.: Stochastic searching on the line and its applications to parameter learning in nonlinear optimization. IEEE Transactions on Systems, Man and Cybernetics 27B, 733–739 (1997)
7. Oommen, B.J., Kim, S.W., Samuel, M., Granmo, O.C.: A solution to the stochastic point location problem in metalevel nonstationary environments. IEEE Transactions on Systems, Man, and Cybernetics 38(2), 466–476 (2008)
8. Oommen, B.J., Raghunath, G.: Automata learning and intelligent tertiary searching for stochastic point location. IEEE Transactions on Systems, Man and Cybernetics 28B, 947–954 (1998)
9. Oommen, B.J., Raghunath, G., Kuipers, B.: Parameter learning from stochastic teachers and stochastic compulsive liars. IEEE Transactions on Systems, Man and Cybernetics 36B, 820–836 (2006)
10. Yazidi, A., Granmo, O.C., Oommen, B.J., Goodwin, M.: Hierarchical stochastic searching on the line. Unabridged version of this paper (to be submitted for publication)

Discretized Bayesian Pursuit – A New Scheme for Reinforcement Learning

Xuan Zhang[1], Ole-Christoffer Granmo[1], and B. John Oommen[1,2,⋆]

[1] Dept. of ICT, University of Agder, Grimstad, Norway
[2] School of Computer Science, Carleton University, Ottawa, Canada

Abstract. The success of Learning Automata (LA)-based estimator algorithms over the classical, Linear Reward-Inaction (L_{RI})-like schemes, can be explained by their ability to pursue the actions with the highest reward probability estimates. Without access to reward probability estimates, it makes sense for schemes like the L_{RI} to first make large exploring steps, and then to gradually turn exploration into exploitation by making progressively smaller learning steps. However, this behavior becomes counter-intuitive when pursuing actions based on their estimated reward probabilities. Learning should then ideally proceed in progressively *larger* steps, as the reward probability estimates turn more accurate. This paper introduces a new estimator algorithm, the *Discretized Bayesian Pursuit Algorithm* (DBPA), that achieves this. The DBPA is implemented by linearly discretizing the action probability space of the Bayesian Pursuit Algorithm (BPA) [1]. The key innovation is that the linear discrete updating rules mitigate the counter-intuitive behavior of the corresponding linear continuous updating rules, by augmenting them with the reward probability estimates. Extensive experimental results show the superiority of DBPA over previous estimator algorithms. Indeed, the DBPA is probably the fastest reported LA to date.

Keywords: Learning Automata, Pursuit Schemes, Bayesian Reasoning, Estimator Algorithms, Discretized Learning.

1 Introduction

Learning Automata (LA) have been widely studied as a "bare bones" model of reinforcement learning. The fastest LA algorithms to date are members of the family of estimator algorithms, initially pioneered by the Pursuit Algorithm (PA) [2], and more recently by the Bayesian Pursuit Algorithm (BPA) [1]. Both PA and BPA pursue the action currently perceived to be the optimal one. The main difference lies in the strategy used in inferring which action is to be considered "optimal". Whereas the PA maintains Maximum Likelihood (ML) estimates to decide which action is currently the best, the BPA utilizes a Bayesian method to achieve a superior estimation. By providing *optimistic* reward probability estimates, the latter estimation approach leads learning in a relatively "more correct" direction, thus making the learning converge faster. In this paper, we present a new algorithm – the Discretized Bayesian Pursuit Algorithm (DBPA),

⋆ Chancellor's Professor; *Fellow: IEEE* and *Fellow: IAPR*. The Author also holds an *Adjunct Professorship* with the Dept. of ICT, University of Agder, Norway.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 784–793, 2012.

which is implemented by linearly discretizing the action probability space of the BPA. To the best of our knowledge, DBPA is the fastest reported LA to date.

1.1 Learning Automata and Their Applications

A Learning Automaton (LA) is an adaptive decision-making unit that learns the best action out of a set of actions, offered by an environment. At each iteration, the LA chooses one action, which is either rewarded or penalized by the environment as a response. The response, in turn, affects which action the LA chooses in the next iteration. The optimal action is defined as the one with the highest probability of being rewarded.

The beauty of an LA is that it learns the optimal action through interaction with the environment, without any prior knowledge about it. The environment can be treated as a "black box". Indeed, independent of the intricacies exhibited by the behavior of the environment, the LA, considering only the responses it receives, learns to choose the best action, and adapts itself to the environment.

Learning criteria for evaluating an LA involve two basic aspects. The first of these is its accuracy, i.e., to what degree the LA is able to adapt itself to the optimal action. The other is its rate of convergence, i.e., how much time it takes for the LA to converge. In the pursuit of designing accurate and fast LA, a number of algorithms have been both proposed and studied. Initial LA, which had time invariant transition and decision functions are considered to be Fixed Structure Stochastic Automata (FSSA). The Tsetlin, Krylov and Krinsky automata [3] are the most notable examples of this type. Later, Variable Structure Stochastic Automata (VSSA) were developed, which can be completely characterized by the function that updates the probability of choosing the actions [3]. Earlier representatives of this type include the Linear Reward-Penalty (L_{R-P}) scheme, the Linear Reward-Inaction (L_{R-I}) scheme and Linear Inaction-Penalty (L_{I-P}) scheme [3]. Of these, the L_{R-I} scheme is the most accurate and the fastest, as it favors rewards over penalties. Thathachar and Sastry [2] were the first to introduce the concept of Pursuit Algorithms (PA), initiating the research on estimator algorithms. As an estimator algorithm, the PA utilizes the Maximum Likelihood (ML) method for estimating the reward probabilities. More recently, in [4], Granmo presented the Bayesian Learning Automata (BLA), which uses Bayesian reward probability estimation, and subsequently bases its exploration on the Thompson sampling principle [5]. Inspired by the performance benefits of the BLA and the PA, the Bayesian Pursuit Algorithm (BPA) was proposed in [1], following the concept of pursuing the currently-estimated best action, while utilizing Bayesian principles in the estimation itself. In the majority of environments in which LA were tested, the BPA outperformed its previous competitors.

LA have found applications in a variety of fields. They have been used in game playing [6,7], parameter optimization [8,9], vehicle path control [10], channel selection in cognitive radio networks [11], assigning capacities in prioritized networks [12], and resource allocation [13]. LA have also been used in natural language processing, string taxonomy [14], graph patitioning [15], and map learning [16].

1.2 Contributions and Paper Organization

In this paper, we propose a new LA algorithm, namely, the Discretized Bayesian Pursuit Algorithm (DBPA). Firstly, the DBPA maintains an action probability vector for

selecting actions. Secondly, it follows the concept of pursuing the action currently inferred to be the "best" . Thirdly, at any time instant, the DBPA uses Bayesian estimates to infer which action is the best. Finally, the DBPA updates its action probability vector according to linear discretized rules. In DBPA, the combination of Bayesian estimation for reward probabilities augmented with linear discretized updates makes learning converge approximately 20% faster than the previously-recorded algorithms including the BPA, as our extensive experimental results demonstrate.

As opposed to our previous paper on the BPA [1], this present paper also analyzes the performance of the BPA in a stricter and more challenging manner, namely, by comparing the performance of the BPA and the PA under their individual optimal learning rates. The advantage of the BPA over the PA thus becomes more evident and persuasive.

The key innovation of this paper is that it points out the incongruity existing in continuous estimator algorithms. Typically, in estimator algorithms, since the estimation of the reward probability is less accurate initially, large changes in the action probabilities at the beginning should be considered as unwise, and almost reckless. Besides, it is also counter-intuitive and unnecessary to reduce the size of the learning steps as the learning proceeds, since as this happens, the reward probability estimates also get more accurate. Unfortunately, continuous estimator algorithms, utilizing linear continuous updating rules for the action probabilities, operate exactly in this manner. The DBPA, on the other hand, uses linear discretized updating rules to mitigate this incongruity.

The paper is organized as follows. In Section 2, we give an overview of the PA, DPA and BPA, as they are the most related algorithms from the family of estimator algorithms. In Section 3, we analyze the incongruity existing in continuous estimator algorithms, and then present the new LA algorithm - the Discretized Bayesian Pursuit - by discretizing the probability space of the BPA. Section 4 provides extensive experimental results showing the advantages of the DBPA over the BPA, and demonstrates that the BPA is truly superior to PA under their individual optimal learning rates. Finally, Section 5, reports opportunities for further research and submits concluding remarks.

2 Related Work

In this section, we briefly review three typical algorithms from the family of estimator algorithms, namely, the Pursuit Algorithm (PA), the Discretized Pursuit Algorithm (DPA), and the Bayesian Pursuit Algorithm (BPA).

The PA, DPA and BPA share a common "pursuit" paradigm of learning – in each iteration, they pursue the action currently perceived to be optimal. Firstly, they maintain an action selection probability vector $P = [p_1, p_2, ..., p_r]$, with $\sum_{i=1}^{r} p_i = 1$ and r being the number of actions. The question of which action is to be selected is decided by sampling from P, which is, initially, uniform. Secondly, they maintain a reward probability estimate vector $\hat{D} = [\hat{d}_1, \hat{d}_2, ..., \hat{d}_r]$, with $\hat{d}_i$ $(i = 1, 2, ..., r)$ being the estimate of the reward probability for Action i. $\hat{D}$ is updated each time an action is selected and a response of either a reward or a penalty is received. At each iteration, the LA treats the action currently possessing the highest reward probability estimate $\hat{d}$ as the optimal one. Finally, based on the inference of the optimal-action and the response from the environment, these LA update the action probability vector P according to a Linear

Reward-Inaction rule [3]. The result is that the action selection probability of the optimal action increases and the other action probabilities decrease. The updated action selection probabilities, P, thus come into effect in the next round of action selection. Subtle differences between the PA, DPA and BPA result in fine performance benefits.

Pursuit Algorithm: The PA utilizes the Maximum Likelihood (ML) method to estimate the reward probability of each action. The ML reward probability estimate $\hat{d}_i$ can be calculated as $\hat{d}_i = \frac{W_i}{Z_i}$, with Z_i being the number of times Action i has been selected, and W_i the number of times Action i has been rewarded.

The PA updates the action probabilities according to the linear continuous rules:

- If selecting an action results in a reward:
  $p_j = (1-\lambda) \times p_j$, $j \neq i$
  $p_i = 1 - \sum_{j \neq i} p_j$, where i is the index of the action with the largest element in $\hat{D}$.
- If selecting an action results in a penalty: $\forall j, p_j = p_j$.

Discretized Pursuit Algorithm: The DPA is implemented by following the paradigm of the PA with the action probability space being discretized. If we denote the distance between two neighbor states as δ, the DPA updates the $\{p_i\}$ as per the discretized rules:

- If selecting an action results in a reward:
  $p_j = max\{p_j - \delta, 0\}$, $j \neq i$
  $p_i = 1 - \sum_{j \neq i} p_j$, where i is the index of the action with the greatest element in $\hat{D}$.
- If selecting an action results in a penalty: $\forall j, p_j = p_j$.

Bayesian Pursuit Algorithm: The BPA also follows the paradigm of a general PA. However, as opposed to using the ML estimates as in the PA, the BPA utilizes Bayesian estimates for reward probability estimation. The Bayesian estimation is based on the *Beta Distribution*, which is the conjugate prior for the Bernoulli distribution. A 95% percentile value of the posterior is chosen as the appropriate estimate. The latter is calculated by means of the respective cumulative distribution $F(x_i; a, b)$:

$$F(x_i; a, b) = \frac{\int_0^{x_i} v^{a-1}(1-v)^{b-1}\, dv}{\int_0^1 u^{a-1}(1-u)^{b-1}\, du}, \quad x_i \in [0,1]. \tag{1}$$

where a and b are two positive parameters determining the shape of *Beta Distribution*.

By virtue of the *Beta Distribution*, the Bayesian estimation is implemented in a computationally simple way.

3 Discretized Bayesian Pursuit Algorithm

The initial motivation for discretizing LA was to increase their rate of convergence and to avoid the need for generating real numbers with arbitrary precision [17]. In continuous algorithms, since the action probability is updated by multiplying a constant $1-\lambda$, the probability of selecting the optimal action can *only* be approached asymptotically to unity, but never be *actually* attained. However, in discrete algorithms, a minimum step

size is obtained by discretizing the probability space, and the updating of the action selection probabilities is achieved by subtracting or adding one or a multiple of the step size. If the automaton is close to the end state, the probability of the optimal action will be increased directly to unity with a few more favorable responses.

In this paper, we state another important reason for discretizing *estimator algorithms*. As can be readily understood, in the early learning period, the reward probabilities will be inaccurate due to the small number of samples available. As learning proceeds, however, the number of samples increases, and hence the reward probability estimates become progressively more accurate. Thus, it is unwise to make large action selection probability changes initially, and correspondingly counter-intuitive to update the action selection probabilities with smaller and smaller increments, as more samples are obtained. Unfortunately, in continuous algorithms, action selection probabilities are updated exactly in this non-intuitive manner. Indeed, the size of the update increments varies with the action selection probability vector itself, tending to be greater earlier on, and smaller as the learning progresses. In other words, when the estimation of the reward probabilities cannot reliably discriminate between the actions, "large" changes are made to the action selection probabilities, causing the algorithm to be affected by a "gravitational" pull towards an inferior action. This is tacitly caused by the action selection mechanism, since large action selection probabilities translate into biases towards the corresponding actions. The LA can thus either fail to converge to the best action or invest effort into pursuing an incorrect action, leading to a reduced convergence rate.

Discretized LA are characterized by their discrete action probability space. They are linear if the probability values are spaced equally in the interval $[0,1]$, otherwise, they are called nonlinear [18,19]. In this paper, we study linear discrete algorithms, where the equally divided state space indicates that the change of action selection probabilities is a fixed value. When one compares the "large-to-small" changes in continuous algorithms, the fixed-value of the changes in discrete algorithms are more reasonable as per the accuracy of estimating the reward probabilities.

With the above reasoning in mind, we improve the latest estimator algorithm – the Bayesian Pursuit Algorithm – by discretizing its action probability space. The new algorithm, given below, is the Discretized Bayesian Pursuit Algorithm (DBPA).

Algorithm: DBPA
Parameters:
α: The action selected by LA.
p_i: The ith element of the action selection probability vector, P.
a_i, b_i: The two positive parameters of the *Beta* distribution for Action i.
$\hat{d}_i$: The ith element of the Bayesian estimates vector $\hat{D}$, given by the 95% upper bound of the cumulative distribution function of the corresponding *Beta* distribution.
m: The index of the maximal component of the reward probability estimates vector $\hat{D}$.
R: The response from the environment, where $R = 0$ (reward) or $R = 1$ (penalty).
δ: The minimum step size.

Initialization:
1. $p_i(t) = 1/r$, where r is the number of actions.
2. Set $a_i = b_i = 1$. Initialize a_i and b_i, by doing Step 1 and Step 2 in "Method" below a small number of times. (i.e., in this paper $10 * r$ times).

Method:
For t:=1 to N **Do**

1. Pick $\alpha(t)$ randomly as per the action selection probability vector $P(t)$. Suppose $\alpha(t) = \alpha_i$.
2. Based on the Bayesian nature of the conjugate distributions, update $a_i(t)$ and $b_i(t)$ according to the response from the environment:
   If $R(t) = 0$ **Then** $a_i(t) = a_i(t-1) + 1; b_i(t) = b_i(t-1)$;
   Else $a_i(t) = a_i(t-1); b_i(t) = b_i(t-1) + 1$;
3. Identify the upper 95% reward probability bound of $\hat{d}_i(t)$ for each action i as:
$$\frac{\int_0^{\hat{d}_i(t)} v^{(a_i-1)}(1-v)^{(b_i-1)}dv}{\int_0^1 u^{(a_i-1)}(1-u)^{(b_i-1)}du} = 0.95$$
4. Update the action selection probability vector $P(t+1)$ according to the following linear discretized rule:
   If $R(t) = 0$ **Then**
   $p_j(t+1) = max\{p_j(t) - \delta, 0\}$, $j \neq m$
   $p_m(t+1) = 1 - \sum_{j \neq m} p_j(t+1)$.
   Else
   $P(t+1) = P(t)$.

End Algorithm: DBPA

Briefly speaking, the DBPA maintains an action probability vector P for selecting actions, runs *Bayesian* reward probability estimates to determine the current best action, and updates the action probabilities as per *linear discretized* rules. The combination of Bayesian estimation and linear discretized updating rules allow the LA to converge in a relatively more correct direction, and thus, achieve a higher rate of convergence.

4 Experimental Results and Analysis

In this section, we evaluate the computational efficiency of the DBPA by comparing it with the latest estimator algorithm, the BPA, as well as the PA and the DPA mentioned in Section 2. The computational efficiency is characterized by the rate of convergence, i.e., the average number of iterations it takes for an algorithm to converge to the optimal action. Extensive experiments have been conducted based on the experimental configurations listed in Table 1.

4.1 Experiment Design

In the experiments considered, Configurations 1, 4 and 7 form the simplest environments, possessing a low reward variance and a large difference between the reward probabilities of the actions. By reducing the differences between the actions, we increase the learning difficulty of the environment. Configurations 2, 5 and 8 achieve this task. The challenge of Configurations 3, 6 and 9 is their high variance combined with the small differences between the actions.

In order to evaluate the algorithms under fair conditions, the experiments were designed by considering the following:

1. To keep the conditions identical, each algorithm sampled all actions ten times each in order to initialize the estimate vector. These extra iterations are also included in the results.
2. As each of the four algorithms depends on an external learning rate, the optimal learning rate has to be found for each algorithm.

3. In each learning experiment, the LA is considered to have converged if the probability of selecting an action is greater than or equal to the threshold 0.999. If the LA converges to the best action, it is considered to have converged correctly.
4. For each algorithm, in each configuration, 750 independent learning experiments were conducted. The optimal learning rate λ or δ is the largest one that achieves 100% accuracy, i.e., all the 750 experiments converge correctly.
5. For these configurations, an ensemble of 100 independent replications with different random number streams was performed to minimize the variance of the optimal learning rate.
6. The algorithms were compared with each other as per their individual optimal learning rates.

Table 1. Bernoulli distributed rewards used in 2-action, 4-action and 10-action configurations

Config./Actions	1	2	3	4	5	6	7	8	9	10
1	0.90	0.60	-	-	-	-	-	-	-	-
2	0.90	0.80	-	-	-	-	-	-	-	-
3	0.55	0.45	-	-	-	-	-	-	-	-
4	0.90	0.60	0.60	0.60	-	-	-	-	-	-
5	0.90	0.80	0.80	0.80	-	-	-	-	-	-
6	0.55	0.45	0.45	0.45	-	-	-	-	-	-
7	0.90	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60
8	0.90	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80
9	0.55	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45

4.2 Experimental Results

Table 2 represents the average number of iterations that each algorithm required to converge to the best action in 2-action, 4-action and 10-action configurations, respectively.

Table 2. The average number of iterations for the LA to converge in different configurations

Conf./Alg.	**PA**	**DPA**	**BPA**	**DBPA**
Conf. 1	76.6040	61.0680	52.7465	54.0889
Conf. 2	891.2050	454.1912	491.1346	361.8910
Conf. 3	1890.3317	929.5736	1007.0200	699.3099
Conf. 4	165.3456	122.4167	110.3835	98.2777
Conf. 5	2110.6593	1042.7655	981.5411	693.5873
Conf. 6	4487.0094	2238.2541	2072.5094	1401.9219
Conf. 7	479.1508	517.0978	465.7514	510.9999
Conf. 8	6138.0661	3601.4218	3248.8014	2711.5099
Conf. 9	13154.1980	7553.5158	5860.3236	4998.4137

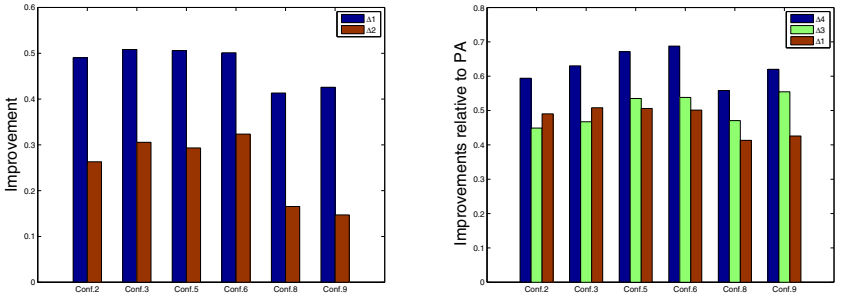
As can be seen from the table, in the simplest configurations of Configurations 1, 4 and 7, the discretized algorithms do not necessarily outperform their continuous counterparts. For example, on the average, the DBPA spent 54 steps to converge to the best action, while the BPA spent only 52 steps. At the same time, it took, on average, 76 steps for the PA to converge correctly, while only 61 steps it took for the DPA to achieve its goal. Besides, discrete algorithms get better results than their continuous counterparts

in 4-action configurations, but in 10-action configurations, continuous algorithms outperform the discrete algorithms. This can be explained by the simplicity of the learning problems associated with the environments. Due to the combination of a low reward variance and a large difference between the reward probabilities of the actions, both ML estimates and Bayesian estimates are able to provide good estimates for the reward probabilities. Therefore, the fixed change in the action probabilities in discrete algorithms yields no evident advantage over the great-to-small changes of action probability in continuous algorithms, and thus the performance of the discrete algorithms is similar to that of their continuous counterparts. Thus, by virtue of the simplicity of the environment, all the four algorithms converge to the best action very quickly.

As a result of the above, our focus is on the relatively more difficult configurations, namely, Configurations 2, 3, 5, 6, 8 and 9. The results in Table 2 show that the DBPA is 26% faster than the BPA in Configuration 2, and 31% faster in Configuration 3. For example, in Configuration 2, the DBPA converges, on the average, in 362 steps, and the BPA needs 491 steps, on average, to converge, which shows an improvement of 26%. In Configuration 3, the DBPA requires, on average, 699 steps for convergence while the BPA requires 1007 steps. The improvement is remarkable, i.e., 31%.

Similarly, the results in Table 2 also present the improvement of the DBPA over the BPA being up to 29% in Configuration 5, 32% in Configuration 6, 17% in Configuration 8 and 15% in Configuration 9. The superiority of the DBPA to its counterpart is clear!

One can also observe that the DPA is obviously faster than the PA in these configurations. If we denote the advantage of the DPA over PA by $\Delta 1$ and the advantage of the DBPA over the BPA by $\Delta 2$, the comparison of $\Delta 1$ and $\Delta 2$ is shown in Fig. 1(a).



(a) Comparison between $\Delta 1$ and $\Delta 2$ (b) Comparison between $\Delta 1$, $\Delta 3$ and $\Delta 4$

Fig. 1. The relative improvements of different algorithms

As the reader will observe, $\Delta 1$ is consistently greater than $\Delta 2$ in these configurations. The results are consistent with our statement in Section 3: In estimator algorithms, when the estimation of the reward probabilities is not accurate enough, updating the action probability in a linear discretized manner will be superior than in a linear continuous manner. The reason is as follows. As compared with Bayesian estimates, the PA uses the less accurate ML estimates for the reward probabilities. Therefore, the linear continuous manner of updating action probabilities in the PA is more likely to lead the learning into a wrong direction, resulting in postponed correct convergence. The DPA, on the other

hand, with its linearly discretized action probability space, mitigates the problem caused by linear continuous action probability updating. However, in the BPA, as the Bayesian estimates already are able to provide relatively accurate reward probability estimates, the improvement by discretizing its action probability space becomes less significant.

In order to evaluate the four estimator algorithms in a comprehensive manner, we set the performance of the PA as a benchmark, and compared their individual performance to the PA. Fig. 1(b) demonstrates the performance of the algorithms relative to the PA, where $\Delta 3$ and $\Delta 4$ represent the advantages of the BPA and the DBPA over the PA.

The result of $\Delta 3$ shows clearly that the BPA is superior to the PA, being approximately 50% faster than the latter. The improvement is by virtue of the superiority of Bayesian estimation to ML estimation. The results demonstrated in Fig. 1(b) also show that the DBPA is the best algorithm among the four estimator algorithms, being approximately 60% faster than the PA. The improvement is due to the combination of Bayesian estimation and the linearly discretized action probability space.

Based on the experimental results and analysis, we draw the following conclusions:

1. Considering the average number of steps required to attain the same accuracy of convergence in the listed configurations, the *Discretized Bayesian Pursuit Algorithm* is the best algorithm among the four estimator algorithms.
2. By evaluating the performance of the algorithms under their individual optimal learning rates, the superiority of the Bayesian Pursuit Algorithm over the Pursuit Algorithm is evident and persuasive.
3. In estimator algorithms, updating the action probabilities in a linear discretized manner is more reasonable than in a linear continuous manner. The extensive experimental results warrant this assertion.

5 Conclusions and Future Work

LA have been studied for decades and a plenitude of algorithms have been proposed, with the BPA being the most recent and fastest one. Although the BPA is superior to previous estimator algorithms, its action selection probabilities are updated in a linear continuous manner, which is counter-intuitive in the light of the demands for initial caution and later confidence, dictated by the accuracy of the reward probability estimates.

In this paper, we have introduced a new algorithm called the Discretized Bayesian Pursuit Algorithm (DBPA). The DBPA is implemented by discretizing the action selection probabilities of the BPA. Since its estimation for reward probabilities is Bayesian, and since it updates the action selection probabilities in discretized equal-sized steps, more aligned with the accuracy of estimates, the DBPA is able to achieve an even higher rate of convergence than the BPA.

This paper also presented a comprehensive comparison between the four estimator algorithms. Besides, it evaluated the performance of the BPA by comparing it with the PA under fair and reasonable conditions. The results confirm that while the BPA is inferior to the DBPA, it is still superior to the PA. Thus, to the best of our knowledge, the DBPA is the fastest reported LA to date.

A linearly discretized action selection probability space represents an intuitive manner of mitigating the incongruity between linear continuous updating and the mechanisms of reward probability estimation. However, the question of how to mitigate the

incongruity to the largest extent, making updating rules work consistently with the estimation mechanism, remains open. Besides, introducing the state-of-the-art DBPA algorithm to various fields of application also opens promising avenues of research.

References

1. Zhang, X., Granmo, O.-C., Oommen, B.J.: The Bayesian Pursuit Algorithm: A New Family of Estimator Learning Automata. In: Mehrotra, K.G., Mohan, C.K., Oh, J.C., Varshney, P.K., Ali, M. (eds.) IEA/AIE 2011, Part II. LNCS (LNAI), vol. 6704, pp. 522–531. Springer, Heidelberg (2011)
2. Thathachar, M., Sastry, P.: Estimator algorithms for learning automata. In: The Platinum Jubilee Conference on Systems and Signal Processing, Bangalore, India, pp. 29–32 (1986)
3. Narendra, K.S., Thathachar, M.A.L.: Learning Automat: An Introduction. Prentice Hall (1989)
4. Granmo, O.: Solving two-armed bernoulli bandit problems using a bayesian learning automaton. International Journal of Intelligent Computing and Cybernetics 3(2), 207–234 (2010)
5. Thompson, W.R.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. Biometrika 25, 285–294 (1933)
6. Lakshmivarahan, S.: Learning Algorithms Theory and Applications. Springer, New York (1981)
7. Oommen, B.J., Granmo, O.C., Asle, P.: Using Stochastic AI Techniques to Achieve Unbounded Resolution in Finite Player Goore Games and its Applications. In: IEEE Symposium on Computational Intelligence and Games 2007, Honolulu, HI (2007)
8. Narendra, K.S., Thathacha, M.A.L.: Learning Automata. Prentice-Hall, Englewood Cliffs (1987)
9. Beigy, H., Meybodi, M.R.: Adaptation of parameters of BP algorithm using learning automata. In: Sixth Brazilian Symposium on Neural Networks 2000, JR, Brazil (2000)
10. Unsal, C., Kachroo, P., Bay, J.S.: Multiple stochastic learning automata for vehicle path control in an automated highway system. IEEE Trans. on Sys., Man, and Cybern., Part A 29, 120–128 (1999)
11. Song, Y., Fang, Y., Zhang, Y.: Stochastic Channel Selection in Cognitive Radio Networks. In: IEEE Global Telecommunications Conference, Washington DC, USA, pp. 4878–4882 (2007)
12. Oommen, B.J., Roberts, T.D.: Continuous learning automata solutions to the capacity assignment problem. IEEE Trans. on Computers 49, 608–620 (2000)
13. Granmo, O., Oommen, B.J., Myrer, S.A., Olsen, M.G.: Learning automata-based solutions to the nonlinear fractional knapsack problem with applications to optimal resource allocation. IEEE Transactions on Systems, Man, and Cybernetics, Part B 37(1), 166–175 (2007)
14. Oommen, B.J., Croix, T.D.S.: String taxonomy using learning automata. IEEE Trans. on Sys., Man, and Cybern. 27, 354–365 (1997)
15. Oommen, B.J., Croix, T.D.S.: Graph partitioning using learning automata. IEEE Trans. on computers 45, 195–208 (1996)
16. Dean, T., Angluin, D., Basye, K., Engelson, S., Aelbling, L., Maron, O.: Inferring finite automata with stochastic output functions and an application to map learning. Maching Learning 18, 81–108 (1995)
17. Oommen, B.J., Lanctot, J.K.: Discretized pursuit learning automata. IEEE Trans. on Sys., Man, and Cybern. 20, 931–938 (1990)
18. Oommen, B.J.: Absorbing and ergodic discretized two-actio learning automata. IEEE Trans. on Sys., Man, and Cybern. SMC-16, 282–296 (1990)
19. Oommen, B.J., Thathachar, M.A.L.: Discretized reward inaction learning atuomata. Journal of Cybernetics and Information Science, 24–29 (1979)

A Method of Software Defects Mining Based on Static Analysis*

Xiaochen Lai[1,2], Kuanjiu Zhou[3], Likun Li[4], Lemin Tang[4], Yanshuang Yao[4], and Liang Yu[4]

[1] Faculty of Electronic Information and Electrical, Dalian University of Technology, Dalian, China
[2] School of Software, Dalian University of Technology, Dalian, China
far.away@tom.com
[3] School of Software, Dalian University of Technology, Dalian, China
zhoukj@dlut.edu.cn
[4] School of Software, Dalian University of Technology, Dalian, China
{lilikun,tanglemin,yaoyanshuang,yuliang}@mail.dlut.edu.cn

Abstract. Software defects are easy to cause when programming by C++ language, because of its features of flexibility and complexity, as well as its large number of undefined behaviors. According to "MISRA C++ 2008" safe subset, a method of software defects mining is raised based on static analysis technology. Source files can be converted into XML intermediate files, while rules in safe subset are expressed by XQuery expressions. And then match each rule to XML intermediate files to find the location of defects in source files. The experimental result of the prototype system shows that the software defects conflicting to safety rules can be mined effectively with low false alarm rate and low false negative rate.

Keywords: software defects, mining, static analysis, extensible markup language, rules matching.

1 Introduction

Software safety is one of the evaluations of software trustworthiness [1]. CVE(Common Vulnerabilities and Exposures), the international authority of the vulnerability disclosure Organization, indicated that the software safety vulnerabilities, represented by buffer overflows and integer overflows, greatly affected the trustworthiness of software [2]. Software defects should be mined and removed during all phases of software life cycle, especially the coding phase with lowest cost. The C++ programming language contains a large number of undefined behaviors that can easily cause serious safety risks if not utilized properly. Software testing, especially the

* Supported by "the Fundamental Research Funds for the Central Universities" under Grant DUT12JR03, and "the Fundamental Research Funds for the Central Universities" under Grant No. 1600-893321.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 794–807, 2012.

method of static analysis, is an effective way to mine the inherent weaknesses of programming languages and improve the accuracy of the program[3-4]. It merely analyzes source code instead of executing the program, which makes it easier to design automatic tools of defect detection [5]. At present, many source code static analysis tools exist [6], most of which, however, are only designed for C source code, such as lint, Split, MOPS[7-9], etc.

Defining language safe subset is an effective way to improve the safety of the program. Using only the grammar permitted in subset strictly can reduce the safety risks in large part. The static analysis tool based on safe subset is used to locate the defects in source code automatically, where rules in safe subset are violated, so as to help programmers to mine software vulnerabilities and improve the safety of the program. Among common safe subsets, "GJB 5369-2005 aerospace software safe subset in C" and "MISRA C 2004" are famous ones based on C language, while CSV[10] is a dedicated static analysis tool for the former safe subset. Proposed by the motor industry software reliability association in European, "MISRA C + + 2008" is a C++ safe subset based on the C++ standard "ISO / IEC 14882:2003". It is a standard to improve software safety in C++ programming, but so far there are no static source code analysis tools supporting for this safe subset.

All the safety rules in "MISRA C + + 2008" safe subset are analyzed and the method on how to design the corresponding defects mining tool is proposed, including source code parsing part, source code expression model and rules matching part. The source code expression model is the core of the system, which defined a standard that can represent all elements of C++ syntax. It can convert any C++ source files into an internal form of XML intermediate file which is suitable for rules matching. The XML intermediate file is not only the output of source code parsing, but also the input of rules matching. As for each C++ source file, syntax directed parsing is carried out by source code parsing part, in order to generate the XML intermediate file according to the definition of source code expression model. The rules matching part is responsible for building the safe knowledge library, matching each rule in the database to the XML intermediate file and locating the match point in source files and generating of the final defects report. Testing by the prototype system named CPSD (CPp Source code Detector) shows statements that violate the specifications of safety subset in a C++ source program can be mined effectively with a smaller false alarm rate and false negative rate.

2 Source Code Expression Model

The flexibility and complexity of C++ language determines the fact that it is hard to directly match the source files with safety rules. Instead, designing a model easy to represent and store information of source files is of great importance. XML is an ideal language for design of the source code expression model, which contains 3 levels: XML file model, XML statement model and XML identifier model.

2.1 XML File Model

The XML file model, defined by XML Schema, corresponds to C++ source files. It is an alternative representation of code that associated with the rules in safety subset in source files. Figure 1 illustrates the structure of XML file model. Each C++ source file corresponds to a SOURCEFILE node in the XML file, which is the top level node. It has 2 attributes and 4 child nodes, described as follows:

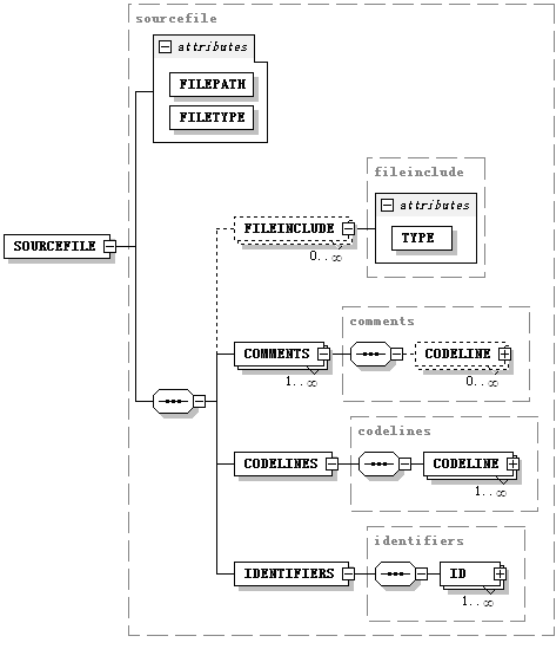


Fig. 1. The structure of file model defined by XML Schema

- FILEPATH: The attribute node saves file path, specifying the absolute path of the file;
- FILETYPE: The attribute node saves file type, indicating whether the file is a C++ source file or a C++ header file;
- FILEINCLUDE: The child node stores the text of include statement, which has a attribute of TYPE to distinguish between different types of header files, such as system header file or user-defined header file;
- COMMENTS: The child node records all the comments in source files. Each block of comment corresponds to one CODELINE child node;
- CODELINES: The child node describes the organization of statements in C++ source files. CODELINES is the main part of the XML file. Each statement in source files corresponds to a CODELINE child nodes belongs to CODELINES;

- IDENTIFIERS: The child node records the variable names, function names, class names, label information and all other identifiers. Each identifier corresponding to an ID child node which belongs to IDENTIFIERS.

2.2 XML Statement Model

XML statement model, defined by XML Schema shown in Figure 2, is the representation of statements in C++ source files that contains the semantic and context information of statement. Each statement in C++ source files corresponds to a CODELINE node containing 2 attribute and 2 child nodes, described as follows:

- CODETYPE: The attribute node indicates the type of statement. It has 91 possible values, such as ASM, FOR, CATCH, THROW and PUBLIC, etc.
- LINENUMBER: The attribute node indicates the line number in source file.
- CODETEXT: The child node contains the statement text of the line in source code;
- SUBLINES: The optional child node is suitable only for compound statement, containing several expanded inline statement, each of which is corresponding to a CODELINE child node. The CODELINE of Simple statement and declaration statement does not contain SUBLINES.

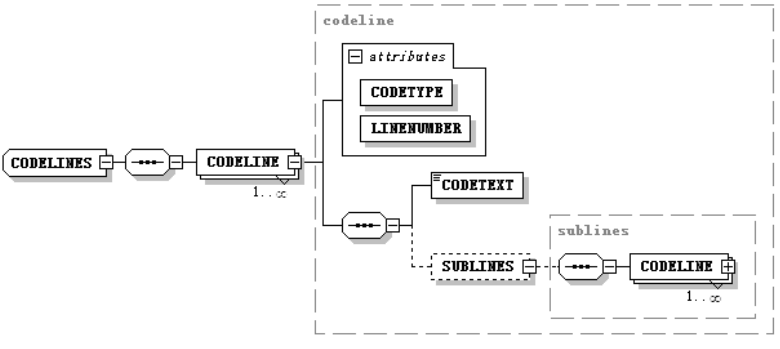


Fig. 2. The structure of statement model defined by XML Schema

2.3 XML Identifier Model

C++ language supports multiple types of identifiers. Generally, an identifier has a variety of attributes, and the model must include all of them. In XML identifier model, each identifier defined in source files corresponds to the child node ID, belonging to node IDENTIFERS. Attributes and child nodes of ID record the information of identifier, such as identifier name, identifier type, storage type, assignment records and other information. The XML Schema model of identifier is shown as Figure 3, which each identifier includes 2 attributes and 14 optional child nodes, described as follows:

- IDTYPE: The type of identifier, with a series of option value up to 35 categories, such as variable attribute(VARIABLE), function definitions(FUNCTIONDEF), class(CLASS), inline function declaration(INLINE_FUN_DEC), class template(TEMPLATE_CLASS).
- LINENUMBER: The attribute to record the line number where identifier appears in source files for the first time.

There are 14 optional child nodes of identifier. Whether the child nodes exist or not is determined by the attribute IDTYPE and the context environment where the identifier appeared in source files. These optional nodes are: identifier name (NAME), identifier type (TYPE), storage type (STORETYPE), assignment records (ASSIGNMENT), reference record (REFERLOG), constants (CONST), access label (ACCESS), structure belonging (OWNER), virtual function mark (VIRTUAL), operator overloaded (OPERATE), parameters (PARAMETER), field (FIELD), base class (PARENT), template (TEMPLATE). Line 10 to 12 of Figure 4 shows class D derived from a virtual base class named B, with the corresponding XML statement model shown as Figure 5 and the XML identifier model shown in Figure 6.

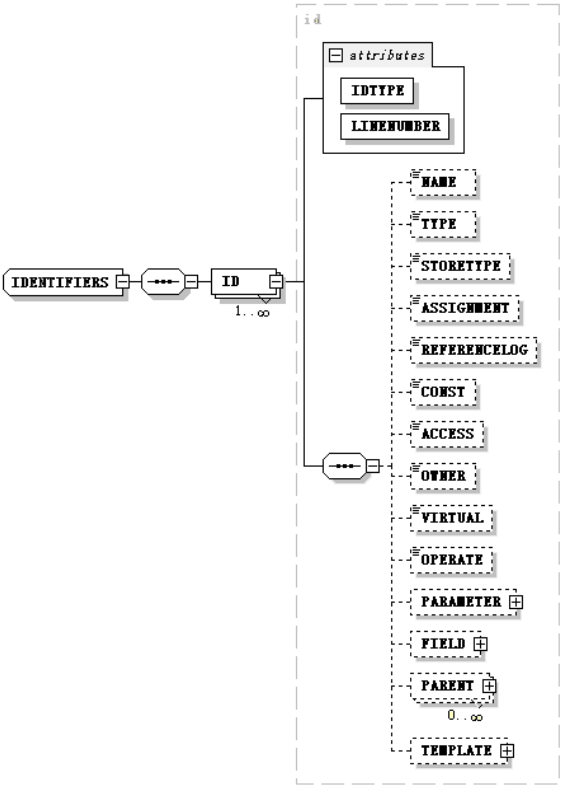


Fig. 3. The structure of identifier model defined by XML Schema

```
0:      namespace NS
1:      {
2:          class B
3:          {
4:              string str;
5:              B()
6:              {
7:                  str=”test”;
8:              }
9:          };
10:         class D:public virtual B
11:         {
12:         };
13:     }
```

Fig. 4. The example of deriving from virtual base class

```
0:  <CODELINE CODETYPE="CLASS_DEF" LINENUMBER="10">
1:    <CODETEXT>class D:public virtual B</CODETEXT>
2:    <SUBLINES>
3:      <CODELINE CODETYPE="LEFTBRACE" LINENUMBER="11">
4:        <CODETEXT>{</CODETEXT>
5:      </CODELINE>
6:      <CODELINE CODETYPE="RIGHTBRACE" LINENUMBER="12">
7:        <CODETEXT>}</CODETEXT>
8:      </CODELINE>
9:    </SUBLINES>
10: </CODELINE>
```

Fig. 5. The XML statement model in example of deriving from virtual base class

```
0:  <ID IDTYPE="CLASS_DEF" LINENUMBER="10">
1:    <NAME>D</NAME>
2:    <TYPE>class</TYPE>
3:    <PARENT>
4:      <ID IDTYPE="CLASS" LINENUMBER="10">
5:        <NAME>B</NAME>
6:        <TYPE>class</TYPE>
7:        <ACCESS>public</ACCESS>
8:        <VIRTUAL>virtual</VIRTUAL>
9:      </ID>
10:   </PARENT>
11: </ID>
```

Fig. 6. The XML identifier model in example of deriving from virtual base class

3 Source Code Parsing Method

Based on source code expression model, proposed the source code parsing method to convert the C++ source files into XML intermediate file, in order to extract the information relevant to safety rules. Research shows that the GXL profile generated from intrinsic abstract syntax tree of compiler contains a large amount of redundant information supporting compiling process[11-13], which can interfere with static analysis tool. Therefore, the method is not suitable for generating XML intermediate file. The static analysis tool itself is responsible for parsing source files.

The structure of source code parsing part is shown in Figure 7. The tasks, including handling headers, substituting macros, analyzing alias of data types, handling preprocessor statement, and converting single-line comment with continuation character into multi-line comments, are carried out by preprocessing sub-module. The lexical analyzer and syntax parser sub-module generates source file syntax tree, which is then converted into a memory image whose structure is suitable to generate XML intermediate file. The memory image is passed to XML generating sub-module and written into XML intermediate file. During this process, the difficulties lie in the design of syntax tree model and storage model of memory image.

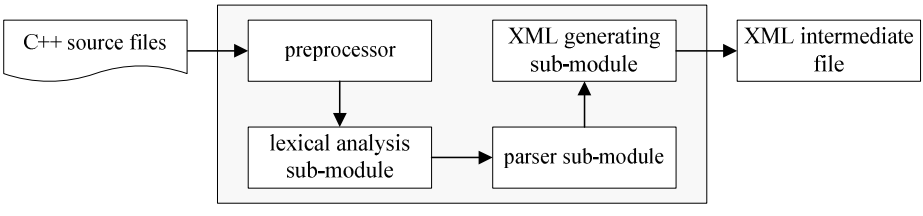


Fig. 7. The architecture of source code parsing

3.1 The Syntax Tree Model Based on Relational Storage Mode

Syntax tree is a hierarchical relationship model, based on modeling the relationship between nodes, containing between father and son nodes the included relationship and the brotherly relationship between nodes of same level. During the process of Syntax tree building, the level of parents is smaller than children 1 while brothers have the same level. In a compound statement block, statements or identifiers at the same level have the same parent node. According to siblings, the line number of left brother is smaller than the right. Based on the above principle, construct the STMT tree recording the statement and statement context information in source files and the ID tree recording the identifier information. The STMT tree and the ID tree of code snippet in Figure 4 are shown in Figure 8(a) and (b) respectively.

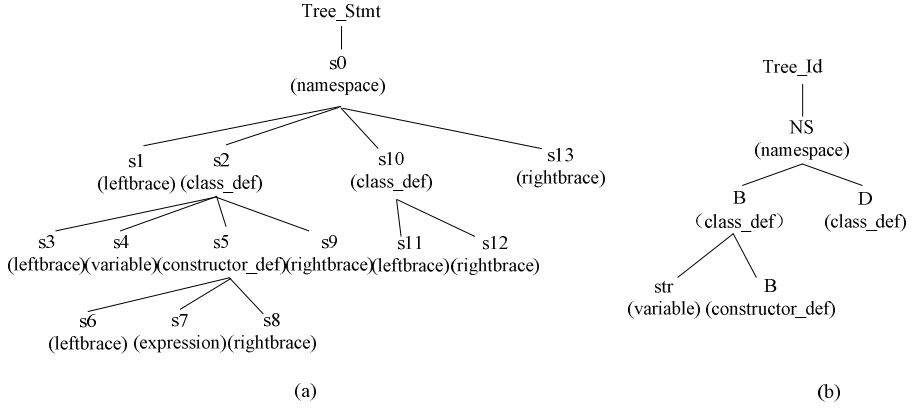


Fig. 8. The STMT tree and ID tree in example of deriving from virtual base class

3.2 Storage Model of Memory Image

Syntax tree nodes and their relationships storing the information of source code, the storage model of memory image must be constructed, recording the whole feature of syntax tree, in order to generate the XML intermediate file. The two data structures, Symbol and Table, are designed to store the syntax tree node information and the relationships between nodes.

Several Table nodes make a Table graph, which can be classified to 2 categories. One class is used to organize the Symbol nodes on behalf of text of the statements. The other class is used to organize the Symbol nodes on behalf of identifiers. Each Table object runs the collection of Symbol object at the same level. The definition of data structure Table is shown in Figure 9.

```
class Table
{
   int level;
   Table* previous;
   Table* next;
   Table* sibling;
   Symbol* firstSym;
   Symbol* lastSym;
   ENTRYOFSYMBOL* buckets[HASHSIZE];
          ……      //The data manipulation part is Omitted.
};
```

Fig. 9. The definition of data structure Table

The process of Table node construction carried out with the analysis on the statement. A Table node is constructed when scanning a block of statements for the first time. A node can be connected to several sub-nodes, the parent node's next pointer pointing to the first child and first child node's sibling pointer pointing to the successor of sibling child nodes, while all of these sub-nodes points to parent node with previous pointer. Figure 10 shows the Table storage model of STMT tree shown in Figure 8(a); Figure 11 shows the Table storage model of ID tree shown in Figure 8(b).

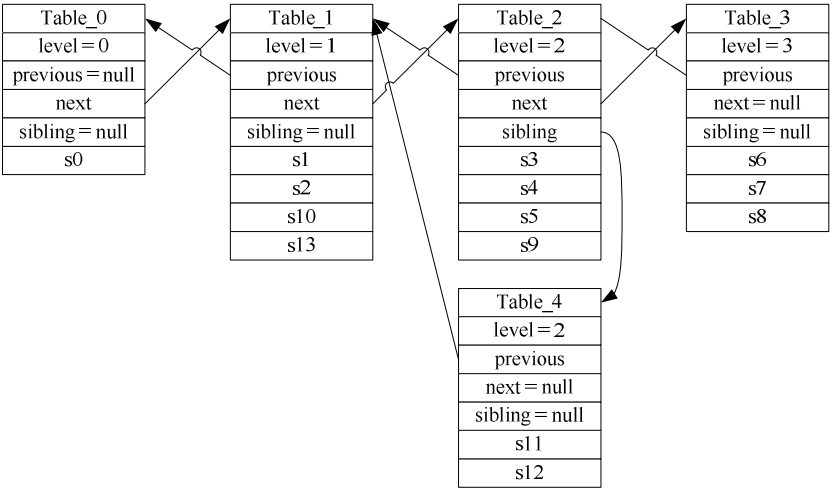


Fig. 10. The Table storage model of STMT tree in example of deriving from virtual base class

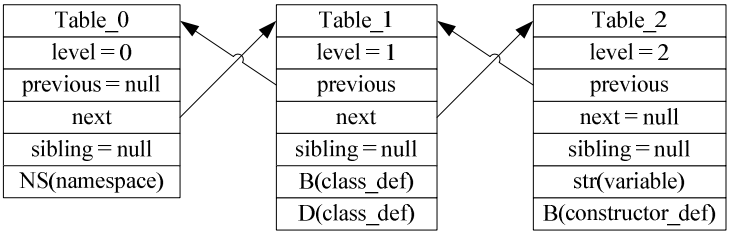


Fig. 11. The Table storage model of ID tree in example of deriving from virtual base class

Symbol objects are divided into two categories: one is used to describe the statement line, corresponding to the CODELINE of source code expression model, and the other used to describe the identifier corresponding to the ID of source code expression model. All the Symbol objects of statements in source code are pointed sequential by the next pointer of Symbol structure. All the Symbol objects of identifiers in source code are pointed sequential by the next pointer of Symbol structure too. Symbol structure is shown in Figure 12.

```
class Symbol
{
private:
        string name;
        int lineno;
        string symtype;
        int level;
        Type* type;
        Symbol* next;
        Symbol* fpara;
        Symbol* tpara;
        Symbol* parent;
        string sclass;
        string symAccess;
        string symOwner;
        ……      //The data manipulation part is Omitted.
};
```

Fig. 12. The definition of data structure Symbol

The Symbol storage structure of STMT tree in Figure 8(a) is shown in Figure 13, which the fpara, tpara, parent pointers are omitted because the values are null. The Symbol storage structure of ID tree in Figure 8 (b) is shown in Figure 14.

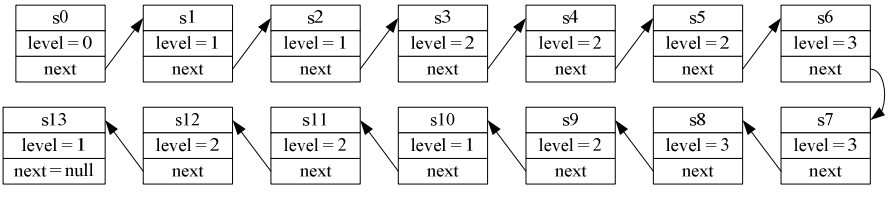


Fig. 13. The Symbol storage model of STMT tree in example of deriving from virtual base class

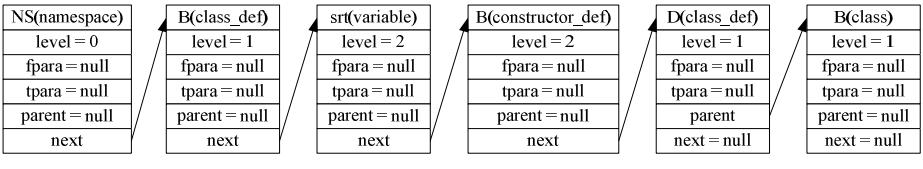


Fig. 14. The Symbol storage model of ID tree in example of deriving from virtual base class

After getting the Symbol and Table storage structure of STMT tree, source code parsing part calls XML generating sub-module to write the two storage structures into XML intermediate file, as the CODELINE node shown in Figure 5. Similarly, the Symbol and Table storage structure of ID tree are written to XML intermediate file, as the ID node shown in Figure 6.

4 Rules Matching Method

The architecture of rules matching part is shown in Figure 15. XML intermediate file is the input. Safe knowledge library contains rules of "MISRA C + + 2008" safety subset. Rules configuration sub-module contains several schemes of rules, each corresponding to a subset of safe knowledge library with its attributes configurable by users. Rules adding sub-module enables users to add their own rules to safe knowledge library, thus makes the library scalable. Relocation sub-module backtracks all the matching points to line numbers of source files and write the line numbers and the rules numbers to defects report. Rules matching engine matches the rules of current scheme to the XML intermediate file to find all matching points. Rules matching part mainly consists of rules classification, rules modeling, safe knowledge library designing and rules adding mechanism.

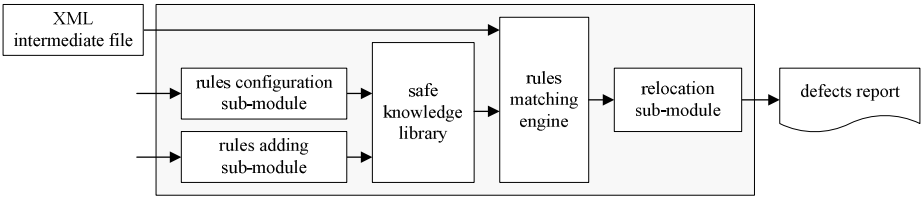


Fig. 15. The architecture of rules matching method

Safe knowledge library is a set of safety rules based on the XML storage model too. XQuery expressions are used to design storage model for rules. The XML Schema storage model of safe knowledge library is shown in Figure 16. Each safety rule is a RULE node, which contains one attribute and two child nodes.

- ID: Attribute of rule number, that is, number of the rule in the "MISRA C + + 2008" safe subset.
- ERRORINFO: Child node of rule information, providing the text description of the rule, which is the error information user needed.
- XQUERY: Child node of query statement, which is an XQuery expression describing applicable object of safety rules and a set of constraints of rules.

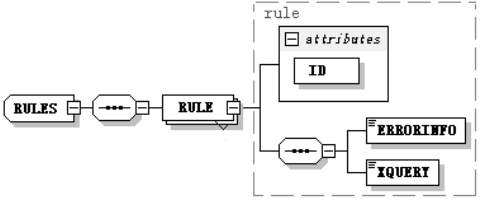


Fig. 16. The storage model of safety rules defined by XML Schema

The XQuery storage model of rule of 6-10-1-1 in safe knowledge is shown in Figure 17.

```
<RULE>
  <ID>6-A-1-1</ID>
  <ERRORINFO>Do not derive from virtual base class</ERRORINFO>
  <XQUERY>declare variable $file external; for $class in doc($file)/SOURCEFILE/
    IDENTIFIERS//ID where $class/TYPE="class" for $parent in $class/PARENT/ID
    where $parent/VIRTUAL='virtual' return data($parent/@LINENUMBER)
  </XQUERY>
</RULE>
```

Fig. 17. The XQuery storage model of safety rule 6-10-1-1

The adoption of XML and XQuery technology in design of safe knowledge library ensure the openness of the system, which particularly beneficial to the addition of new rules to system, expanding the safe knowledge library. Thus, the system is no longer confined to "MIRSA C + + 2008". When introducing new rules, as long as the ID numbers and descriptions are assigned, design the XQuery expressions of the new rules and then add them to safe knowledge library.

5 Prototype System

According to the principle above, CPSD, a prototype of source code defects mining tool based on "MISIA C + + 2008" safe subset is designed. To verify the practical performance of the prototype system, 10 files are selected randomly from the source package of QT, the C++ graphical user interface library. Enable all the safety rules for testing, and the results are shown in Table 1:

Table 1. The testing result of prototype system

Classification of rules	Defects existed	Defects reported	False negative	False alarm	False negative rate	False alarm rate
Declaration and definition	45	44	1	0	2.2%	0
Programming style	58	58	0	0	0	0
Flow control	37	36	1	0	2.7%	0
Preprocessing	22	22	0	0	0	0
Variables and functions	55	56	1	2	1.8%	3.6%
Computing and processing	24	23	2	1	8.3%	4.2%

Test results shows that the prototype system supports the first 4 categories of rules well with a low false negative rate and false alarm rate, because the design of XQuery expressions is not very hard. Taking the rules 6-10-1-1 mentioned previously for example, the prototype system only need to detect whether there is the keyword "virtual" in source files, thus no complex logic operations are involved. The accuracy of the last 2 categories is relatively low, because there are more constraints and the relationships between constraints are complex. Moreover, these rules involve not only statements, but also context information related to them, which makes it more difficult to design proper XQuery expressions. The main task of next step is to better the XQuery expressions to reduce the false alarm rate and false negative rate for some complex rules.

6 Conclusions

A static analysis method based on "MISRA C + + 2008" safe subset is presented in order to mine software defects in coding stage and reduce testing cost. Practical test shows that the prototype system can effectively mine the majority of defects violating the rules of safe subset with a low false alarm rate and false negatives rate, thus proves the method feasible.

References

1. Chen, H., Wang, J., Wei, D.: High Confidence Software Engineering Technologies. Acta Electronica Sinica 31(12), 1933–1938 (2003)
2. Christey, S., Martin, R.A.: Vulnerability Type Distributions in CVE. The MITRE Corporation, 1–38 (2007)
3. Hoare, C.A.R.: The verifying compiler: A grand challenge for computing research. Journal of the ACM 50(1), 63–69 (2003)
4. Zhang, J.: Sharp Static Analysis of Programs. Chinese Journal of Computers 31(9), 1549–1553 (2008)
5. Cui, Z., Wang, L., Li, X.: Target-Directed Concolic Testing. Chinese Journal of Computers 34(6), 953–964 (2011)
6. Hwang, J.G., Jo, H.J., Kim, B.H., Jeong, R.G.: Development of Automatic Testing Tool for Software Coding Rules for Railway Signalling. In: IEEE T&D Asia (2009)
7. Chess, B., McGraw, G.: Static analysis for security. IEEE Security & Privacy (6), 67–69 (2004)
8. Larochelle, D., Evans, D.: Statically Detecting Likely Buffer Overflow Vulnerabilities. In: Proc.10th Usenix Security Symp. (USENIX 2001), pp. 177–189. Usenix Assoc. (2001)
9. Chen, H., Wagner, D.: MOPS:An Infrastructure for Examining Security Properties of Software. In: Proc. 9th ACM Conf. Computer and Communications Security (CCS 2002), pp. 235–244. ACM Press (2002)
10. Zheng, H., Zhou, K.: Research on XML based static software security analysis. In: Proc. 2nd WRI World Congress on Software Engineering (WCSE 2010), pp. 141–144 (2010)

11. Antoniol, G.: XML-Oriented gcc AST Analysis and Transformations. In: Proceedings of the Third IEEE International Workshop on Source Code Analysis and Manipulation, pp. 869–901 (2005)
12. Kraft, N.A., Duffy, E.B., Malloy, B.A.: Grammar Recovery from Parse Trees and Metrics-Guided Grammar Refactoring. IEEE Transactions on Software Engineering 35(6), 780–794 (2009)
13. Kraft, N.A., Malloy, B.A., Power, J.F.: An Infrastructure to Support Interoperability in Reverse Engineering. Information and Software Technology 49(3), 292–307 (2007)

An Extended ISOMAP by Enhancing Similarity for Clustering

Hong Yu, Xianchao Zhang, Yuansheng Yang,
Xiaowei Zhao, and Lei Cai

School of Electronics and Information Engineering,
Dalian University of Technology, Dalian,China
Software School, Dalian University of Technology, Dalian, China
{hongyu,vivan_dlut,xczhang,yangys,cailei}@dlut.edu.cn
http://www.springer.com/lncs

Abstract. Isomap is an important dimension reduction method for clustering data with relatively large features. Isomap uses geodesic distance instead of Euclidean distance to reflect geometry of the underlying manifold, while it ignores the classification principle that the distance between samples on different manifolds should be large and the distance between samples on the same manifold should be small. In this paper, we employed a path based distance to extend Isomap for clustering. The path based distance measure strengthens the similarity of the points on the same manifold. The useful behavior of the similarity strengthening Isomap is confirmed through numerical experiments with several data sets.

Keywords: Isomap, Dimension Reduction, Clustering.

1 Introduction

Clustering is the process of grouping or classifying collection samples, into classes called clusters so that the samples within a cluster are "similar" to one another, yet"dissimilar" to samples in other clusters. Many varieties of clustering methods have been proposed over the past several decades [1][2], such as K-Means, Fuzzy c-Means, SOM and so on. Earlier or "classical" clustering methods typically dealt with smaller sets of features (lower dimensional data) and considered all the features of the data simultaneously relevant to each of the underlying clusters of the data. With the advanced technology in data collection, many clustering applications are now characterized by relative large features. But not all features of the data are relevant to the clustering result, and the presence irrelevant features have the potential to eliminate clustering tendency and mislead the clustering algorithms. There are many approaches to address this problem. The simplest approach is dimension reduction[3][4][6] techniques, including principal component analysis (PCA),multidimensional scaling (MDS) and manifold learning [3][9][10]. Isomap(or isometric feature mapping) is an isometric manifold learning method, which extends multidimensional scaling (MDS) by considering

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 808–815, 2012.

Dijkstra's geodesic distances (shortest paths) on a weighted graph instead of Euclidean distances[3]. Isomap methods mainly focus on preserving the global manifold structure and ignore the classification principle (i.e., maximizing the distances between samples on different manifolds or minimizing the distances between samples on the same manifold).

In this paper,we give the *ε-neighbors* path definition to reflect the manifold smooth consistence perfectly and employ it to measure the distance between samples. The *ε-neighbors* based distance enhances the similarity between samples on the same manifold. With the *ε-neighbors* path based distance measure, we extend Isomap to clustering. Encouraging experimental results on artificial and real data demonstrate the effectiveness of the proposed approach.

The rest of this paper is organized as follows. We begin with a brief review of Isomap in section 2.In section 3,we present our definition of the *ε-neighbors* path based distance and the extended Isomap for clustering. Experimental results are then presented in section 4.Finally we give the conclusion and some discussions.

2 Review of Isomap

Isomap[3][7] combines the major algorithmic features of PCA and MDS with the flexibility to learn a broad class of nonlinear manifolds. The geodesic distance employed by Isomap reflects the intrinsic geometry of the underlying manifold[3].As shown in Fig. 1, Isomap can unfold the dataset by preserving the intrinsic structure completely. This unfolding problem is difficult for MDS.

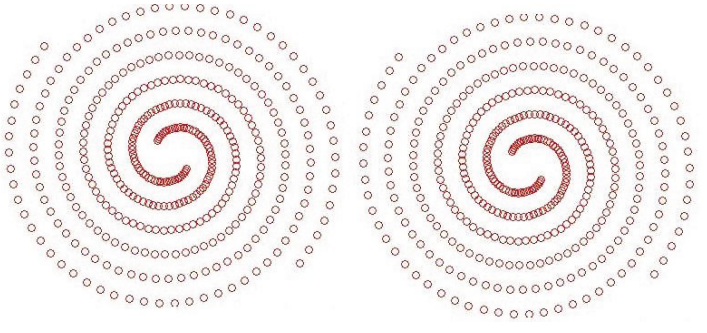


Fig. 1. An artificial data set "Two-spirals"(left) and the unfolding result with Isomap(right)

The detail[3] of Isomap is as follows.

Input: N data points $\{x_1,x_2,...,x_N\}$in the input space X.

Output:coordinate vectors $\{y_1,y_2,...,y_N\}$in a d-dimensional Euclidean space Y.

Step1: Identify *ε-neighborhood*or k nearest neighbors of each input data point and construct a neighborhood graph $D_X(i,j)$where edge lengths between points in a neighborhood are set as their Euclidean distances otherwise set as ∞.

Step2: Compute geodesic distances $D_G(i,j)$ that are associated with the sum of edge weights along shortest paths between all pairs of points.

Step3: Construct d-dimensional embedding. Let $\lambda_p$be the p-*th* eigenvalue (in decreasing order) of the matrix -$HD^2H/2$ where H=I-$e_N{e_N}^T/N$, e_N=$[1,1,...,1]^T$ $\in R^N$, and ${\nu_p}^i$ be the i-*th* component of the p-*th* eigenvector. Then set the p-*th* component of the d-dimensional coordinate vector y_i equal to $\sqrt{\lambda_p \nu_p^i}$.

Although Isomap unfolds the dataset well, it can't reflect that the distance between samples from the same class is small and the distance between samples from different classes is large (as point a and point b shown in Fig. 2). When we use the reduction result and traditional clustering methods such as k-means to cluster, we get poor result. In this paper, we improve the dimension reduction quality for clustering by decreasing the distance between samples from same the class and increasing the distance between samples from different classes.

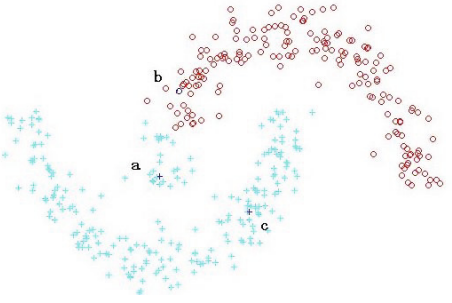


Fig. 2. Dimension Reduction result by Isomap

3 Extending ISOMAP by Enhancing Similarity for Clustering

3.1 Distance Based on the *ε-neighbors* Path

The geodesic distance used in Isomap reflects the intrinsic geometry of the underlying manifold but neglects that in the reduction subspace the distance between samples from the same class should be small and the distance between samples from different class should be large. For this drawback, Isomap fails to serve the clustering. The clustering result on a toy dataset "Two-moon" with the attributes obtained by Isomap is shown in Fig. 2.

Although the geodesic distance $d_{a,b}$=$d_{a,c}$, the similarity $s_{a,b}$ should be larger than $s_{a,c}$ because point a is on the same manifold with point b. Unfortunately, Isomap only holds that points on the same manifold in the original attributes space are still on the same manifold in the embedding space. In order to improve the clustering result, the dimension reduction method needs to strengthen the dissimilarity of the points on different manifolds and similarity of the points on the same manifold. Following the analysis, we employ the ε-*neighbors* path of point a and b to define the distance.

Definition 1. *ε-neighbors* of a point represents the nearest neighbors whose distance from the point is less than ε by Euclidean distance,where ε is an empirical parameter.

Definition 2. *ε-neighbors* path of point and is the following simple path: $P_{(a,b)} = \{a{=}\nu_0 \rightarrow \nu_1 \rightarrow \nu_2 \rightarrow ... \rightarrow \nu_i \rightarrow \nu_j \rightarrow ... \rightarrow b | \nu_j \in \varepsilon\text{-}neighbors(\nu_i)\}$

Based on the *ε-neighbors* path definition, there may be no path between two points which lie on different manifolds. As shown in Fig. 3, there is a *ε-neighbors* path between a and b while there is no such path between point a and c. It coincides that point a and b lie on the same smooth manifold and point a and c lie on different manifolds.

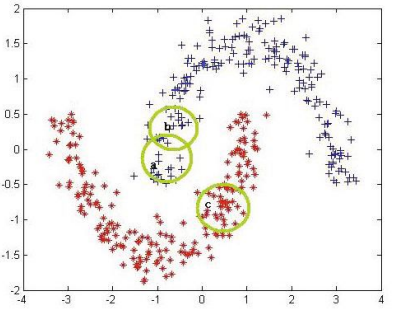


Fig. 3. An artificial data set " Two-moon"

With the *ε-neighbors* path of point a and b, we define the distance between point a and b as:

$$d_{(a,b)} = \begin{cases} d_{(a,b)} & if\ there\ is\ no\ \varepsilon - neighbors\ path\ between\ a\ and\ b \\ min_{p \in P_{a,b}}\, max_{0 \le h < |P|}\, d_{(p_h, p_{h+1})} & else \end{cases} \quad (1)$$

With our distance measure, the distance between point a and point b is smaller than that between point a and point c. Different with Euclidean distance and geodesic distance, our distance measure strengthens the similarity of the points on the same manifold(class).

3.2 Extending ISOMAP by Enhancing Similarity for Clustering

With the *ε-neighbors* path based distance, we propose the similarity enhancing Isomap for clustering as follows.

Input: Data set X,number of clusters k,the neighborhood parameterε.

Output: Clusters $\{C_1, C_2, ..., C_k\}$.

Step1:Identify *ε-neighborhood*or k nearest neighbors of each input data point and construct a neighborhood graph $D_X(i,j)$where edge lengths between points in a neighborhood are set as their Euclidean distances otherwise set as ∞.

Step2: Compute the *ε-neighbors* path based distances $D_G(i,j)$ that are defined as (1).

Step3: Same to Isomap,get the coordinate vectors $\{y_1,y_2,...,y_N\}$in a d-dimensional Euclidean space Y.

Step4: Cluster the points $\{y_1,y_2,...,y_N\}$ with traditional clustering algorithms such as k-means into clusters$\{ P_1, P_2, ..., P_k\}$.

Step5: Assign the original point x_i into cluster C_j if and only if y_i is assigned into cluster P_j .

This extended Isomap enhances the dissimilarity of the points on different manifolds and similarity of the points on the same manifold fully. We call this method as $An extended ISOMAP by Enhancing the Similarity for Clustering$ $(ESC-ISOMAP)$. It gives significant improvement in clustering results. As shown in Fig. 4, we can get a better embedding for "two-spirals" clustering with the ε-*neighbors* path based distance.

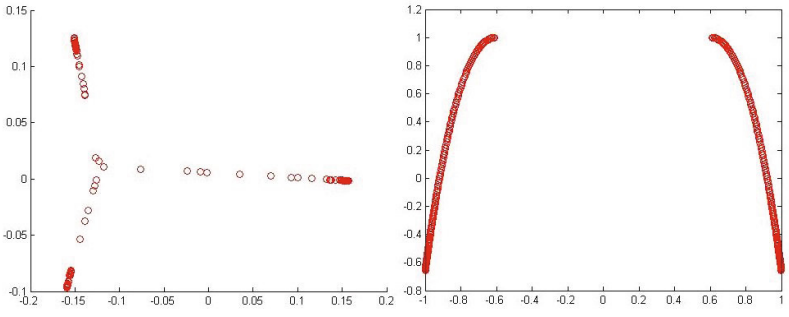


Fig. 4. Two-dimensional embedding for the data set "two-spirals" using Isomap with geodesic distance(left) and ε-*neighbors* path based distance(right)

4 Experiments

To test the proposed algorithm, we have applied it to several artificial and real data sets.

4.1 Evaluation Measures

We use normalized mutual information (NMI)[5] to evaluate a clustering result, which is an information theoretical criterion for the evaluation of clustering algorithms. Given two clustering results, $\Delta = \{C_1, C_2, \ldots, C_c\}$ and $\Delta' = \{C'_1, C'_2, \ldots, C'_k\}$ of $X(|X| = n)$, let n_i and n'_i be the number of objects in cluster C_i and C'_i separately. Let n_{st} denote the number of objects that are in cluster C_s as well as in cluster C'_t, then the normalized mutual information of Δ and Δ' is

$$\mathrm{NMI} = \frac{\sum_{s=1}^{c}\sum_{t=1}^{k}\log\left(\frac{nn_{st}}{n_s n'_t}\right)}{\sqrt{\left(\sum_{s=1}^{c} n_s \log\frac{n_s}{n}\right)\left(\sum_{t=1}^{k} n'_t \log\frac{n'_t}{n}\right)}}. \tag{2}$$

Given the true labels δ of X and a clustering result ξ, we have $0 \leq NMI(\delta, \xi) \leq 1$. When δ equals ξ, $NMI(\delta, \xi) = 1$.The larger the NMI is, the better the clustering performance is.

We compare the clustering performance of the extended Isomap(noted as ESC-Isomap) and Isomap by NMI.

4.2 Clustering Results

We first demonstrate the effectiveness of our algorithm on four artificial data sets shown in Fig. 5.which have been mentioned in the literature and may be not convex data sets. Fig. 5 displays the original datasets. We first use Isomap with geodesic distance and *ε-neighbors* path based distance separately on the data sets to detect the new representation. In particular, the simple k-means clustering algorithm has no difficulties to detect the clusters in this new representation. Readers not familiar with k-means can employ other clustering algorithms.We found that the clustering results are insensitive to the selected clustering algorithms.The neighborhood parameter is set as *(the max distance + the min distance)//(cluster number)* intuitively. The comparing results with NMI are shown in Table 1. The *ε-neighbors* path based distance measure strengthens the local and global similarity between the points from the same manifold. It is the main reason for our algorithm's perfect performance.

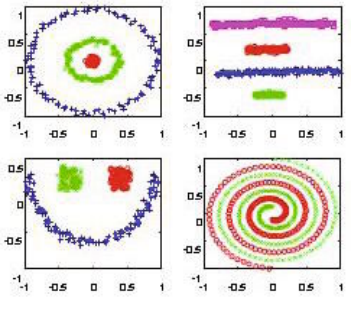


Fig. 5. Four Artificial Data Sets

Table 1. Clustering Results on artificial data sets evaluated by NMI

Data Set	Clustering with Isomap	Clustering with ESC-Isomap
threecircles	0.1601	1
fourlines	0.597	1
smile	0.529	1
twospirals	0.6095	1

We also demonstrate the effectiveness of our algorithm on two real data sets from various domains as shown in Table 2. These data sets have known labels and are mainly used for classification tasks.

Table 2. Results on real data sets evaluated by NMI

Data Set	Iris	Sonar
Number of Instances	150	208
Number of Attributes	4	60
Number of Clusters	3	2

The clustering results are shown in Table 3. With the evaluation measure, the similarity strengthening Isomap outperforms Isomap. But the clustering results are still not satisfying. Two reasons should be responsible. First there are some noise data[8] on the real data sets which interfering the clustering. Second the data sets are highly nonlinear and the clustering algorithms tend to yield poor results.

Table 3. Real data sets description

Data Set	Clustering with Isomap	Clustering with ESC-Isomap
Iris	0.0105	0.0258
Sonar	0.7419	0.7812

5 Conclusions

In this paper, we have presented a definition of *ε-neighbors* path to encode the data points' similarity on the same manifold and proposed the similarity enhanced Isomap for clustering. The *ε-neighbors* path based Isomap enhances the similarity between points on the same manifold(class),while geodesic distance usually neglects the similarity and just holds the intrinsic structure. After the dimension reduction, many different clustering algorithms can be used not only k-means. Experimental results on artificial and real data sets demonstrate the efficient performance. Our future work will focus on efficient methods to eliminate interfering by the noise data and improve the clustering result further.

References

1. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice Hall, Englewood Cliffs (1988)
2. Wu, M., Scholkopf, B.: A Local Learning Approach for Clustering. In: NIPS, pp. 1529–1536 (2007)
3. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science 290, 2319–2323 (2000)
4. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley-Interscience, New York (2001)
5. Strehl, A., GhoshZ, J.: Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. Journal of Machine Learning Research 3(3), 583–617 (2003)

6. Roweis, S.T., Saul, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. Science 290(5500), 2323–2326 (2000)
7. Pradhan, C., Mishra, S.: Optimized ISOMAP Algorithm Using Similarity Matrix. In: 3rd International Conference on Electronics Computer Technology (ICECT), pp. 212–215 (2011)
8. Li, B., Huang, D.-S., Wang, C.: Improving The Robustness of ISOMAP by Denoising. In: IEEE World Congress on Computational Intelligence, Hong Kong, pp. 266–270 (2008)
9. Nie, F., Xu, D., Tsang, I.W.-H., Zhang, C.: Flexible Manifold Embedding: A Framework for Semi-Supervised and Unsupervised Dimension Reduction. IEEE Transactions on Image Processing 19(7), 1921–1932 (2010)
10. Shi, L., Yang, Q., Liu, E., Li, J., Dong, Y.: An Incremental Manifold Learning Algorithm Based on the Small World Model. In: Li, K., Fei, M., Jia, L., Irwin, G.W. (eds.) LSMS 2010 and ICSEE 2010, Part I. LNCS, vol. 6328, pp. 324–332. Springer, Heidelberg (2010)

Movie Keyframe Retrieval Based on Cross-Media Correlation Detection and Context Model

Yukang Jin[1], Tong Lu[1,2,⋆], and Feng Su[1]

[1] State Key Laboratory for Novel Software Technology, Nanjing University
Nanjing, 210093, China
[2] Jiangyin Institute of Information Technology of Nanjing University, China
jyk0822@hotmail.com, lutong@nju.edu.cn

Abstract. In this paper, we propose a novel cross-media correlation detection method for movie keyframe retrieval. We first compute the temporal saliency on both the video and audio streams in a movie separately, then locate the resonance regions that the saliency changes in these two modalities show strong correlations. Next, starting from resonance regions, we propagate the similarity of visual and auditory characteristics through neighboring movie regions based on a temporal movie context model, segmenting the movie into a sequence of coherent parts from which keyframes are extracted. The experimental results on actual movie clips show that, compared to the single-modality algorithms, our method gives improved retrieval performance in completeness and precision due to the efficient exploitation of the context and correlations between complementary multi-modalities.

Keywords: cross-media correlation detection, context model, similarity propagation, keyframe retrieval.

1 Introduction

With the wide use of digital multimedia technologies, massive multimedia materials have been produced, which raises strong demands for efficient analyzing and mining techniques [1][2][3][4]. In recently years, besides the success of many techniques working on single media type, exploitation of cross-media correlation information in multimedia content analysis and processing have attracted increasing interests [5][6][7].

Studies on cross-media processing can be sorted into two categories: cross-media index techniques and multi-modal correlation analysis techniques. Cross-media index techniques focus on automatically labeling un-annotated multimedia data [8][9]. Such approaches first convert clustered visual or auditory features to indices into certain feature dictionary, then analyze and generate linked image-text (or audio-text) translation representation and construct the

⋆ Corresponding author.

H. Jiang et al. (Eds.): IEA/AIE 2012, LNAI 7345, pp. 816–825, 2012.

results based on it. Multi-modal correlation analysis approaches [6][7][10] focus on extracting statistical correlations between different modalities from their co-occurrence relationships. However, such methods also have limitations. For example, people just focus on the detection and exploitation of the correlations between multi-modalities resulting from the same source.

Specifically, in this paper, we propose a novel cross-media correlation detection and similarity propagation algorithm based on temporal context model. As shown in Fig. 1, our method consists of two stages: firstly, we compute the temporal visual and auditory saliency separately and detect the correlations between two modalities to locate the resonance regions. Then, we exploit context model to propagate the similarity of neighboring movie regions to obtain more precise boundaries of the regions for keyframe extraction. The experiments of keyframes extraction on movie clips show the efficiency of our method compared to the single-modality algorithm. The main contributions of our paper are: first, our algorithm, which is based on human saliency, is more reasonable; second, we take the movie contextual relationship into account.

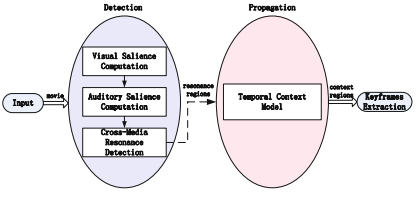


Fig. 1. Block diagram of the proposed algorithm

The paper is organized as follows: In Section 2, we introduce the cross-media correlation detection method, which consists of three parts: visual saliency computation, auditory saliency computation and correlation detection on both. In Section 3, we describe the context model and the similarity propagation algorithm. The experiment results are given in section 4.

2 Cross-Media Correlation Detection

In this section, we describe the measurements to compute the saliency index for the visual and auditory stream, which are further ordered temporally to form the visual and auditory saliency curves. Based on them, we analyze the typical relationships between two curves and identify the so-called resonance regions, which are exploited later for cross-media correlation propagation.

2.1 Visual Saliency Computation

[11] proposes a visual attention index (VAI) method for visual saliency modeling. But it is for surveillance videos consisting of stationary scenes and occasional

saliency motions, which can be identified by local abnormal oscillations of attention indices. A movie, however, usually contains fast changing scenes, either in visual or auditory or both channels, for which the attention indices always fluctuate.

In our algorithm, we denote the the whole movie attention index (MAI) as

$$\mathrm{MAI} = \{\mathrm{MAI}_1, \mathrm{MAI}_2, \ldots, \mathrm{MAI}_W\} \tag{1}$$

where MAI_a is the movie attention index (MAI) of movie scene a. MAI_a computation is divided into two parts: dynamic attention index and static attention index. The final movie attention index (MAI) of each scene is computed as

$$\mathrm{MAI}_a = \frac{1}{L}(w_{T,a} * A_{T,a} + w_{S,a} * A_{S,a}) \tag{2}$$

where $A_{T,a}$ is the dynamic attention index and $A_{S,a}$ is the static attention index, $w_{T,a}$ and $w_{S,a}$ are the relative weights of each component, L is the total number of frames in the scene a.

The dynamic attention is based on the the fact that human visual attentions are usually attracted by motion contrast. We divide every frame f into N blocks of size 8*8 and denote the motion vector of block i as (dx_i, dy_i). The motion intensity of each block is $\gamma_i = \sqrt{dx_i^2 + dy_i^2}$. As a result, we build an orientation histogram $v(j)$ for all the blocks and the motion attention of block i is computed as

$$A_{T,a} = \sum_{f}^{L} \sum_{i}^{N} (\gamma_i * (1 - \frac{v(b(i))}{\sum_{j=1}^{H} v(j)})) \tag{3}$$

where H is the maximum bin index of the histogram, $b(i)$ is the bin index of block i.

The static attention is defined to reflect the phenomenon that attention is also attracted by the objects presented in the still images. We compute the color contrast index $C_i = \sum_q d(p_i, q)$ of each block p_i by comparing its color histogram with its neighboring blocks q and summing up the distances. Then, the static attention index is defined as

$$A_{S,a} = \frac{1}{N} \sum_{f}^{L} \sum_{i}^{N} (w_i \times C_i) \tag{4}$$

where w_i is a Gaussian falloff weight relative to the frame f center.

To fuse the dynamic and static attention index together, we define the weights in (2) for each attention as $w_{T,a} = A_{Tave,a} \times \exp(1 - A_{Tave,a})$, $w_{S,a} = 1 - w_{T,a}$ and $A_{Tave,a} = Max(A_{T,a}) - Mean(A_{T,a})$. Finally, we get the movie visual saliency curve in Fig. 2.

2.2 Auditory Saliency Computation

We employ [12] to identify the general auditory background in the movie clips, thus consider those audio scenes with great deviation from the background are

salient to audience. The algorithm works as follows: The audio stream is first segmented into M uniformly-sized non-overlapping audioframes $\{f_i\}_{i=1,2,\ldots,M}$. Each audioframe f_i consists of $N = 2205$ amplitude samples of signals, obtained by sampling an audioframe of duration 0.2 second with a sampling rate of 11.025 kHz. Therefore, we can organize the audioframes into a $N \times M$ matrix $S = [f_1, f_2, \ldots, f_M]$, compute the mean audioframe f_m and subtract it from S.

Next, we identify the subspace of audio background X_b by choosing a set of K eigenvectors corresponding to K small eigenvalues, and use them to form the $N \times K$ eigen-audiospace projection matrix E_b.

Then, to learn the audio background characteristics, we extract a set of audio background frames that no foreground signals of interest are present, average them and subtract f_m from it to form the zero-mean background frame f_b. We finally project f_r into the eigen-audiospace X_b to obtain $\bar{f}_b$,

$$\bar{f}_b = E_b f_b \tag{5}$$

which is used as the reference background frame.

Based on $\bar{f}_b$, for one projected audioframe $\bar{f}_i$ $(i = 1, 2, \ldots, M)$ of the input audio stream, we compute the distance $D(i)$ and the reference background frame:

$$D(i) = ||\bar{f}_i - \bar{f}_b|| \tag{6}$$

which measures its deviation from the background.

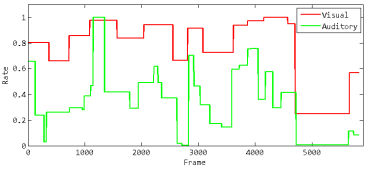


Fig. 2. Example of visual saliency curve (red) and auditory saliency curve (green)

Finally, to group audioframes into scenes, we compute the visual similarity [10] on neighboring frames and split the audio stream at the local minima. For each audio segment, we average $D(i)$ and use it as the auditory saliency index in Fig. 2.

2.3 Relationships of Curves

To achieve an initial rough segmentation of the input movie into coherent regions based on the visual and auditory saliency curves, we summarize the relationships between two curves into four typical forms, as shown in Fig. 3. In each form, there are three temporally ordered scene regions. Specifically, in form (a) and (c), the middle region corresponds to the physical phenomenon called "resonance", in which a dynamical system with periodic oscillations at some frequencies e_i shows a remarkable response when it's subjected to a periodic forcing of frequencies

near one of the e_i [13]. In some sense, a movie is just like a dynamical system, which also contains temporal oscillations in visual and/or auditory channels. Because of the interactions of video and audio, movies can attract the audience and resonate with them. Therefore, the middle regions in (a) and (c) are considered strongly correlated and contain most keyframes. In (b) and (d), the curves do not harmonize with each other in the time axis and are considered less correlated with each other. Although the curves in Fig. 2 are not strictly in accordance with the four forms in Fig. 3, we can still find regions in Fig. 2 that are like the form of the middle region in (a) and (c) of Fig. 3.

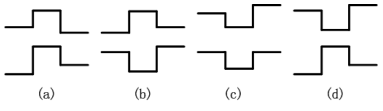


Fig. 3. Illustration of the typical temporal relationships and the resonance region of the visual and auditory salience curves

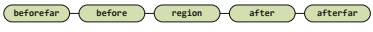


Fig. 4. The temporal context model of movies

The movie segments with the salience curves of the form (a) are typically attractive, sharp, fierce in both visual and auditory channel, while segments of the form (c) are usually motionless, static and harmonious. The audio clips and the image frames in these regions are strongly correlated because of resonance. In this paper, such resonance middle regions are identified as

$$M_i = [t^b_{V,i}, t^e_{V,i}] \cap [t^b_{A,i}, t^e_{A,i}]$$

where $t^b_{V,i}$ and $t^e_{V,i}$ are the beginning and end time, respectively, of the i-th region of the form (a) or (c) found in the visual salience curve, while $t^b_{A,i}$ and $t^e_{A,i}$ are those found in the auditory salience curve. $[t^b_{V,i}, t^e_{V,i}]$ and $[t^b_{A,i}, t^e_{A,i}]$ overlap in the time axis but may not align with each other precisely.

3 Movie Context Model and Similarity Propagation

For more precise refinement of the M_i region boundaries, in this section, we propose a temporal context model for movies and develop a cross-media correlation algorithm with similarity propagation.

3.1 Temporal Context Model

As shown in Fig. 2, the emergence of the auditory event and the visual event may not align strictly on the time axis like the form (a) and (c) in Fig. 3. More important, the regions of the form (b) and (d) in Fig. 3 may also contain keyframes.

Therefore, we need efficient measurements to extend the resonance regions to incorporate coherent frames with similar visual and auditory characteristics.

[14] introduces a novel approach for object categorization that incorporates two types of context co-occurrence of relative location and local appearance features. Every object or event in one scene has its spatial or the temporal context relationships with the environment. For movies, this temporal context relevance of scenes is particularly evident. For example, before an explosion event in a war movie, there are usually soldiers, aircrafts, guns, bombs, fires, the voice of soldiers and aircrafts and so on. Also after the explosion, there are usually wounded soldiers, crying sounds, aircrafts and fires. In our method, these cues are used to propagate the similarity through both the visual channel and auditory channel and help to distinguish different categories of scenes.

We develop a temporal context model for movies to capture relative time location constraints. As shown in Fig. 4, we define the context relationships — beforefar, before, after and afterfar, each of which stands for a neighboring relationship of the middle region M relative to other parts in the time sequence. As described before, the movie is first roughly decomposed into several resonance regions based on the interaction of the visual and auditory curves. Denoting the middle region by M, the movie parts before M by B and the after parts by A, we define the temporal context relationships of the middle region with the neighboring movie parts as:

$$S(B, M, A) = \sum_{i,j} V_{ij} d_{ij} + \sum_{i,j} A_{ij} d_{ij} \tag{7}$$

where V_{ij} and A_{ij} are the similarity weight (for frame j of region i) computed as Euclidean distance of the auditory/visual features of frames in A and B relative to those in M. d_{ij} is the temporal distance of the frame relative to the middle region.

3.2 Similarity Propagation Algorithm

Based on the previous analysis, we give our temporal contextual similarity propagation algorithm to adaptively segment the movie into the sequence of closely correlated parts, from which we extract the keyframes.

The main idea of this algorithm in Table 1 is: we compute S(B,M,A) to measure the correlation between the middle region and the neighboring parts. Low correlation means that the middle region has low correlation with the neighboring parts and does not need to use temporal context model for similarity propagation. If M has high correlation with neighboring parts, according to the type of movie part, we choose corresponding procedure to update M. Finally, the keyframes are extracted from M with a fixed uniform interval (every 35 frames in our experiment).

In step 1, $S(B, M, A)$ is computed to measure the correlation between the resonance region and its neighboring parts. According to the correlation, we decide to relocate the boundaries of the resonance regions or not in step 2.

Table 1. Temporal contextual similarity propagation algorithm

Algorithm The details of similarity propagation algorithm
Initialization. B = $\{\theta \mid \theta$ is a region in front of M$\}$, A = $\{\psi \mid \psi$ is a region after M$\}$;
Setp1. $S' = S(B, M, A)$;
Step2. If $S' < \nu$ (ν is a empirical threshold) Goto **Output.**
Step3. Repeat (movie part type is before):
a. Add one before movie part α;
b. Compute $\Omega = S(\alpha, B \cup M, A)$ - S';
c. If $\Omega < 0$ Goto **Step4.**
d. $S' = S(\alpha, B \cup M, A)$;
e. $M = \alpha \cup M$;
f. Goto **Step3.**
Step4. Repeat (movie part type is after):
a. Add one after movie part β;
b. Compute $\Omega = S(B, M \cup A, \beta)$ - S';
c. If $\Omega < 0$ Goto **Output.**
d. $S' = S(B, M \cup A, \beta)$;
e. $M = M \cup \beta$;
f. Goto **Step4.**;
Output. The result M.

Decided by the context type of a movie part, step 3 or step 4 is chosen to update M. Note that in the updating module, we first compute the difference Ω between S and S'. S' and M are updated if $\Omega > 0$, while the updating will be ended if $\Omega < 0$.

4 Experiments

4.1 Dataset

Our movie dateset contains Pearl Harbor (Touchstone Pictures, 2001), some fragments of The Big Bang Theory (CBS, 2007) and Titanic (Lightstorm Entertainment, 1997). To evaluate the performance, we choose a fragment of 4 minutes movie from Pearl Harbor. In the visual aspect, the test movie fragment contains fierce sharp scenes (explosion, flying planes and so on), as well as harmonious and static scenes (office, park and so on). In the auditory aspect, it includes acute explosion sounds, as well as people talks and silences. We extract 128 dimensional SIFT features from each movie frame and 21 dimensional MFCC features from each audio signal segment. Some example frames of the test movie fragment are shown in Fig. 5.

Fig. 5. Example video frames in the test movie fragment

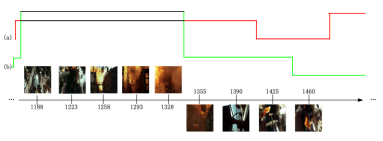


Fig. 6. Keyframe retrieval results from the explosion scenes (below) and the corresponding visual (a) and auditory (b) saliency curves (above)

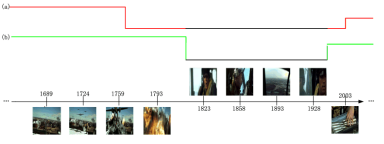


Fig. 7. Keyframe retrieval results from the plane scenes (below) and the corresponding visual (a) and auditory (b) saliency curves (above)

4.2 Results and Comparison

According to the resonance regions and temporal context model, the keyframe results of this movie clip are divided into four parts: Fig. 6, Fig. 7, Fig. 8 and Fig. 9. In Fig. 6, we show the keyframe retrieval results on the battle scene in test movie fragment. For the shown saliency curves, the detected middle resonance regions is colored with black. Keyframes shown come from two sources: above the time axis shows the keyframe from the resonance region, while below are keyframes from the neighboring movie parts identified by the temporal context model.

Fig. 7 and Fig. 8 show the keyframe retrieval results from two different types. From the results, we can see that, for different scene types, the temporal constraints in the proposed movie context model have different contributions to the retrieval. In Fig. 6, the after relationship makes more contribution than the before relationship, while in Fig. 7 and Fig. 8, the before relationship yields more keyframes than the after one.

In Fig. 9, the keyframes extracted are mainly from the middle resonance regions, while the temporal context model contributes little. This is because, for that scene, the middle resonance region has little similarity with its neighboring parts.

For comparisons, we also implement the algorithm proposed in [11] and show its keyframe retrieval results on the test movie fragment in Fig. 10. The scale shows that [11] pays more attention to more concentrative fierce frames (11 of all 12 keyframes). It also can be seen from Fig. 10 that the method in [11] pays more attentions to those fierce, drastically changing video frames like the battle scenes, as 11 of all 12 keyframes it extracts are of such type. On the other hand,

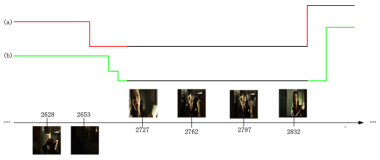


Fig. 8. Keyframe retrieval results from the indoor scenes (below) and the corresponding visual (a) and auditory (b) saliency curves (above)

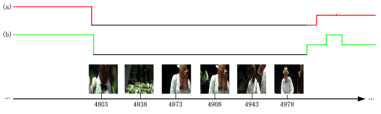


Fig. 9. Keyframe retrieval results from the outdoor scenes (below) and the corresponding visual (a) and auditory (b) saliency curves (above)

it ignores the intermediate talking scene, which also attracts audiences' attention and have been successfully located by our method.

To further verify the performance, we invite 10 persons to see the movie and choose the frames which they are most interested in and get the rate in Table 2 by comparing the frame content with our algorithm and [11]. The text movies are divided into three types: Pearl Harbor is a film of war epic, while The Big Bang Theory is a TV drama and Titanic is an emotion film. War epic type films usually contain fiercely changing scenes, while emotion type films contain fewer changing scene than epic war type ones. In contrast, TV dramas has the fewest changing scenes.

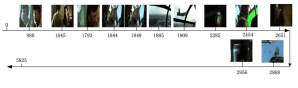


Fig. 10. Keyframe retrieval results by [11]

Table 2. Comparision

Movie	[11](%)	Our algorithm(%)
Pearl Harbor	57.14	76.19
Titanic	61.90	71.43
The Big Bang Theory	63.64	62.50
Average	60.89	70.04

From Table 2, we can see that Pearl Harbor with complex structure, our algorithm make obvious improvement compared to [11]. The result of Titanic also make better performance, while The Big Bang Theory with simple structure shares similar performance. The results also show that for movies that consist of richer contents and representations and more complicated structure, our method yields better results.

5 Conclusions

This paper proposes a novel cross-media correlation detection and similarity propagation method based on the context model, which is further applied in movie keyframe retrieval. Experimental results show that, compared to methods operating on single modality, our method significantly increases the completeness of the retrieval result as well as its closeness to the actual retrieval intention by interactions of video and audio of a movie. Further work includes improving of the efficiency of correlation learning and extending the context model for

describing richer structures of a movie. The performance of the proposed algorithm on other benchmark datasets will also be explored.

Acknowledgement. The work described in this paper was supported by the the New Century Excellent Talent of Ministry of Education Project under NCET-11-0232, National Science Foundation of China under Grant No. 61021062, and the 973 Program of China under Grant No. 2010CB327903.

References

1. Yu, B., Ma, W.Y., Nahrstedt, K., Zhang, H.J.: Video summarization based on user log enhanced link analysis. In: ACM Multimedia Conference, pp. 382–391 (2003)
2. Feng, S., Manmatha, R., Lavrenko, V.: Multiple bernoulli relevance models for image and video annotation. In: Computer Vision and Pattern Recognition, pp. 1002–1009 (2004)
3. Datta, R., Li, J., Wang, J.Z.: Content-based image retrieval: approaches and trends of the new age. In: Multimedia Information Retrieval, pp. 253–262 (2005)
4. Chang, E.Y., Goh, K., Sychay, G., Wu, G.: Cbsa: content-based soft annotation for multimodal image retrieval using bayes point machines. IEEE Transactions on Circuits and Systems for Video Technology 13, 26–38 (2003)
5. Beal, M.J., Attias, H., Jojic, N.: Audio-Video Sensor Fusion with Probabilistic Graphical Models. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 736–750. Springer, Heidelberg (2002)
6. Wang, J., Zeng, H.J., Chen, Z., Lu, H., Tao, L., Ma, W.Y.: Recom: reinforcement clustering of multi-type interrelated data objects. In: Research and Development in Information Retrieval, pp. 274–281 (2003)
7. Wang, X.J., Ma, W.Y., Xue, G.R., Li, X.: Multi-model similarity propagation and its application for web image retrieval. In: ACM Multimedia Conference, pp. 944–951 (2004)
8. Blei, D.M., Jordan, M.I.: Modeling annotated data. In: Research and Development in Information Retrieval, pp. 127–134 (2003)
9. Barnard, K., Duygulu, P., Forsyth, D.A., de Freitas, N., Blei, D.M., Jordan, M.I.: Matching words and pictures. Journal of Machine Learning Research 3, 1107–1135 (2003)
10. Zhang, H., Zhuang, Y., Wu, F.: Cross-modal correlation learning for clustering on image-audio dataset. In: ACM Multimedia Conference, pp. 273–276 (2007)
11. Peng, J., Xiaolin, Q.: Keyframe-based video summary using visual attention clues. IEEE Multimedia 17, 64–73 (2010)
12. Kyperountas, M., Kotropoulos, C., Pitas, I.: Enhanced eigen-audioframes for audiovisual scene change detection. IEEE Transactions on Multimedia 9, 785–797 (2007)
13. Benzit, R., Sutera, A., Vulpiani, A.: The mechanism of stochastic resonance (1981)
14. Galleguillos, C., Rabinovich, A., Belongie, S.: Object categorization using co-occurrence, location and appearance. In: CVPR (2008)

Author Index