Theory and Applications of Natural Language Processing
Edited volumes

Alessandro Oltramari
Piek Vossen · Lu Qin
Eduard Hovy *Editors*

New Trends of Research in Ontologies and Lexical Resources

Ideas, Projects, Systems

Springer

Theory and Applications
of Natural Language Processing

Aims and Scope

The field of Natural Language Processing (NLP) has expanded explosively over the past decade: growing bodies of available data, novel fields of applications, emerging areas and new connections to neighboring fields have all led to increasing output and to diversification of research.

"Theory and Applications of Natural Language Processing" is a series of volumes dedicated to selected topics in NLP and Language Technology. It focuses on the most recent advances in all areas of the computational modeling and processing of speech and text across languages and domains. Due to the rapid pace of development, the diversity of approaches and application scenarios are scattered in an ever-growing mass of conference proceedings, making entry into the field difficult for both students and potential users. Volumes in the series facilitate this first step and can be used as a teaching aid, advanced-level information resource or a point of reference.

The series encourages the submission of research monographs, contributed volumes and surveys, lecture notes and textbooks covering research frontiers on all relevant topics, offering a platform for the rapid publication of cutting-edge research as well as for comprehensive monographs that cover the full range of research on specific problem areas.

The topics include applications of NLP techniques to gain insights into the use and functioning of language, as well as the use of language technology in applications that enable communication, knowledge management and discovery such as natural language generation, information retrieval, question-answering, machine translation, localization and related fields.

The books are available in printed and electronic (e-book) form:

* Downloadable on your PC, e-reader or iPad
* Enhanced by Electronic Supplementary Material, such as algorithms, demonstrations, software, images and videos
* Available online within an extensive network of academic and corporate R&D libraries worldwide
* Never out of print thanks to innovative print-on-demand services
* Competitively priced print editions for eBook customers thanks to MyCopy service http://www.springer.com/librarians/e-content/mycopy

For other titles published in this series, go to
www.springer.com/series/8899

Alessandro Oltramari • Piek Vossen
Lu Qin • Eduard Hovy
Editors

New Trends of Research in Ontologies and Lexical Resources

Ideas, Projects, Systems

Springer

Editors
Alessandro Oltramari
Psychology Department
Carnegie Mellon University
Pittsburgh, PA
USA

Piek Vossen
Faculty of Arts, Language Cognition and Communication (LCC)
Vrije University Amsterdam
Amsterdam
Netherlands

Lu Qin
Department of Computing
Hong Kong Polytechnic University
Hong Kong

Eduard Hovy
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA
USA

ISSN 2192-032X
ISBN 978-3-642-31781-1
DOI 10.1007/978-3-642-31782-8
Springer Heidelberg New York Dordrecht London

ISSN 2192-0338 (electronic)
ISBN 978-3-642-31782-8 (eBook)

Library of Congress Control Number: 2012954237

Printed on acid-free paper

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

In Memory of Emanuele Pianta

Contents

List of Figures

Chapter 1
Introduction

Alessandro Oltramari, Piek Vossen, Lu Qin, and Eduard Hovy

As human practice testifies, communicating through (natural) language is the way that enables mutual comprehension and effective knowledge transfer between agents. In order to effectively exchange information, agents need to share a lexicon of words as well as to access the world model(s) underlying the lexicon. This model can be represented by an ontology, whose proper function is to group together similar concepts, define their mutual relationships, support property inheritance and reasoning.

This book focuses on the integration between ontologies and lexicons as the *condicio sine qua non* to represent, elicit and exchange knowledge contents in information systems, web services, text processing and several domains of application. In this variegate context, computational lexical resources and computational ontologies[1] converge in the task of providing the semantic description of knowledge contents: according to this picture, the former contain the surface–level units that support the (mono– or multi–) lingual access to any knowledge content,

[1]We make use of the adjective 'computational', here, to refer to ontologies and lexicons which are encoded in suitable machine language, enabling computational processability.

A. Oltramari (✉)
Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: aoltrama@andrew.cmu.edu

P. Vossen
VU University of Amsterdam, Amsterdam, Netherlands
e-mail: piek.vossen@vu.nl

L. Qin
Hong Kong Polytechnic University, Hong Kong, Hong Kong SAR
e-mail: csluqin@comp.polyu.edu.hk

E. Hovy
Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: ed.hovy@cs.cmu.edu

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*,
Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_1, © Springer-Verlag Berlin Heidelberg 2013

while the latter capture the logical structure connecting those contents. Together, computational lexicons and ontologies contribute to characterize the elements of a given semantic space and specify the different relations holding among them.

The lexicon and the ontology each capture different types of information. The lexicon, housing language–specific syntactic and morphological information, is more than the set of labels for concepts defined in the ontology. The ontology, supposedly language–neutral, captures the formal meanings and interrelationships of concepts which are not contained in lexicons. When dealing with the semantic features of natural language text, a range of ontological models applies. Most natural language texts do not contain just core conceptual relations between entities in a specific domain, e.g., between diseases, causes of diseases, and treatments, but typically also cover the more complex aspects of human communication, including uncertainties, cognitive processes, emotions and social interactions.

The aim of this book is not to supply a comprehensive survey of the state–of–the–art research in ontologies and lexical resources, but to provide a (possibly complementary) work that looks towards the next–generation systems, shifting the focus from the present to the 'future of OntoLex[2]: in fact, this publication is constituted by a firm selection of the most significative research topics in the field, focusing on new framework of integration, hybrid models, use–cases, applications and domains of interest.

The first section illustrates the importance of harmonizing linguistic resources using well-defined semantic models and formats, e.g. Linked Data (Chap. 1), LingNet (Chap. 2) or by means of a deeper conceptual analysis of the notion of 'linguistic sense' (Chap. 3). In Sect. 4.2, Chap. 4 presents an integrated system to extract and represent relevant events from textual descriptions, reasoning over them through a suitable modular ontology and make them available through a multi–lingual resource. Chapter 5 illustrates a similar framework, but focusing on multi-media contents. In Chap. 6, the authors point out that annotating multi–media with lightweight ontologies is useless if we want to perform high–level reasoning: to improve the results, annotations have to be grounded on more expressive common sense ontologies. In–between Chaps. 4, 5, and 7 presents an integrated system for extracting and reasoning over events: instead of massively exploiting text processing and NLP techniques, Chap. 7 outlines an integrated cognitve system where ontologies are used to disambiguate the output of computer vision algorithms run over a video dataset of prototypical human activities. In Sect. 8.3, Chap. 8 presents an algorithm for reference resolution based on Ontological Semantics, an original branch of research that combines ontological and lexical semantics; Chap. 9 delineates an ontology-based semantic interepreter, based on grammar constraints

[2]This acronym originates from the homonymous fortunate series of workshops. Originating in 2000 through a visionary initiative by Atanas Kiryakov and Kiril Simov, and hosted twice by LREC (2002 and 2004), OntoLex has turned into a regular meeting for a growing interdisciplinary community of lexicographers, ontologists and computational linguists. This book has been actually inspired by some of the papers presented at OntoLex 2010 – http://www.loa.istc.cnr.it/program220810.pdf

and Chap. 10 focuses on the role of semantic technologies for enhancing information retrieval systems. The conclusive section explores the role of ontologies and lexical resources for sentiment analysis. Chapter 11 argues that, if it is possible to extract affective information from text only when explicit emotional words are used, it's also necessary to properly elicit ontological knowledge from textual information in order to identify and filter out implitic emotions. Finally, Chap. 12 also focuses on affective computing research, presenting a model where rich semantic features of emotion-related texts may vary according to the context.

We are pleased to have assembled a book of such high quality. It demonstrates the depth and diversity of thinking across several related communities regarding the relationships between (word–level) lexicons and (concept–level) ontologies. We believe that the book gathers together complementary perspectives and hope that it will serve both as a strong record of current theorizing and as an impetus for new and innovative research.

Part I
Achieving the Interoperability of Linguistic Resources in the Semantic Web

Chapter 2
Towards Open Data for Linguistics: Linguistic Linked Data

Christian Chiarcos, John McCrae, Philipp Cimiano, and Christiane Fellbaum

Abstract 'Open Data' has become very important in a wide range of fields. However for linguistics, much data is still published in proprietary, closed formats and is not made available on the web. We propose the use of linked data principles to enable language resources to be published and interlinked openly on the web, and we describe the application of this paradigm to the modeling of two resources, WordNet and the MASC corpus. Here, WordNet and the MASC corpus serve as representative examples for two major classes of linguistic resources, lexical-semantic resources and annotated corpora, respectively.

Furthermore, we argue that modeling and publishing language resources as linked data offers crucial advantages as compared to existing formalisms. In particular, it is explained how this can enhance the interoperability and the integration of linguistic resources. Further benefits of this approach include unambiguous identifiability of elements of linguistic description, the creation of dynamic, but unambiguous links between different resources, the possibility to query across distributed resources, and the availability of a mature technological infrastructure. Finally, recent community activities are described.

C. Chiarcos (✉)
Information Sciences Institute, University of Southern California, Marina del Rey, CA, USA
e-mail: chiarcos@isi.edu

J. McCrae · P. Cimiano
Semantic Computing Group, Cognitive Interaction Technology Center of Excellence (CITEC), University of Bielefeld, Bielefeld, Germany
e-mail: jmccrae@cit-ec.uni-bielefeld.de; cimiano@cit-ec.uni-bielefeld.de

C. Fellbaum
Computer Science Department, Princeton University, Princeton, NJ, USA
e-mail: fellbaum@princeton.edu

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_2,

2.1 Motivation and Overview

Language is arguably one of the most complex forms of human behaviour, and accordingly, its investigation involves a broad width of formalisms and resources used to analyze, to process and to generate natural language. An important challenge is to store, to connect and to exploit the wealth of language data assembled in half a century of computational linguistics research. The key issue is the **interoperability** of language resources, a problem that is at best partially solved [25]. Closely related to this is the challenge of **information integration**, i.e., how information from different sources can be retrieved and combined in an efficient way.

As a principal solution, Tim Berners-Lee – the founder of the World Wide Web – proposed the so called *linked data principles* to publish open data on the Web. These principles represent rules of best practice that should be followed when publishing data on the Web [4]:

1. Use URIs as (unique) names for things.
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using Web standards such as RDF, and SPARQL.
4. Include links to other URIs, so that they can discover more things.

We argue that applying the linked data principles to lexical and other linguistic resources has a number of advantages and represents an effective approach to publishing language resources as open data. The first principle means that we assign a unique identifier (URI) to every element of a resource, i.e., each entry in a lexicon, each document in a corpus, every token in a corpus as well as to each data category that we use for annotation purposes. The benefit is that this makes the above mentioned resources uniquely and globally identifiable in an unambiguous fashion. The second principle entails that any agent wishing to obtain information about the resource can contact the corresponding web server and retrieve this information using a well-established protocol (HTTP) that also supports different 'views' on the same resource. That is, computer agents might request a machine readable format, while web browsers might request a human-readable and browseable view of this information as HTML. The third principle requires the use of standardized, and thus, inter-operable data models for representing (RDF, [29]) and querying linked data (SPARQL, [35]). The fourth principle fosters the creation of a network of language resources where equivalent senses are linked across different lexical-semantic resources, annotations are linked to their corresponding data categories in data category repositories, etc.

In the definition of linked data, the **Resource Description Framework (RDF)** receives special attention. RDF was originally designed as a language to provide metadata about resources that are available both offline (e.g., books in a library) and online (e.g., eBooks in a store). RDF provides a data model that is based on labelled directed (multi-)graphs, which can be serialized in different formats, where

Table 2.1 Selected relations from existing RDF vocabularies and possible fields of application

Domain	Example	Reference
Meta data	`creator`	Dublin core meta data categories
General relations between resources	`sameAs`	Web ontology language (OWL)
Concept hierarchies	`subClassOf`	RDF schema
Relations between vocabularies	`broader`	Simple knowledge organization scheme
Linguistic annotation	`lemma`	NLP interchange format

the nodes identified by URIs are referred to as 'resources'.[1] On this basis, RDF represents information in terms of *triples* – a *property* (relation, in graph-theoretical terms a labelled edge) that connects a *subject* (a resource, in graph-theoretical terms a labelled node) with its *object* (another resource, or a literal, e.g., a string). Every RDF resource and every property is uniquely identified by a URI. They are thus globally unambiguous in the web of data. This allows resources hosted at different locations to refer to each other, and thereby to create a network of data collections.

A number of RDF-based vocabularies are already available, and many of them can be directly applied to linguistic resources. A few examples are given in Table 2.1. In this way, the RDF specification provides only elementary data structures, whereas the actual *vocabularies* and domain-specific *semantics* need to be defined independently. For reasons of interoperability, existing vocabularies should be re-used whenever possible, but if a novel type of resource requires a new set of properties, RDF also provides the means to introduce new relations, etc.

RDF has been applied for various purposes beyond its original field of application. In particular, it evolved into a generic format for data exchange on the Web. It was readily adapted by disciplines as diverse as biomedicine and bibliography, and eventually it became one of the building stones of the Semantic Web. Due to its application across discipline boundaries, RDF is maintained by a large and active community of users and developers, and it comes with a rich infrastructure of APIs, tools, databases, and query languages. Further, RDF vocabularies do not only define the labels that should be used to represent RDF data, but they also can introduce additional constraints to formalize specialized RDF sub-languages. For example, the **Web Ontology Language (OWL)** defines the data types necessary for the representation of ontologies as an extension of RDF, i.e., *classes* (concepts), *instances* (individuals) and *properties* (relations).

In the remainder of this chapter, we explore the benefits of linked data, considering in particular the following advantages:

Representation and modelling Lexical-semantic resources can be described as labelled directed graphs (feature structures, [27]), as can annotated corpora [3].

[1]The term 'resource' is ambiguous here. As understood in this chapter, resources are structured collections of data which can be represented, for example, in RDF. Hence, we prefer the terms 'node' or 'concept' whenever *RDF resources* are meant.

RDF is based on labelled directed graphs and thus particularly well-suited for modelling both types of language resources.

Structural interoperability Using a common data model eases the integration of different resources. In particular, merging multiple RDF documents yields another valid RDF document, while this is not necessarily the case for other formats.

Federation In contrast to traditional methods, where it may be difficult to query across even multiple parts of the same resource, linked data allows for federated querying across multiple, distributed databases maintained by different data providers.

Ecosystem Linked data is supported by a community of developers in other fields beyond linguistics, and the ability to build on a broad range of existing tools and systems is clearly an advantage.

Expressivity Semantic Web languages (OWL in particular) support the definition of axioms that allow to constrain the usage of the vocabulary, thus introducing formal data types and the possibility of checking a lexicon or an annotated corpus for consistency.

Conceptual interoperability The linked data principles have the potential to make the interoperability problem less severe in that globally unique identifiers for concepts or categories can be used to define the vocabulary that we use and these URIs can be used by many parties who have the same interpretation of the concept. Furthermore, linking by OWL axioms allows us to define the exact relation between two different concepts beyond simple equivalence statements.

Dynamic import URIs can be used to refer to external resources such that one can thus import other linguistic resources "dynamically". By using URIs to point to external content, the URIs can be resolved when needed in order to integrate the most recent version of the dynamically imported resources.

We elaborate further on these aspects in this chapter. It is structured as follows: Sect. 2.2 describes the modelling of linguistic resources as linked data and identifies deficits and prospective advantages of using linked data for linguistic resources. Section 2.3 elaborates some of the benefits of this representation. Section 2.4 summarizes recent community activities promoting the publication of language resources as linked data.

2.2 Modelling Linguistic Resources as Linked Data

We consider two important classes of language resources, the first of which is **lexical-semantic resources**, i.e., resources that provide information about lexemes and their relation to other lexemes (e.g., machine-readable dictionaries, semantic networks, semantic knowledge bases, ontologies and terminologies). The second class of language resources considered here are **annotated corpora**, i.e., collections of textual (spoken, written or gestural) data annotated with linguistic characteristics.

For both types of resources, we describe state-of-the-art approaches, briefly motivate the application of linked data principles, and then describe modelling these resources using RDF and OWL.

Resource modelling involves two aspects: (1) the specification of data structures and consistency constraints over these, and (2) the conversion of data into these representations. RDF encodes labelled directed graphs and is thus capable to represent both lexical-semantic resources and linguistic corpora, as both can be described with directed graphs. For reasons of symmetry, also different types of annotated corpora are enumerated.

Unlike other graph-based modelling formalisms applied to language resources, e.g., GraphML [5], RDF provides additional means to formalize specific data types, and thereby to establish a **reserved vocabulary** and to introduce **structural constraints** for nodes, edges or labels. Such constraints are necessary, e.g., for corpora, to avoid confusion between RDF representations of corpus infrastructure (corpus, subcorpus, document, annotation layer) and meta data (information about the resource as a whole).

As an illustration of the benefits of modelling linguistic data as linked data, let us consider the following example. Imagine we would like to get all occurrences in a corpus (e.g. MASC, Sect. 2.2.2) of synonyms of 'land' in the sense of *'(the territory occupied by a nation)'* (in WordNet 3.1, Sect. 2.2.1) with synonyms 'country' and 'state'. In order to get such occurrences, one would first use the WordNet data model – suitably abstracted by some API – and query for elements in the synset corresponding to 'land' as *'(the territory occupied by a nation)'*. This 'query' would yield: 'land', 'country' and 'nation'. Then, using another data model and appropriate APIs or query interfaces, we would then search for occurrences of 'land', 'country' or 'nation' in the MASC corpus annotated with the corresponding sense ID key from WordNet. This shows that it is cumbersome and difficult to answer such queries which span multiple resources as one is forced to use different data models, APIs etc.

The benefit of using RDF and linked data principles to model linguistic resources is that it provides a graph-based model that allows representing different types of linguistic resources (corpora, treebanks, lexical-semantic resources) in a uniform way, thus supporting uniform querying across resources. The query sketched above, for example, can be represented in a single, and simple SPARQL expression as shown in Sect. 2.3.1.2.[2] And as RDF and SPARQL employ URIs to designate elements, it is even possible to query data not stored in a single repository, but that are accessible through different SPARQL endpoints. With a mechanism that can distribute the relevant parts of a query to the repositories that contain the relevant MASC and WordNet data (Sect. 2.3.2), answering such a query is indeed straightforward.

[2]We provide a SPARQL endpoint under http://monnetproject.deri.ie/lemonsource_query, which provides access to the examples discussed in this chapter.

In the following we discuss in more detail how both corpora (such as MASC) and lexical-semantic resources (such as WordNet) can be modelled using RDF and what the particular advantages are.

2.2.1 Modelling Lexical-Semantic Resources: WordNet

2.2.1.1 WordNet Data Structures

WordNet [17, 34] is a particularly influential lexical-semantic resource, and very prototypical in many aspects. It is a manually constructed electronic lexical resource, organized around concepts and the words expressing them. WordNet draws its motivation from theories of human lexical memory, which indicate that people store knowledge about concepts in a well-structured, economic fashion and attempts to implement this model. The current version 3.1 includes over 117,000 concepts expressed by nouns, verbs, adjectives, and adverbs.[3]

A concept in WordNet is represented as a set of (roughly) synonymous words that all refer to the same entity, event, or property. Synset members can be interchanged without altering the truth value of a context. Formally, WordNet is a directed acyclic graph, where synsets are interlinked by edges standing for means of conceptual-semantic relations. The most important is the super-/subordinate (hyponymy) relation. It links generic to increasingly specific synsets like *land* to *kingdom* and *sultanate*. Synset pairs referring to part-whole concepts (*land-midland*, *wheel-car*, etc.) are also connected, as are synsets expressing semantic opposition (*hot-cold*, *arrive-leave*, etc.) and a range of temporal relations (see [17]).

2.2.1.2 Generic Data Structures: Lexical Markup Framework

To facilitate interoperability among lexical-semantic resources, feature structures (i.e., directed acyclic graphs) have been suggested as a generalization over resource-specific data structures [40]. Feature structures are a flexible and general formalism, which became the basis for subsequent standardization, in particular, in the Lexical Markup Framework (LMF, [19]). LMF represents a metamodel to represent semantic information in NLP lexicons and machine-readable dictionaries. It has been successfully applied to develop resources such as Uby [22], an openly available, large-scale lexical-semantic resource. Uby integrates nine independent resources for English and German, including WordNet, Wiktionary, Wikipedia, FrameNet, VerbNet, and OmegaWiki, which are linked with each other on sense level. However, the LMF format is not an open format (in the sense that its specification is not freely available), and in its standard serialization as XML, it does not consider

[3] http://www.wordnet.princeton.edu

how resources can be uniquely identified on the web. Furthermore, according to the experience of Uby, application of the format requires domain-specific modifications to the standard schema.

An RDF formalization tackles some of these problems, and this has been suggested by the LMF developers themselves.[4] Providing lexical-semantic resources as linked data actually allows us to integrate LMF resources with other resources previously converted to RDF, e.g., in the context of the developing Semantic Web.

2.2.1.3 From LMF to RDF: *lemon*

Independently from LMF, there has already been some work towards the integration of WordNet with the Semantic Web, notably [39], who provided a simple mapping from WordNet to RDF, and augmented it with OWL semantics so that reasoning could be applied to the structure of the resource. However the format chosen for this resource was specific to the underlying data model of WordNet. For this reason, [33] propose the interchange model *lemon* (Lexicon Model for Ontologies) that supports publishing lexical-semantic resources as linked data on the basis of the following principles:

LMF-based (to allow easy conversion from non-linked data resources);
RDF-native (publishing as linked data, with RDFS and OWL used to describe the semantics of the model);
Modular (separation of lexicon and ontology layers, so that *lemon* lexica can be linked to existing ontologies in the linked data cloud);
Externally defined data categories (linking to data categories in annotation terminology repositories, rather than being limited to a specific part-of-speech tag set);
Principle of least power (the smaller the model and the less expressive the language, the wider its adoption and the higher the reusability of the data, [38]).

This model is illustrated in Fig. 2.1. *lemon* has been used as a basis for integrating the data of the English Wiktionary,[5] a (human-readable) dictionary created along 'wiki' principles, with the RDF version of WordNet [33]. As *lemon* derives from LMF but integrates with the existing Semantic Web formalisms, there was some need to adapt the data model. It was found that WordNet's model was fairly close to *lemon* and LMF, with only minor differences in the modelling of inflectional variants of lexical entries. However, the semantic modelling was more significantly different as *lemon* uses OWL to represent semantics.

[4] http://www.tagmatica.fr/lmf/LMF_revision_14_In_OWL29october2007.xml
[5] http://en.wiktionary.org/

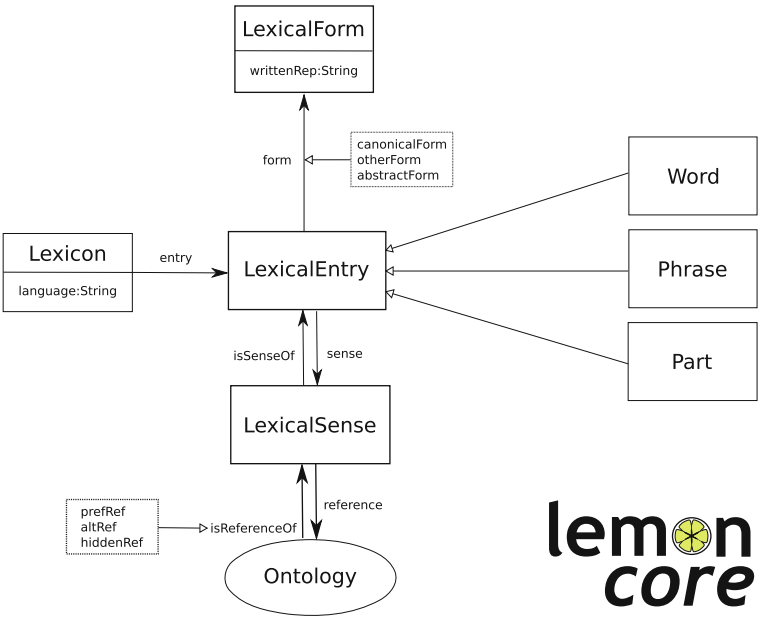

Fig. 2.1 The core of the *lemon* model

2.2.2 Modelling Annotated Corpora: MASC

2.2.2.1 The Manually Annotated Sub-Corpus

The Manually Annotated Sub-Corpus (MASC, [28]) is a corpus of 500,000 tokens of contemporary American English text drawn from the Open American National Corpus, written and spoken, and chosen from a variety of genres.[6] MASC comprises various layers of annotations, including parts-of-speech, nominal and verbal chunks, constituent syntax, annotations of WordNet senses, frame-semantic annotations, coreference, document structure and illocutionary structure. The tools that generated the annotations of the MASC corpus use different output formats. In order to establish interoperability between them, MASC distributions adopt a generic data model, the Graph Annotation Format (GrAF, [26]). By use of multi-layer annotations, MASC allows all annotations of a particular piece of text to be integrated into a common representation that provides lossless and comfortable access to their linguistic information.

[6] www.anc.org/MASC

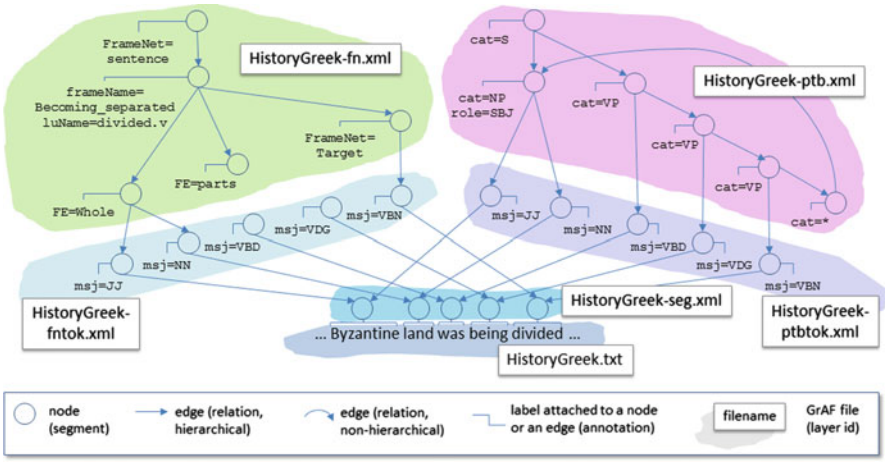

Fig. 2.2 Representing and integrating annotations for syntax and frame-semantics in a directed graph

2.2.2.2 Generic Data Structures for Annotated Corpora: GrAF

State-of-the-art approaches on interoperable formats for annotated corpora are based on the assumption that all linguistic annotations can be represented by means of labelled directed acyclic graphs [3]. To a certain extent, this echoes the application of feature structures to lexical-semantic resources (feature structures are labelled directed acyclic graphs).

One representative example for graph-based generic formats is the GrAF format. Like other state-of-the-art approaches that implement graph-based data models for linguistic corpora [7, 11], GrAF is a special-purpose XML standoff format. Standoff formats are based on a physical separation between primary data (e.g., text, audio or video) and different layers of annotations. In Fig. 2.2, this is shown for an example sentence from the MASC corpus: All annotations of a document are grouped together in a set of XML files pointing to the same piece of primary data. Different file names in the figure represent the respective annotation layer. Distributing annotations across different files, however, results in a highly complex structure with multiple dependencies between individual files. Consequently, standoff formats introduce a relatively large technical overhead that makes it difficult to work with large data in practice. While standoff formats have become widely accepted, the efficient processing, storing and retrieval of standoff data requires formalisms that support the free linking of elements, and that are thus fundamentally different from hierarchical data models such as XML that are optimized for tree structures, rather than general graphs.

Figure 2.2 shows the graph-based modelling and its XML standoff serialization for two selected layers of annotations for the clause *'Byzantine land was being divided'*. To the left, the figure shows FrameNet annotations [2] and to the right

PennTreebank-style syntax annotations [30]. Both annotations are synchronized with each other and the primary data through a shared base segmentation file.

2.2.2.3 From Standoff XML to RDF: POWLA

Standoff XML can be hard to process, and the corresponding infrastructures and standards are still under development. RDF, however, already provides a rich technological ecosystem for labelled directed graphs, and GrAF data structures can be easily converted to RDF. Rendering generic data models for annotated corpora in RDF has been suggested before, e.g., by Cassidy [8] and Chiarcos [10].

Chiarcos [10] described POWLA, an RDF/OWL linearization of PAULA, a generic data model for the representation of annotated corpora [14, 15]. PAULA is similar in scope and design to GrAF and also builds on traditional standoff XML. POWLA consists of two basic components: (1) an OWL/DL ontology that defines the valid data types, relations and constraints as classes, properties and axioms; (2) an RDF document that represents a corpus as a knowledge base consisting of individuals, instantiated object properties and data values assigned to individuals through datatype properties. POWLA formalizes the structure of annotated corpora and linguistic annotations of textual data. With respect to the latter, it provides data types such as `Node` and `Relation` (as well as more specialized data types) that directly reflect the underlying graph-based data model. With OWL/DL axioms, the relationship between these data types can be formalized and automatically verified, e.g., that `Relation` and `Node` are disjoint, and that every `Relation` is connected by one `hasSource` and one `hasTarget` property with a particular `Node`.

A GrAF converter is provided under http://purl.org/powla, it replicates the structure of the GrAF file exactly in RDF/OWL. As with the original GrAF representation, annotated corpora represented in this way are structurally interoperable (different annotations use the same representation formalism), but in this form, they can be queried using RDF query languages like SPARQL, they can be stored in RDF databases, and OWL/DL reasoners can be applied to validate the consistency of the data.

2.3 Benefits of Linked Data for Linguistics

Aside from representation, Sect. 2.1 identified five specific advantages of modelling linguistic resources as linked data. These include structural interoperability (same format for different types of resources), the querying of physically distributed resources (federation), enhanced conceptual interoperability (same vocabulary for different resources), a rich ecosystem of formalisms and technologies, and the possibility to create resolvable links between resources that are maintained by different data providers (dynamic import).

2.3.1 Structural Interoperability

Structural ('syntactic') interoperability of a language resource in NLP corresponds to the 'ability [of an NLP tool] to process it immediately without modification to its physical format', i.e., structural interoperability 'relies on specified data formats, communication protocols, and the like to ensure communication and data exchange' [25]. This involves two aspects: The capability to **provide access to the data** depending on the needs of the data consumer (a human user or some software tools), and the use of the **same format** for different resources such that they can be processed in a uniform way. To this definition of structural interoperability we should add another desideratum that partially follows from both aspects, namely that different resources are accessible with uniform query languages, and that information from different sources can be easily **merged**.

2.3.1.1 Structural Interoperability by Content Negotiation

Servers that publish data on the web can (and should) provide multiple versions of the data. This is possible as the HTTP protocol supports **content negotiation** [18, p. 67–70], i.e., a user or agent that accesses a particular resource can specify the format they want by means of the HTTP `Accept` header. This allows a lexical resource to be identified by a single URI, but display human-readable HTML to users accessing the page through a web browser and the original RDF data to web agents. Upon accessing a resource URI, the server responds with the first specified data format given by the user or an error if no acceptable format can be rendered. In this way, language resources can be published on the web using Semantic Web standards, human readable forms and other serializations.

A similar method called *transparent content negotiation* [24] allows the RDF and HTML versions of the page to be identified by a separate URI to the resource itself. Here instead of responding with the correct data type, the server redirects the client to a new URL for the appropriate data format. or example, the server may direct the client to add the suffix `.rdf` for the linked data and `.html` for the human-readable version.

2.3.1.2 RDF as a Structurally Interoperable Format

We have seen that RDF is suitable for representing two major types of linguistic resources, and thus we can achieve structural interoperability in the sense that information from these two RDF documents (and actually, the documents themselves) can be merged without the need to create a new schema. It is thus easy to formulate uniform queries that work over heterogeneous language resources. As an example, we can combine information from the linked data version of WordNet and the POWLA formalization of the MASC corpus, e.g., the task to find all tokens in a corpus that refer to *land* as a political unit (synonyms from the WordNet synset `land%1:15:02::`).

Using RDF representations of WordNet and MASC, however, it is no longer necessary to access separate APIs for MASC, GrAF and WordNet. Instead, the task to integrate information from different resources can be easily achieved by applying standard RDF query languages like SPARQL [35] to a repository in which both resources are contained. The sense keys are thus URIs in a RDF version of WordNet such as lwn:synset-land-noun-2. Hence a query as below can be formulated:

```
PREFIX lemon: <http://www.monnet-project.eu/lemon#>
PREFIX lwn: <http://monnetproject.deri.ie/lemonsource/wordnet/> .
SELECT ?token {
  lwn:synset-land-noun-2 lemon:isReferenceOf ?sense
  ?sense lemon:isSenseOf ?entry .
  ?entry rdfs:label ?synonym .
  ?token powla:hasString ?synonym .}
```

2.3.2 *Linking and Federation*

Linked data is built on URIs as globally unique identifiers. They have the key advantage that resources can be unambiguously identified, thus supporting the creation of a linked web in analogy to the current web of documents (but using properties to link resources instead of the document-oriented, unlabelled HTML hyperlinks). Linked data thus does typically not exist as a set of files on a hard disk or as data in a single data base, but instead as a network of related resources on the web. In other words, techniques must be (and have been) provided that allow queries over linked data to be **federated** over multiple different repositories, physically located at different servers across the world [6, 21, 23, 36].

Rather than querying for WordNet senses and linguistic annotations stored in a single RDF repository, we thus can directly address the public SPARQL endpoint of *lemon source* [32] to access WordNet senses in a subquery:

```
PREFIX lemon: <http://www.monnet-project.eu/lemon#> .
PREFIX lwn: <http://monnetproject.deri.ie/lemonsource/wordnet/> .
SELECT ?token {
  service <http://monnetproject.deri.ie/lemonsource_query/> {
    lwn:synset-land-noun-2 lemon:isReferenceOf ?sense .
    ?sense lemon:isSenseOf ?entry .
    ?entry rdfs:label ?synonym .
  }
  ?token powla:hasString ?synonym .
}
```

If the query engine was configured to do so, it may be able to infer which endpoints to query for certain data based on the URIs used in the query [37]. By building on a standard method for federation of queries on the web, we ensure that the systems take advantage of effective algorithms for federating queries. In this way, information from corpora and lexical-semantic resources can be successfully integrated with each other even if these resources are physically distributed over different repositories.

2.3.3 Conceptual Interoperability

RDF does not only establish structural interoperability among and between lexical-semantic resources and corpora, but also between these and resources like terminology repositories or meta-data repositories. In combination with the possibility to query distributed resources, this can also be exploited to enhance the **conceptual interoperability** between language resources, i.e., the use of shared vocabularies for linguistic analyses and metadata.

Ide and Pustejovsky [25] define conceptual ('semantic') interoperability of NLP tools as 'the ability to automatically interpret exchanged information meaningfully and accurately in order to produce useful results'. Further, they suggest that this can be achieved 'via deference to a common information exchange reference model' for language resources and NLP tools.

Different communities create their own grammatical annotations, and although they follow the common goal to establish conceptual interoperability, they have been developed for different use cases, and – even worse – they represent different terminological traditions. Two representative repositories are the General Ontology of Linguistic Description (GOLD, [16]) and the ISO TC37/SC4 Data Category Registry (ISOcat, [41]). Adopting a linked data approach, however, it is possible to link these repositories with each other, i.e., either to link from one resource to the other, or to create mediator ontologies that provide a linking between these repositories. The Ontologies of Linguistic Annotation [9, OLiA] are a modular set of ontologies that establish such a linking. OLiA consists of a *Reference Model*, which specifies the common terminology that different annotation schemes can refer to, as well as *Annotation Models* that formalize annotation schemes and tagsets for about 70 different languages. For every Annotation Model, a *Linking Model* defines relationships between concepts/properties with the Reference Model. In the same way, the Reference Model is linked with several terminology repositories, including GOLD and ISOcat.

Considering annotations in a corpus, say, the syntax annotations of the word *land* from Fig. 2.2, attribute-value pairs like `msj=NN` attached to a particular POWLA `Node` can be exploited to assign this `Node` the superclass `penn:CommonNoun` from the Annotation Model that formalises the corresponding annotation scheme. Through the linking, it can be inferred that this `Node` is also an `olia:CommonNoun` in the Reference Model and that it is an instance of both `isocat:DC-1256` and `gold:CommonNoun`. It would thus become compatible and aligned with any annotation scheme that is linked to either GOLD or ISOcat.

By this kind of linking we can create chains of resources leading to links that would not have been trivial to discover otherwise. As an example, assume that we are interested in studying a particular lexeme in a lexical-semantic resource and that we would like to inspect its usage in a particular corpus. Many lexicons, e.g., those developed on the basis of LexInfo [31], include references to ISOcat data categories. The link between these and the OLiA Reference Model can be discovered – for example – by querying a Semantic Web Search Engine for references to the

ISOcat data category. Dereferencing the OLiA Reference Model, we can find the corresponding Annotation Model concepts that define, inter alia, the corresponding part-of-speech tags. This information can then be exploited to generate corpus queries to retrieve example sentences for the lexeme which combine lemma and spelling information with the appropriate part-of-speech tags. Such queries could then be applied even to corpora that are not provided as linked data.

2.3.4 Ecosystem

RDF comes with a rich repository of tools and formalisms for the processing of graph-based data structures. Using it as representation formalism for multi-layer annotations provides us with convenient means for modelling, manipulating, storing and querying directed labelled graphs. Linked data has achieved success in a wide variety of fields and in fact the linked data paradigm is being applied to a number of domains[7] and is thus supported by a comparably large and active user community.

One consequence is the existence of multiple standards and recommendations maintained by the W3C (e.g., RDFS, OWL, SPARQL) for which new extensions are being developed at a rapid pace.[8] Moreover, there exist a large number of commercial and open-source tools to process linked data, in particular repositories for storing and querying. There are frequent benchmarks of the performance of these tools.[9] In addition, search engines index all the linked data available and allow the discovery of new services.[10]

2.3.5 Dynamic Import

In the traditional approach on modelling language resources, cross-links between different resources are typically represented by attribute-value pairs whose value contains the string representation of IDs as defined within another language resource. Within the linked data approach, however, such information can be represented by a resolvable URI, and is thus accessible in its complete and up-to-date form. When the resource that is referred to is augmented by additional

[7] Other domains where the linked data principles have been applied, include, e.g., geography [20], biomedicine [1], cultural history (http://www.europeana.eu) or government data (e.g., http://data.gov and http://data.gov.uk).

[8] For example, the W3C Semantic Web Activity reported on developments for Media Resources, Data Provenance and Microdata in the first 2 weeks of February 2012

[9] http://www4.wiwiss.fu-berlin.de/bizer/berlinsparqlbenchmark

[10] Examples include http://swoogle.umbc.edu, http://www.sindice.net, http://swse.deri.ie, and http://watson.kmi.open.ac.uk.

information, then a system can access this information even though it was not available at the time when the annotation (say, a WordNet sense) was created. Maintenance efforts nowadays necessary to maintain the proper linking of corpora with the most recent WordNet edition available can thus be reduced to a minimum. Furthermore, the use of URIs instead of system-defined IDs solves another problem, namely that such informal ID references are usually not unambiguous. For example, the version of the WordNet referred to a resource can be indicated by its full URI avoiding the need to explicitly state the version number.

However, dynamism can be a "double-edged sword". Although continuous corrections may improve the quality of a resource, this entails the risk that references from external resources are no longer valid, e.g., because a sense has been redefined, split or merged with another. Following an established publication practice for linguistic resources, it is thus advisable to provide stable release editions and to indicate these differences in the corresponding URIs.

2.4 Community Efforts Towards Lexical Linked Data

Publishing language resources using such interoperable representations, formally defined data types and resolvable URI to designate elements of linguistic analysis/annotation allows existing linguistic resources to be connected. Aside from the benefits enumerated in the last section, this facilitates the distributed, but highly synchronized development of linguistic resources. The technological infrastructure developed around RDF makes it an attractive candidate for the creation, exchange and processing of language resources in different sub-disciplines of linguistics, NLP and neighbouring fields. Its genericity allows researchers from all these different subcommunities to share data and experiences; thereby, RDF encourages interdisciplinary cooperation.

Consequently, linked data is at the core of recent community activities. We describe two initiatives heading towards the creation of a linked (open) data cloud of linguistic data.

2.4.1 The Open Linguistics Working Group

The Open Linguistics Working Group (OWLG, [12])[11] of the Open Knowledge Foundation was founded in late 2010 as an initiative of experts from different fields concerned with linguistic data, including academic linguists (e.g. typology, corpus linguistics), applied linguistics (e.g. computational linguistics, lexicography and language documentation), and information technology (e.g. Natural Language

[11] http://linguistics.okfn.org

Processing, Semantic Web). The primary goals of the working group are to promote the idea of **open linguistic resources**, to explore **means for their representation**, and to encourage the **exchange of ideas** across different disciplines.

A number of concrete community projects have been initialized,[12] including the documentation of workflows, documenting best practice guidelines and collecting use cases with respect to legal issues of linguistic resources. Of particular importance in this context is the collection of representative resources available under open licenses, the identification of possible links between these resources and, consequently, the creation of a **Linguistic Linked Open Data cloud**.[13]

For resources published under open licenses, an RDF representation yields the additional advantage that resources can be interlinked, and it is to be expected that an additional gain of information arises from the resulting network of resources. So, although the OWLG is dedicated to open resources in linguistics in general, and not a priori restricted to linked data, a general consensus has been established within the OWLG that Semantic Web formalisms provide crucial advantages for the publication of linguistic resources, some of which have been illustrated here as well.

The idea of linked data is gaining ground: data sets from different subdisciplines of linguistics and neighbouring fields are currently prepared. Recent activities include subject areas as diverse as language acquisition, the study of folk motifs, phonological typology, translation studies, pragmatics and comparative lexicography [13]. The OWLG represents a platform for the exchange of ideas, data and information across all these different fields.

2.4.2 W3C Ontology-Lexica Community Group

The Ontology-Lexica Community (OntoLex) Group,[14] was founded as a W3C Community and Business Group in September 2011. It aims to produce specifications for a **lexicon-ontology model** that can be used to provide rich linguistic grounding for domain ontologies. Rich linguistic grounding includes the representation of morphological, syntactic properties of lexical entries as well as the syntax-semantics interface, i.e. the meaning of these lexical entries with respect to the ontology in question. An important issue herein will be to clarify how extant lexical and language resources can be leveraged and reused for this purpose. As a by-product of this work on specifying a lexicon-ontology model, it is hoped that such a model can become the basis for a web of lexical linked data: a network of lexical and terminological resources that are linked according to the linked data principles forming a large network of lexical-syntactic knowledge.

[12] http://wiki.okfn.org/Wg/linguistics

[13] http://linguistics.okfn.org/llod

[14] http://www.w3.org/community/ontolex

Five general requirements for the lexicon-ontology model were identified:

RDF/OWL The actual model is an OWL ontology, a specific lexicon instantiating the model is a plain RDF document.

Multilingualism The model supports the specification of the linguistic grounding with respect to any language.

Semantics by reference The meaning of a lexical entry is specified by referencing the URI of the concept or property in question.

Flexible infrastructure The lexicon-ontology model is extensible by new constructs as needed, e.g. by a certain application, and it makes no unnecessary choices with respect to which linguistic data categories to use, i.e., leaving open the possibilities to have very different instantiations of the model.

Interoperability Reuse of relevant standards (e.g. LMF).

2.5 Summary

In this chapter, we suggested that modelling linguistic resources as linked data provides a number of crucial advantages as compared to existing formalisms. In particular, modelling linguistic resources in RDF can lead to enhanced **interoperability** (and thus, scalability) for applications, **knowledge integration**, and access to **distributed resources**, and last but not least the rich **infrastructure** provided by the Semantic Web community can be applied to develop infrastructures for NLP, computational lexicography or corpus linguistics. In this way, linked data might facilitate the work of application developers, users of language resources and the natural language processing community as a whole.

A specific characteristic of RDF and linked data in general is that resources and their components (e.g., entries in a dictionary) are represented by URIs, thus enabling the **globally unambiguous referencing** of data. By using resolvable URIs to refer to other resources, resources can be **interlinked** and thereby integrated. For example, a corpus can be directly connected to a lexical-semantic resource, different lexical-semantic resources can be queried simultaneously and information from various sources can be combined. Further, we described recent **community efforts** in the NLP and Semantic Web communities heading towards the provision of a larger set of linguistic resources as linked data.

Overall, in this chapter we have discussed the benefits of publishing linguistic data as linked data and outlined a vision, sketching the potential, implications and applications thereof. The vision we have outlined is not a far-fetched one. From a technological point of view, the main ingredients are already in place, in particular RDF, OWL and SPARQL. Furthermore, as linked data grows in popularity across multiple disciplines, tools that can be applied to linguistic linked data will only increase in number and power.

Acknowledgements The work of Christian Chiarcos was supported by a postdoc fellowship of the German Academic Exchange Service (DAAD). The work of John McCrae and Philipp Cimiano

was developed in the context of the Monnet project, which is funded by the European Union FP7 program under grant number 248458 and the CITEC excellence initiative funded by the DFG (Deutsche Forschungsgemeinschaft). Christiane Fellbaum's work is supported by a grant from the U.S. National Science Foundation (CNS 0855157). We would also like to thank Nancy Ide and two anonymous reviewers for valuable comments and feedback.

References

1. Ashburner, M., Ball, C.A., et al.: Gene ontology: tool for the unification of biology. Nat. Genet. **25**(1), 25–29 (2000)
2. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet project. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL-1998), Montréal, pp. 86–90 (1998)
3. Bird, S., Liberman, M.: A formal framework for linguistic annotation. Speech Commun. **33**(1), 23–60 (2001)
4. Bizer, C., Heath, T., Berners-Lee, T.: Linked data – the story so far. Int. J. Semant. Web Inf. Syst. (IJSWIS) **5**(3), 1–22 (2009)
5. Brandes, U., Eiglsperger, M., et al.: Graph markup language (GraphML). In: Tamassia, R. (ed.) Handbook of Graph Drawing and Visualization. Chapman & Hall/CRC, London (2010)
6. Buil-Aranda, C., Arenas, M., Corcho, O.: Semantics and optimization of the SPARQL 1.1 federation extension. In: The Semantic Web: Research and Applications, pp. 1–15. Springer, Heraklion (2011)
7. Carletta, J., Evert, S., et al.: The NITE XML Toolkit: data model and query. Lang. Resour. Eval. J. (LREJ) **39**(4), 313–334 (2005)
8. Cassidy, S.: An RDF realisation of LAF in the DADA annotation server. In: Proceedings of the 5th Joint ISO-ACL/SIGSEM Workshop on Interoperable Semantic Annotation (ISO-5), Hong Kong (2010)
9. Chiarcos, C.: An ontology of linguistic annotations. LDV Forum **23**(1), 1–16 (2008)
10. Chiarcos, C.: Interoperability of corpora and annotations. In Chiarcos, C., Nordhoff, S., Hellmann, S. (eds.) Linked Data in Linguistics, pp. 161–179. Springer, Heidelberg (2012)
11. Chiarcos, C., Dipper, S., et al.: A flexible framework for integrating annotations from different tools and tagsets. TAL (Traitement automatique des langues) **49**(2), 217–246 (2008)
12. Chiarcos, C., Hellmann, S., et al.: The open linguistics working group. In: Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC-2012), Istanbul (2012a)
13. Chiarcos, C., Nordhoff, S., Hellmann, S. (eds.): Linked Data in Linguistics. Representing Language Data and Metadata. Springer, Heidelberg (2012b)
14. Chiarcos, C., Ritz, J., Stede, M.: By all these lovely tokens ... Merging conflicting tokenizations. J. Lang. Resour. Eval. (LREJ) **4**(45), 53–74 (2012c)
15. Dipper, S.: XML-based stand-off representation and exploitation of multi-level linguistic annotation. In: Eckstein, R., Tolksdorf, R. (eds.) Proceedings of Berliner XML Tage 2005 (BXML-2005), Berlin, pp. 39–50 (2005)
16. Farrar, S., Langendoen, D.T.: An OWL-DL implementation of GOLD: an ontology for the Semantic Web. In: Witt, A., Metzing, D. (eds.) Linguistic Modeling of Information and Markup Languages. Springer, Dordrecht (2010)
17. Fellbaum, C.: WordNet. MIT, Cambridge (1998)
18. Fielding, R., Gettys, J., et al.: Hypertext transfer protocol – HTTP/1.1. Internet RFC 2068 (1997)
19. Francopoulo, G., George, M., et al.: Lexical markup framework (LMF). In: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006), Genoa (2006)

20. Goodwin, J., Dolbear, C., Hart, G.: Geographical linked data: the administrative geography of Great Britain on the Semantic Web. Trans. GIS **12**, 19–30 (2008)
21. Guéret, C., Kotoulas, S., Groth, P.: TripleCloud: an infrastructure for exploratory querying over web-scale RDF data. In: Proceedings of the 2011 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011), Lyon, pp. 245–248 (2011)
22. Gurevych, I., Eckle-Kohler, J., et al.: Uby – a large-scale unified lexical semantic resource based on LMF. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL-2012), Avignon, pp. 580–590 (2012)
23. Hartig, O., Bizer, C., Freytag, J.C.: Executing SPARQL queries over the web of linked data. In: The Semantic Web – ISWC 2009, Heraklion, pp. 293–309 (2009)
24. Holtman, K., Mutz, A.: Transparent content negotiation in HTTP. Internet RFC 2295 (1998)
25. Ide, N., Pustejovsky, J.: What does interoperability mean, anyway? Toward an operational definition of interoperability. In: Proceedings of the 2nd International Conference on Global Interoperability for Language Resources (ICGL 2010), Hong Kong (2010)
26. Ide, N., Suderman, K.: GrAF: A graph-based format for linguistic annotations. In: Proceedings of the First Linguistic Annotation Workshop (LAW 2007), Prague, pp. 1–8 (2007)
27. Ide, N., Le Maitre, J., Véronis, J.: Outline of a model for lexical databases. In: Zampolli, A., Calzolari, N., Palmer, M.S. (eds.) Current Issues in Computational Linguistics: In Honour of Don Walker, Giardini, pp. 283–320 (1995)
28. Ide, N., Fellbaum, C., et al.: The manually annotated sub-corpus: a community resource for and by the people. In: Proceedings of the ACL 2010 Conference Short Papers, Uppsala, pp. 68–73 (2010)
29. Klyne, G., Carroll, J.J, McBride, B.: Resource description framework (RDF): concepts and abstract syntax. Technical report, W3C Recommendation (2004)
30. Marcus, M.P., Santorini, B., Marcinkiewicz, M.A.: Building a large annotated corpus of english: the penn treebank. Comput. Linguist. **19**(2), 313–330 (1994)
31. McCrae, J., Spohr, D., Cimiano, P.: Linking lexical resources and ontologies on the Semantic Web with Lemon. In: The Semantic Web: Research and Applications, Heraklion, pp. 245–259 (2011)
32. McCrae, J., Montiel-Ponsoda, E., Cimiano, P.: Collaborative semantic editing of linked data lexica. In: Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC-2012), Istanbul (2012a)
33. McCrae, J., Montiel-Ponsoda, E., Cimiano, P.: Integrating WordNet and wiktionary with lemon. In: Chiarcos, C., Nordhoff, S., Hellmann, S. (eds.) Linked Data in Linguistics, pp. 25–34, Springer, Heidelberg (2012b)
34. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)
35. Prud'Hommeaux, E., Seaborne, A.: SPARQL query language for RDF. W3C working draft (2008)
36. Quilitz, B., Leser, U.: Querying distributed RDF data sources with SPARQL. In: The Semantic Web: Research and Applications, pp. 524–538. Springer, Berlin/Heidelberg (2008)
37. Schenk, S., Petrák, J.: Sesame RDF repository extensions for remote querying. In: Proceedings of the 7th Znalosti Conference (Znalosti-2008), Bratislava (2008)
38. Shadbolt, N., Hall, W., Berners-Lee, T.: The semantic web revisited. IEEE Intell. Syst. **21**(3), 96–101 (2006)
39. Van Assem, M., Gangemi, A., Schreiber, G.: Conversion of WordNet to a standard RDF/OWL representation. In: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006), Genoa, pp. 237–242 (2006)
40. Véronis, J., Ide, N.: A feature-based model for lexical databases. In: Proceedings of the 14th International Conference on Computational Linguistics (COLING-1992), Nantes, pp. 588–594 (1992)
41. Windhouwer, M., Wright, S.E.: Linking to linguistic data categories in ISOcat. In: Chiarcos, C., Nordhoff, S., Hellmann, S. (eds.) Linked Data in Linguistics, pp. 99–107. Springer, Heidelberg (2012)

Chapter 3
Establishing Interoperability Between Linguistic and Terminological Ontologies

Wim Peters

Abstract Linguistic description is important for the re-use of lexical resources and the interpretation of text. Linguistic knowledge plays an important role in defining and enriching ontological knowledge. The fact that there are multiple proposed (de facto) standard models from the terminological, linguistic and localization fields creates a need for interoperability between linguistic models. These models complement each other or overlap to a certain extent, which creates linguistic confusion. This paper presents the LingNet model and its implementation, which enables interoperability between and comparison of different models, and the harmonization of linguistic description across application domains, allowing a user to a customized combination of elements from different models according to criteria of coverage, complementarity and granularity of linguistic description.

3.1 Introduction

From a practical point of view, linguistic and terminological standards are in daily use for the purpose of resource creation, such as term banks, dictionaries and translation memories. These different application areas are often unaware of cross-border standards and best practices.

With the development of semantic web applications that require interoperability between textual elements, linguistic/terminological resources and ontologies, the availability of relevant resources is of paramount importance. Applications increasingly depend on sharing and merging textual and lexical resources, and the time is ripe for putting mechanisms into place in order to make conceptual and linguistic classifications interoperable and exploitable in a uniform fashion.

W. Peters (✉)
Department of Computer Science, University of Sheffield, Sheffield, UK
e-mail: w.peters@dcs.shef.ac.uk

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_3,

Formally, the relationship between ontology and lexicon is an uneasy one. Depending on the level of formality of the involved ontologies, linguistic and ontological information seem to partially overlap and complement each other. In lightweight ontologies where most of the onus of semantic interpretation rests on the label of each conceptual node, there does not seem to be a distinction between ontological classes on the one hand, and lexical entries on the other.

In formal ontologies however, both types of information pertain to different domains of representation. The general consensus is that ontological and linguistic information is best separated, in order to allow for a fully flexible way to associate linguistic elements with ontology elements [5, 16]. Linguistic information is associated with concept labels, which are only considered to be evocative of the concept they are associated with. It is only the ontological structure that provides the formal semantics of the concept in intensional or extensional terms, whereas the concept name is no more than a string identifier. The ontology representation languages RDF and OWL are inherently not capable of capturing links between linguistic complexity and conceptual structuring within a single ontology. RDF[1] is the W3C recommended framework for making resource descriptions available on the web. Uniform identifiers (URIs) make it possible to link linguistic resources across multiple RDF graphs [12]. OWL[2] is one further step of formalization, with e.g. additional properties and cardinality restrictions.

This partitioning of the linguistic and conceptual semantic domain is formally expressed in the LMM (Linguistic Meta-Model) architecture[3] [18] that functions as an umbrella ontology for bridging a.o. lexical resources and ontologies. It provides a semiotic-cognitive representation of linguistic knowledge and grounds it in a formal semantics. LMM models the main semiotic notions by means of three classes: Reference, Meaning and Expression, formalizing the distinctions of the semiotic triangle [15], and providing a formal relationship between linguistic elements and ontological concepts by means of the object property "expresses", which links the linguistic domain with the conceptual domain (see Fig. 3.1). Similar triangular models have been proposed earlier, e.g. [14].

Syntactic lexicalization patterns and lexical semantics can suggest relations between existing ontology elements. A very simple example are modifier-headword relations in English compounding, where the linguistic structure of the compound "blue fin tuna" is indicative of its ontological modelling as a subclass of the concepts represented by the head word "tuna". Within the standard ontology formalisms, any such conceptually relevant linguistic information can only be harnessed by explicit reification strategies that use this information. Linguistic patterns underlying more refined ontological representation need to be re-engineered into ontological knowledge constructs after axiomatizing linguistic entities and their relations. For instance, the RDF triple BlueFinTuna rdfs:subClassOf Tuna can only

[1] http://en.wikipedia.org/wiki/Resource_Description_Framework

[2] http://en.wikipedia.org/wiki/Web_Ontology_Language

[3] http://www.ontologydesignpatterns.org/ont/lmm/LMM_L2.owl

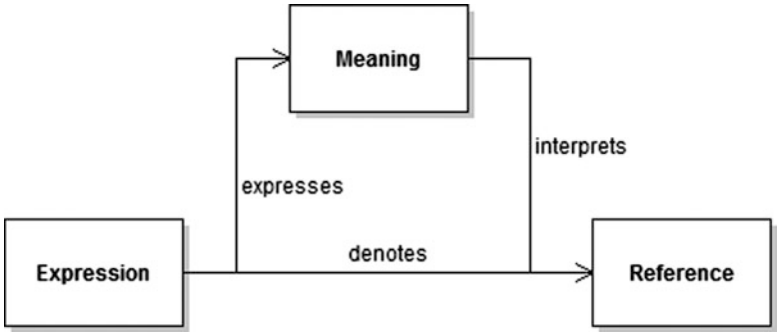

Fig. 3.1 The LMM semiotic triangle

be created in a linguistically motivated way after the identification of the head within the compound "blue fin tuna", and the subsequent re-engineering of this linguistic head-modifier construction into a subclass axiom. Another example is the formalization of verb arity structures as described in LexOnto [6] and the Lexical Markup Framework (LMF) [9], where predicates and predicate arguments are modeled as (event) classes with participants.

An important observation is that most presently available non-linguistic ontologies are lightweight, in the sense that their number of logical axioms is small, mostly restricted to subclass relations, and the naming of the concepts and the (lexically-based) semantic relations between them determines the understanding of the conceptual structure. The importance of linguistic knowledge is therefore much greater in lightweight ontology engineering.

The linguistic descriptive vocabulary that enables NLP-based reification strategies such as the ones described above should ideally be unique and universal. At present, however, there exist various standardized models and best practices, which complement each other or overlap to a certain extent. Moreover, many non-standard models with deviating terminology and coverage compound the linguistic confusion.

Therefore, given the existence of this variety of (standard) linguistic models, it is necessary to establish interoperability between their vocabularies in a principled way in order to enable interdisciplinary re-use and comparison. Given the limited, even if growing, number of linguistic models, this interoperability should be manually specified in a maximally exhaustive way, in order to enable full harmonization and assessment of the differences and similarities between models, and maximum exploitation of the now interoperable linguistic information.

3.2 Linguistic Knowledge

Overall, linguistic knowledge is expressed in various ways in terminological and linguistic resources. The nature and format of this knowledge is determined by a number of factors, such as user needs and the required level of adhesion to existing

standards for the representation of the linguistic knowledge. Individual linguistic and terminological resources largely differ in the explicit linguistic information they capture, which may vary in format, content granularity and motivation (linguistic theories, intended users etc.) [13].

There are many proposed standards and best practices for encoding linguistic and terminological knowledge, both from the (computational) linguistic and the semantic web side. They differ in representation format and level of formalization. Many linguistic resources, including text corpora and the TC37/SC4 data category registry [10] (containing a.o. the ISO16620 standard vocabulary) are encoded in XML, but an increasing number of linguistic resources are represented as populated RDF or Owl models.

Formalization of linguistic information is necessary in order to fully capture the partitioning between lexical and conceptual knowledge mentioned in Sect. 3.1. The overall purpose of all linguistic and terminological modelling ontologies is to associate multilingual linguistic knowledge with formal conceptual ontology elements. Many of these modelling ontologies have been engineered re-using elements from linguistic and terminological standards.

The advantage of RDF and Owl is that resources can be queried and linked in a uniform fashion with the Sparql[4] ontology query language, which allows the identification of linguistic phenomena that impact ontological categorization, and both the comparison and interoperability of linguistic descriptions. The many RDF/OWL ontologies for linguistic modelling that exist cover the whole spectrum of computational linguistic specification to ontology engineering. On the computational linguistic end the Lexical Markup Framework (LMF) [9] presents a linguistic description of lexical knowledge, whereas semantic web initiatives such as LingInfo [4], LexOnto [6], LexInfo [5] and the Linguistic Information Repository (LIR) [1, 16] capture various parts of the linguistic descriptive domain at the ontology engineering end. Lemon [11] is also a model for sharing lexical information on the semantic web, and draws on LexInfo, LIR and LMF. It aims to encompass all aspects of lexical encoding.

GOLD [8] is a richly axiomatized ontology for descriptive linguistics. It is intended to capture the knowledge of a well-trained linguist, and can thus be viewed as an attempt to codify the general knowledge of the field.

With respect to interoperability, The NLP Interchange Format (NIF[5]) is an RDF/OWL-based format that aims to achieve interoperability between Natural Language Processing (NLP) tools, language resources and annotations.

OLiA[6] represents a repository of reference categories for morphosyntax, syntax and is informally interlinked with ISOCAT and GOLD.

[4] http://www.w3.org/TR/rdf-sparql-query/

[5] http://nlp2rdf.org/nif-1-0

[6] http://nachhalt.sfb632.uni-potsdam.de/owl/

The present-day situation is that with so many (de facto) standard descriptive systems there is a problem with exhaustiveness: many do not cover all aspects of linguistic description to the highest possible level of granularity. For instance, LMF is quite underspecified in its definitions of class attributes. Most models are partially overlapping and/or complementary. This is not necessarily a drawback, because overlap indicates that there is consensus about the modelling of the linguistic domain. However, there is no mechanism yet to formally capture the commonalities and differences between descriptive systems. For instance, both LIR and LexInfo share the LMF-based component module to describe segmentation of complex entries. The orthographic information from LIR (that uses elements from ISO12620) complements LexInfo on the one hand, whereas on the other hand the syntactic subcategorization of LexInfo complements LIR. Interoperability can be obtained by defining mappings between these models that allows the detection of overlap, complementarity and navigation. Since the RDF/OWL level is the semantically most expressive level to express this interoperability, this means that all (non-)standard models should be re-engineered if they are not available in RDF/OWL already, and that all relevant ontologies should become networked.

3.3 Networking Linguistic Ontologies

Aligning or networking linguistic ontologies can happen in several ways. The first question we ask is which mapping principle we should adopt. One can envisage the creation of a standardized list of units with a maximum level of required granularity of linguistic description, which functions as an "interlingua" [17], an approach adopted by e.g. OLiA. An important requirement is that the "interlingual" standard representation needs to be descriptively exhaustive and more granular in its linguistic description than any other element from other vocabularies, in order to fully and flexibly capture the linguistic conceptualizations expressed by these elements. The models whose elements are mapped to interlingual elements provide local dependencies between the "interlingual" data categories. ISOCAT's data category registry is aiming for this goal. However, the ISOCAT list of linguistic and terminological data categories is still under development, and it has not yet been fully established. Moreover, the issue of the formalization of conceptual overlap between standard vocabularies adopted by ISOCAT has not yet been resolved.

Another modelling approach (the one advocated in this article) does not presume that the linguistic domain is uniformly modeled by the available descriptive (de facto) standard systems. We cannot assume that all model concepts can be exhaustively represented in an "interlingua". Different models should be allowed to partition the domain of linguistic description in different ways in order to allow inconsistency, incoherence and disparate modelling. Therefore, no particular model should in principle function as an interlingua, and interoperability is achieved by means of distributed pair-wise mappings between elements from different models. Pair-wise mapping according to a formal mapping model will allow a

detailed comparison of the ways in which the models differ and overlap, and a formal description of this comparison. Moreover, from these pair-wise mappings theoretical consensus and best practice for linguistic modelling may possibly emerge.

The second question we ask ourselves is what mapping vocabulary we should use. We discuss two options:

1. The adoption or definition of a set of mapping relations in the form of object properties, for instance the SKOS[7] [2] mapping relations broadMatch, closeMatch, exactMatch, narrowMatch and relatedMatch. A disadvantage of adopting SKOS is that this vocabulary only comprises a rather coarse-grained set of mappings. Another disadvantage is that SKOS mappings cannot cover links between different types of ontology elements, e.g. classes and properties.
2. The full reification of mapping relations within a mapping model. This option promotes mappings to first class citizens (i.e. models them as classes rather than properties). The advantage of this reification of mapping relations into classes is that it allows us to describe them as ontological objects and model the relations in a fine-grained and extendable fashion at the cost of a higher level of complexity.

We have opted for approach 2, which has greater descriptive power than approach 1, because it can link elements of any type. For instance, it can describe an equivalence relation between a property and a class (for example hasPartOfSpeech and PartOfSpeech), which is impossible to do using SKOS. In contrast to approach 1, approach 2 does not presume that the linguistic domain is uniformly modeled by the available descriptive systems. Different models are allowed to constitute different partitioning options, in order to allow inconsistency, incoherence and disparate modelling. Therefore, no particular model should function as an interlingua acting as the hub for interoperability. This should rather be achieved by means of distributed pair-wise mappings. Distributed mapping avoids localized inclusion of whole ontologies, and favours a more modular approach. Conceptually coherent building blocks can be identified, modularized, and imported in order to cover the exact enrichment requirements of the user, enabling a pick-and-mix combination of elements from models according to criteria of coverage, complementarity and granularity of linguistic description.

This network architecture changes the way in which existing ontology elements are re-used. It no longer requires the traditional OWL/RDF import operation, which includes the whole ontology rather than its desired target elements. Also, it can accommodate the generally adopted practice of defining new concepts that are equivalent to existing standards. LingNet's distributed mapping avoids localized inclusion of whole ontologies, and favours a more modular approach. Conceptually coherent building blocks can be identified, modularized, and imported in order to cover the exact enrichment requirements of the user, enabling a pick-and-mix combination of elements from models according to criteria

[7] http://www.w3.org/2004/02/skos/core

of coverage, complementarity and granularity of linguistic description. This is the methodology we adopted while creating our model for the mapping of linguistic knowledge, LingNet, which is described in Sect. 3.5.

3.4 Related Work

The issues of modelling the relation between lexicon and ontology, and the formalization and interoperability of linguistic and terminological information have led to a number of standardization and collaboration initiatives.

The SKOS core vocabulary, a standard from the W3C SKOS community (see also previous section), is a lightweight OWL ontology created to facilitate web-oriented taxonomies and thesauri. In order to model the relation between a lexical element and a concept (the "expresses" relation in Sect. 3.1, SKOS lacks the expressiveness to fully describe the labels of concepts because it does not encode a concept label as a concept but as a datatype property skos:label, which is equivalent to xml:lang.

To address this limitation, W3C proposes the SKOS eXtension for Labels (SKOS-XL) [20] for describing lexical entities. The skosxl:Label concept that is part of this extended vocabulary can now be used to model the relation between lexicon and ontology by means of a predicate like "skosxl:Label lmm:expresses owl:Thing".

As mentioned before, SKOS also provides a mapping vocabulary to capture semantic relations between ontology elements. It identifies a number of semantic relations, in particular broadMatch, closeMatch, exactMatch, narrowMatch and relatedMatch.

Along the same lines RELCAT [20] defines a number of relations in order to accommodate the linking of local/personal linguistic data categories to elements from the ISOCAT registries. These relations more or less mirror the SKOS relations.

As stated in Sect. 3.3, since these mapping relations only provide coarse-grained mappings, and do not offer the possibility to further characterize additional aspects of the semantic relation holding between linguistic descriptive elements, the definition and formalization of a maximally descriptive mapping between vocabularies and formalisms will remain a crucial factor in the work of the various ongoing initiatives addressing different communities. Whereas SKOS is widely used in the semantic web community, RELCAT originates from (computational) linguistic community initiatives such as CLARIN and META-NET. CLARIN[8] is committed to establish an integrated and interoperable research infrastructure of language resources and its technology. It aims at offering a stable, persistent, accessible and extendable eHumanities infrastructure. META-NET[9] is a Network

[8] http://www.clarin.eu

[9] http://www.meta-net.eu/

of Excellence serving the multilingual European information society by establishing interoperability between language technology and resources.

A recently established W3C Ontolex interest group[10] develops models for lexicons and their relation to ontologies, and investigates the added value of using such models in semantic web NLP applications.

The Open Linguistics Working Group of the Open Knowledge Foundation[11] works towards a linked open data cloud of linguistic resources, which applies the linked data paradigm to linguistic knowledge.

3.5 LingNet

The LingNet model is a more complex model than a set of binary semantic relations in the form of e.g. SKOS semantic relations. It promotes mappings to classes rather than properties, which enables us to add further structure to each mapping. The novelty of the LingNet model does not lie in its modelling method, but in its combination of selected mapping methods and its application to the linguistic/terminological domain.

3.5.1 The LingNet Model

The basic structure of LingNet is as follows:

- Each mapping between a source and target ontology has one or more mapping assertions that describe a semantic relation between a source ontology class and a target ontology class.
- Mappings are first-class objects that exist independently of the ontologies.
- Mappings are directed and there can be more than one mapping between two ontologies.

This mapping structure is based on the ontology mapping metamodel as described by Brockmans et al. [3] (see Fig. 3.2). The advantage of this metamodel for linguistic interoperability is that it is formalism-independent. The mapping metamodel takes a number of different kinds of semantic relations that have been proposed in the literature into account. Most common are the following kinds of semantic relations:

Equivalence states equality of the connected elements represent the same aspect of the real world according to some equivalence criteria.

[10] http://www.w3.org/community/ontolex/

[11] http://okfn.org/

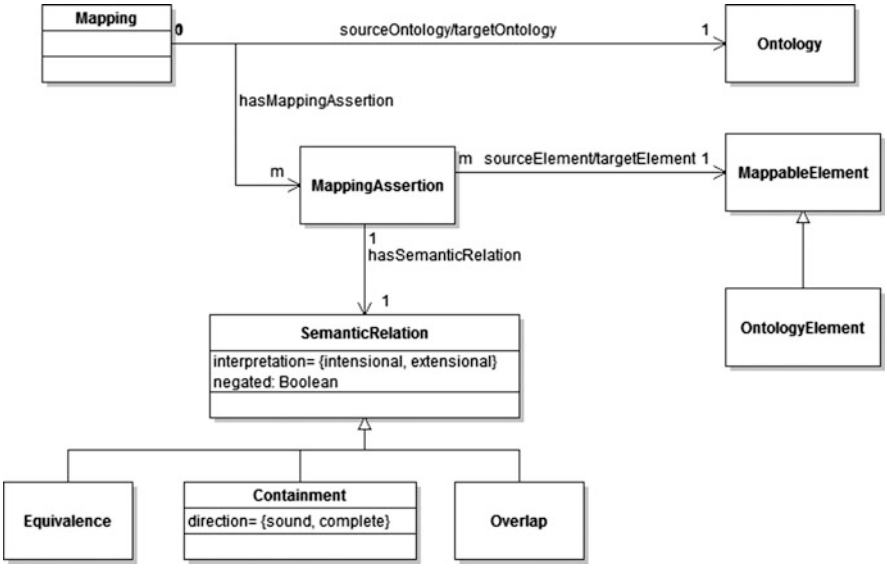

Fig. 3.2 The LingNet metamodel

Containment states that the element in the source ontology represents a more specific aspect of the world than the element in the target ontology. Depending on which of the elements is more specific, the containment relation is defined in the one or in the other direction.

Overlap states that the connected elements represent different aspects of the world, but have an overlap in some respect. In particular, it states that some objects described by the element in the one ontology may also be described by the connected element in the other ontology.

The mappings are knowledge-based, i.e. they do not require language-specific constructs for mappings. The reification of mapping relations into classes allows us to describe them as ontological objects and model the relations in a detailed and extendable fashion.

LingNet (see Fig. 3.3) provides a more specific subcategorization than the mapping metamodel. It has the following subclasses of the Containment concept as extensions:

TaxonomicalContainment
  Hypernymy
MeronymicalContainment
  Part
  Member

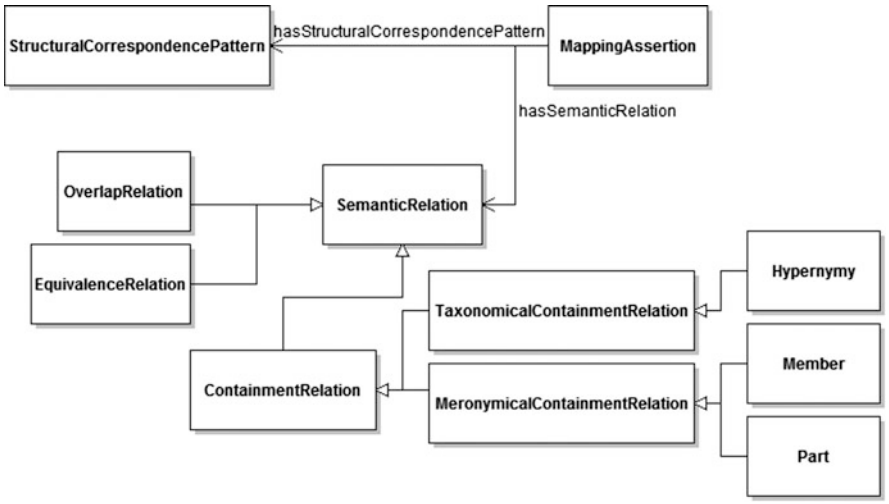

Fig. 3.3 The LingNet model extension

These classes make the specifics of the Containment relation (set inclusion) explicit by means of a distinction between taxonomic and meronymic containment, which are both important for capturing lexical semantic relations. This is to be regarded as an incremental extension, which is by no means exhaustive.

An additional property of LingNet is that it references structural differences between ontologies according to a list of structural alignment types. Even if ontologies share conceptually equivalent elements, they often express their content in different ways, because they differ from each other in structural terms. For instance, the LexInfo concept Noun is equivalent to the LMF concept LexicalEntry, with 'noun' as the value of its partOfSpeech attribute. This is a typical example of a class to class-plus-attribute transformation, which is one of a series of structural transformations observed and collected by Scharffe et al. [19], which regulate regularly observed structural transformations between different configurations, and are referenced by LingNet.

3.5.2 LingNet Implementation

In order to initiate the population of the LingNet model, we have mapped classes of six models covering terminology, linguistics, translation memories and semiotics. The binary mappings involve 72 ontology elements and 55 mapping assertions. These constitute the first version of the populated model.[12]

[12] http://www.gate.ac.uk/ns/ontologies/LingNet/LingNet-v0.1.owl

Figure 3.3 illustrates the modular architecture of LingNet, in which the standard models are aligned by a mapping mechanism (consisting of an extended mapping metamodel and structural alignment patterns), and connected to ontology elements through a lexical-ontological interface.

The models that have been included so far in LingNet are the following:

- LIR[13]: Linguistic Information Repository
- TMX[14]: Translation Memory eXchange
- XLIFF[15]: XML Localization Interchange File Format
- MLIF[16]: Multi Lingual Information Framework
- LMF[17]: Lexical Markup Framework: An ISO metamodel for describing computational lexicons.
- LexInfo[18]: aligns the LingInfo and LexOnto ontologies with LMF.
- LMM[19]: Linguistic Meta-Model for the semiotic grounding of linguistic expressions (see Sect. 3.1).

TMX, XLIFF and MLIF all concern the standard encoding of translation memories. TMX and XLIFF are widely used, whereas MLIF is a proposed standard awaiting ISO conformation.

The LIR [1, 16, 17] represents a core set of linguistic information units that cover a range of linguistic phenomena covering mostly multilinguality, orthography and morpho-syntax. It has incorporated a selection of data categories from existing standard representations for linguistic and terminological resource description such as the following, in particular ISO 16642:2003,[20] Computer applications in terminology – TMF[21] (Terminological Markup Framework).

Because we base our work on the earlier mentioned LIR model, LingNet's present binary mappings covered until now are between classes from the LIR and the other models. As a core set of linguistic and terminological descriptors to be used for the description of ontology concept labels, LIR only contains a subset of all standard data categories. This limitation means that it lacks the ability to cover the whole range of linguistic/terminological description. A full size set of pair-wise mappings between all models isforeseen, and the inclusion of new (de facto) standard is

[13] https://gate.ac.uk/ns/ontologies/LingNet/lir.owl

[14] http://en.wikipedia.org/wiki/Translation_Memory_eXchange

[15] http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.pdf

[16] http://mlif.loria.fr/

[17] http://www.lexicalmarkupframework.org/

[18] http://lexinfo.net/lexinfo

[19] http://www.ontologydesignpatterns.org/ont/lmm/LMM_L2.owl

[20] http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=32347

[21] http://www.loria.fr/projets/TMF/

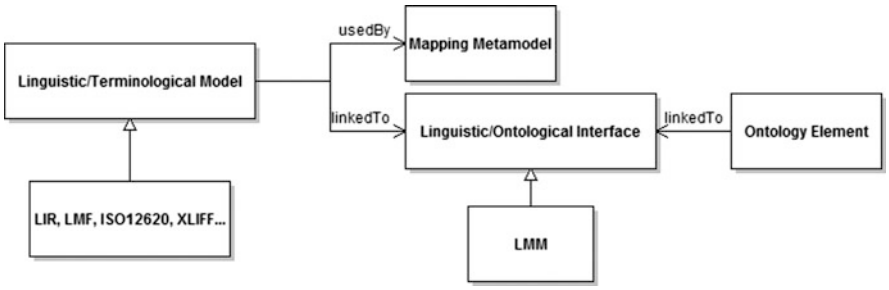

Fig. 3.4 The LingNet architecture

foreseen in the near future, such as the GOLD, Lemon and NIF ontologies, and ISO12620[22] for terminology.

The ontological models of the considered standards are partly available on the web, as in the case of LMF, LexInfo, LMM and LIR. TMF, XLIFF and MLIF have been manually re-engineered by the author in an ad-hoc fashion. The individual ontologies are available from: http://gate.ac.uk/ns/ontologies/LingNet/mapped-ontologies/.

Figure 3.4 illustrates the modelling architecture for LingNet. Linguistic and terminological models are mapped onto each other by means of a mapping mechanism that takes into account knowledge-based and structure-based mappings. Lexical entries from these models are linked through LMM to ontology concepts as their linguistic realizations.

3.6 Discussion

In its present form, LingNet performs the same as SKOS in that it covers the main semantic relations between ontology elements. The main difference is its extendibility because of the fact that the relation has been reified. This enables the incremental refinement of the semantic relation. Especially for e.g. overlap relations, a richer vocabulary is needed to capture the intricacies of this rather general relation. When two ontology elements are related through an OverlapRelation, one can extend this class with a refinement module to capture the difference between the two mapped concepts in term of the information contained in the definitions that are associated with them. The source and target concepts can differ according to their definition genus or differentia, as illustrated in Fig. 3.5 below. Further refinements can capture other aspects of the exact nature of the differences.

[22] http://www.ttt.org/clsframe/datcats.html

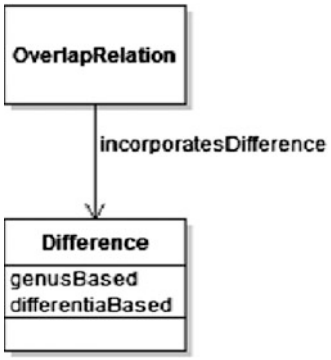

Fig. 3.5 Refinement module for OverlapRelation

LingNet offers the potential of a principled re-use of linguistic information. The interoperability of the standard models it establishes, allows a flexible choice of standard modelling for any particular resource, and the potential for conversion of the resource model into any networked standard format. Moreover, it enables the comparison of different standards and their evaluation with respect to coverage and descriptive adequacy, and the collaborative exploration of their commonalities and harmonization. This provides a solid base for future standardization activities.

The LingNet metamodel can, in principle, be used to semantically align any ontologies, but LingNet itself is specifically geared towards linguistic description. At this moment, LingNet covers a number of models, which collaboratively model a wide spectrum of linguistic phenomena. It is flexible, in that it allows the additional embedding of any other model, which will incrementally corroborate and expand its coverage. The mapping relations provide the interoperability without enforcing full consistency between all models.

The LingNet model is extensible in that it allows the inclusion of e.g. additional mapping relations and modules for more fine-grained comparison of concepts across models.

The populated LingNet model constitutes a first step towards a full network in which ontology elements from all networked linguistic/terminological ontologies are connected to each other. At the moment the LIR functions as hub, i.e. relations are defined between the LIR and other vocabularies, and therefore the coverage of the network is restricted to the areas of linguistic description covered by LIR (orthography, morphosyntax, semantics, translation), which are covered to varying degrees by the different NLP application areas of linguistics, terminology and translation (see Fig. 3.6). At present, the standard models are indirectly aligned in LingNet through the LIR. It is our intention to extend the alignments to all ontology pairs in order to achieve full interoperability. Given the limited number of standard models pair-wise alignment is a feasible task. Moreover, we will investigate the integration of mapping alternatives other than the pair-wise approach in order to conflate binary mappings into e.g. LMF's mapping axes.

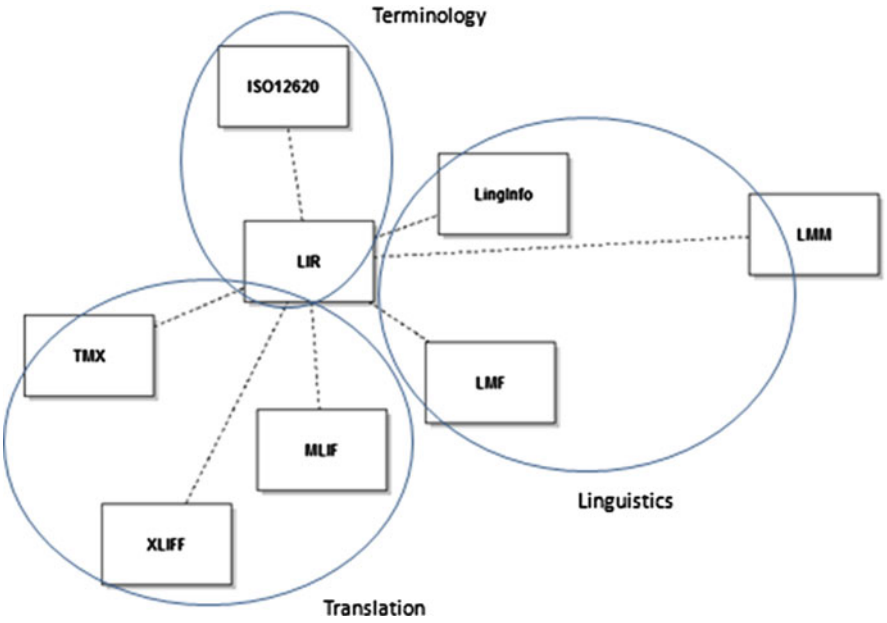

Fig. 3.6 LingNet's linguistic and terminological domain coverage

3.7 Conclusion and Future Work

In this chapter, we have described LingNet, a model for the detailed mapping of linguistic and terminological descriptive vocabularies and structures. The LingNet model adopts a number of modelling decisions from the literature: a knowledge-based, formalism-independent metamodel for capturing semantic alignments between ontologies for linguistic/terminological description [3], references to external mapping patterns for structural mappings [19], and the integration of a lexicalization relation for linking linguistic/terminological resources to ontological concepts [18].

Future work will focus on the inclusion of more (proposed) standards for linguistic modelling such as Lemon and GOLD. Because the alignments need to be correct and exhaustive, and the number of (de facto) standards is relatively small, this will be a continuation of the manual work described in this paper.

Furthermore, we intend to establish a full integration of structural alignments, which have for now only been referenced in LingNet. The exhaustive identification of ontology elements involved in the structural alignments and their full implementation, as modeled by Scharffe et al. [19], will complement LingNet's concept-based alignments.

Finally, we will look into linking the LingNet model with specific existing representation formalisms by, for instance, establishing an automatic conversion into the alignment server format [7].

Acknowledgements This work was partly funded by the NeOn project (IST-2004-2.4.7, http://www.neon-project.org).

References

1. Aguado de Cea, G., Gómez-Pérez, A., Montiel-Ponsoda, E., Peters, W.: Enriching ontologies with multilingual information. Nat. Lang. Eng. Camb. **17**, 283–309 (2010)
2. Bechhofer, S., Miles, A.: SKOS simple knowledge organization system reference. W3C recommendation, W3C. http://www.w3.org/TR/2009/REC-skos-reference-20090818/ (2009)
3. Brockmans, S., Haase, P., Stuckenschmidt, H.: Formalism-independent specification of ontology mappings – a metamodeling approach. In: Meersman, R., Tari, Z., et al. (eds.) OTM 2006 Conferences, Springer, Montpellier (2006)
4. Buitelaar, P., Sintek, M., Kiesel, M.: A lexicon model for multilingual/multimedia ontologies. In: Proceedings of ESWC06 (2006)
5. Buitelaar, P., Cimiano, P., Haase, P., Sintek, M.: Towards linguistically grounded ontologies. In: The Semantic Web: Research and Applications, 6th European Semantic Web Conference, Proceedings of ESWC 2009, Heraklion, vol. 5554. Springer, Berlin/Heidelberg (2009)
6. Cimiano, P., Haase, P., Herold, M., Mantel, M., Buitelaar, P.: Lexonto: a model for ontology lexicons for ontology-based NLP. In: Proceedings of the ISWC07 OntoLex Workshop, Busan (2007)
7. Euzenat, J.: An API for ontology alignment. In: Proceedings of the 3rd ISWC, Hiroshima, pp. 698–712 (2004)
8. Farrar, S., Langendoen, T.: A linguistic ontology for the semantic web. GLOT Int. **7**(3), 97–100 (2003)
9. Francopoulo, G., George, M., Calzolari, N., Monachini, M., Bel, N., Pet, M., Soria, C. LMF for multilingual, specialized lexicons. In: LREC, Genova (2006)
10. Kemps-Snijders, M., Windhouwer, M.E., Wittenburg, P., Wright, S.E.: ISOcat: a revised ISO TC 37 data category registry. In: Presentation at the Conference on Terminology and Information Interoperability – Management of Knowledge and Content (TII 2008), Moscow (2008)
11. McCrae J., Spohr D., Cimiano P.: Linking lexical resources and ontologies on the semantic web with lemon. In: Proceedings of the 8th Extended Semantic Web Conference (ESWC 2011), Heraklion (2011)
12. Miller, E., Manola, F.: RDF Primer, W3C recommendation, W3C. http://www.w3.org/TR/2004/REC-rdf-primer-20040210/ (2009)
13. Montiel-Ponsoda, E., Aguado de Cea, G., Gómez-Pérez, A., Peters, W.: Modelling multilinguality in ontologies. In: Coling Companion Volume – Posters and Demonstrations, Manchester (2008)
14. Ogden, C.K., Richards, I.A.: The Meaning of Meaning: A Study of the Influence of Language upon Thought and of the Science of Symbolism. Routledge & Kegan Paul, London (1923)
15. Peirce, C.S.: Collected Papers of Charles Sanders Peirce. MIT Press, Cambridge (1958)
16. Peters, W., Montiel-Ponsoda, E., Aguado de Cea, G.: Localizing Ontologies in OWL. In: Proceedings of the ISWC07 OntoLex Workshop, Busan (2007)
17. Peters, W., Montiel-Ponsoda, E., Aguado de Cea, G., Espinoza, M., Gómez-Pérez, A., Sini, M.: Multilinguality and localization support for ontologies, NeOn project deliverable D2.4.2. http://www.neon-project.org/nw/Deliverables (2008)

18. Picca, D., Gangemi, A., Gliozzo, A.: LMM: an OWL metamodel to represent heterogeneous lexical knowledge. In: Proceedings of the of the International Conference on Language Resources and Evaluation (LREC), Marrakech (2008)
19. Scharffe, F. Euzenat, J., Fensel, D.: Towards design patterns for ontology alignment. In: Wainwright, R.L., Haddad, H. (eds.) Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), Fortaleza, pp. 2321–2325 (2008)
20. Schuurman, I., Windhouwer, M.A.: Explicit semantics for enriched documents. What do ISOcat, RELcat and SCHEMAcat have to offer? In: Proceedings of the 2nd Supporting Digital Humanities conference (SDH 2011). Copenhagen, 17–18 Nov 2011

Chapter 4
On the Role of Senses in the Ontology-Lexicon

Philipp Cimiano, John McCrae, Paul Buitelaar, and Elena Montiel-Ponsoda

Abstract This chapter investigates the notion of 'sense' in the ontology-lexicon interface. As a realization of the ontology-lexicon interface, we are concerned with so called 'ontology lexica' which specify the meaning of lexical entries by reference to a given ontology. We propose that in the context of the ontology-lexicon interface a 'sense' can be understood as a three-faceted entity, i.e. as a (i) reification of the link between a lexical entry and the ontological reference, (ii) as subset of all the uses of the word that can be interpreted as referring to the same ontological reference, and (iii) as an implicitly defined subconcept. We also provide a new definition of the traditional notions of homonymy, synonymy, metonymy etc. in the ontology-lexicon interface.

4.1 Introduction

Ontology-based natural language processing (NLP) applications interpret language with respect to the vocabulary of a given (domain) ontology. Take the example of an ontology-based question answering system [26] and the following input question: "Who painted the Mona Lisa?". A query in SPARQL that represents the semantics of this question with respect to the DBPedia ontology [2] is given in Fig. 4.1.

P. Cimiano (✉) · J. McCrae
Semantic Computing Group, CITEC, Universität Bielefeld, Bielefeld, Germany
e-mail: cimiano@cit-ec.uni-bielefeld.de; jmccrae@cit-ec.uni-bielefeld.de

P. Buitelaar
Unit for Natural Language Processing, DERI, National University of Ireland, Galway, Ireland
e-mail: paul.buitelaar@deri.org

E. Montiel-Ponsoda
Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain
e-mail: emontiel@delicias.dia.fi.upm.es

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_4,

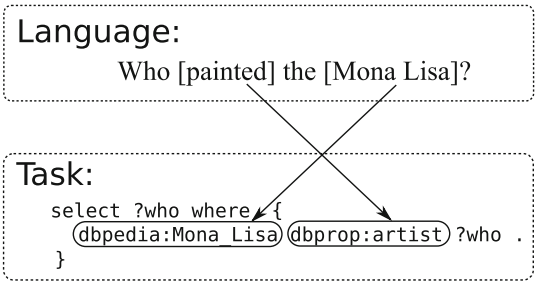

Fig. 4.1 Example of mapping of natural language into a task vocabulary

The interpretation of linguistic input is a compositional process and requires knowledge about how the lexical atoms – words or phrases – which occur in a given domain are interpreted in the context of a given ontology. An important assumption that we build on in this chapter is that the meaning of a word cannot be specified universally, i.e. independently of any application or domain, but that the meaning of a lexical entry is specific for the vocabulary defined by a given ontology. We refer to this principle as "semantics by reference".

The principle of 'semantics by reference' implies that the expressivity and the granularity at which the meaning of words can be expressed depends on the meaning distinctions made in the ontology. Consequently, there might be aspects of the meaning of lexical entries that can simply not be represented with respect to the semantic vocabulary of a given ontology ([6, 7]). We will give two examples for this: a monolingual and a cross-lingual one. Consider the words 'mosque' and 'synagogue'. In an ontology on urban planning, there might exist no specific concepts to represent the meaning of these words other than the more general concept `ReligiousBuilding`. Thus, with respect to this ontology, both 'mosque' and 'synagogue' will be interpreted as referring to the concept `ReligiousBuilding`. Clearly, this does not capture the 'full lexical meaning'[1] of these words, even though for the application and domain ontology in question the differences between a mosque and a synagogue might be irrelevant.

Let us now consider a cross-lingual example. Consider a geographic ontology which includes the concept of a `Watercourse`. With respect to this ontology, the full lexical meaning of the French words 'rivière' and 'fleuve' cannot be represented, as they encompass differing aspects that are not captured in this ontology. An ontology in our sense can essentially be seen as an artifact that represents a particular conceptualization of such a domain, limiting the representation of word meaning to those distinctions that are actually relevant in the context of the given ontology and/or domain. As such in the context of ontology-based NLP, it follows that the meaning of words is highly specific for a given ontology and that it should

[1] By full lexical meaning we refer to the meaning that an average speaker of a language who shares common knowledge with his/her community associates with a word in their mental lexicon with respect to some language of thought.

be possible to make principled choices concerning the number and granularity of senses that a word has as these senses may be said to correspond directly to explicit ontological distinctions made in the ontology.

The ontology-dependent meaning of a lexical entry (word or phrase) is captured in what we call an *ontology-lexicon*. A key question is how *senses* in an ontology-lexicon differ from those employed in traditional lexical resources such as WordNet. In fact, we could ask if explicit senses are needed at all in the context of the ontology-lexicon interface where ontology elements (classes, relations, individuals) essentially capture the meaning of words. We argue, however, that in the context of an ontology-lexicon, the role of a sense is to reify the link between a lexical entry and the concept it *evokes*, at the same time providing a hook into the ontology, thus allowing the larger context of the evoked concept to be exploited for interpretation. In addition, this reified link allows to model properties, including register as well as pragmatic constraints and implications related to the usage of the given word as evoking the concept in question.

Thus, we argue that in the context of the ontology-lexicon interface, a *sense* can be understood as a three-faceted entity. On the one hand, a sense can be understood as a *reification of a pair of lexical entry and its corresponding reference in the ontology*, i.e. the evoked concept so to speak. This is useful to state conditions under which it is possible to interpret the word as referring to that concept. Secondly, a sense can be regarded as *a subset of all the uses of the given lexical entry that refer to the concept in question*. Thirdly, a sense represents also a *hypothetical concept* that, if added to the ontology, would be a subclass of the evoked concept. This hypothetical concept accounts for the full lexical meaning of the word in question, but neither exists explicitly in the lexicon nor the ontology.[2]

The inclusion of explicit senses in the lexicon – as reifications of lexical entry/concept pairs – does not imply that the meanings of a word are fixed. In fact, through the interplay between the ontology as background knowledge and the given linguistic context (i.e. a specific sentence in which the word in question appears), further aspects of the meaning of a word can be brought into the foreground by a process that produces a semantic interpretation of the sentence. Lexical meanings in the ontology-lexicon can therefore be generated upon need, given the constraints of lexical context and semantic scope of the ontology. In this sense, an ontology thus supports a generative process in the sense of Wierzbicka [27] and Pustejovsky [25] by which further aspects of the meaning of a word can be derived from the ontology. Yet, these additional meaning aspects need not form part of the semantics of the word in the narrow sense, but are part of the larger ontological context and can be 'recruited' to support the interpretation of the word or phrase in a particular sentence. In line with this, our analysis allows us to provide a new account of the

[2] As our reviewer has pointed out, the hypothetical concept thus needs to exist in some ontology. In fact, it does, but not in the actual domain ontology, but rather in our ontology of the lexicon-ontology interface.

phenomenon of polysemy and metonymy in the context of the ontology-lexicon interface.

In this chapter, we will elaborate on these issues, providing a theoretical account of the role of senses in the ontology-lexicon interface. We will further provide an analysis compatible with the principle of 'semantics by reference' of aspects of the interpretation of words that have been traditionally subsumed under the phenomenon of polysemy. We argue that such aspects can be accounted for by a process that brings into the foreground further aspects of the meaning of words as a byproduct of the interpretation of a sentence given a specific linguistic context and the ontology as background knowledge. We will also briefly present the *lemon* model for representing an ontology-lexicon ([11,21,22]) and discuss how the above theoretical considerations have influenced the design of this model.

4.2 Senses: Universal or Context-Specific?

Traditionally, senses are regarded as specifying the various meanings of a word. Approaches differ essentially in whether they assume a finite and fixed amount of senses or postulate an open and highly context-specific set of senses. The specification of the set of senses (interpretations, meanings) for a given lexical entry is a central task in lexicography, involving decisions on whether to 'lump' potential senses together or to 'split' them into individual senses [15]. In practice, this ends up being a very subjective task in which lexicographers are guided by factors such as the purpose of the lexicographic resource, its envisioned users, the frequency of use of a certain meaning, or its predictability from other senses [19] and as such it has been questioned whether this is useful for NLP applications [17]. This view seems to be validated by the task of Word Sense Disambiguation (WSD) where it seems that many sense distinctions are not natural even for humans, as inter-annotator agreement for WSD seems to have a limit of about 80 % [13]. A traditional approach to defining senses is by cross-lingual comparison [12]. The distinction between 'paper' (as a material) vs. a (news)-paper (as an information container) is for example inherent in many languages[3] and one could thus argue that these two senses are language-independent or 'natural' (if not to say universal). Furthermore, it has been shown in the context of machine translation that this approach is helpful and outperforms approaches based on fixed catalogues of senses such as found in WordNet [8]. However, cross-lingual differences are not a solid basis to identify different word senses. Take the example of 'computer' with its two senses 'a machine for performing calculations automatically' and 'an expert at calculation (or at operating calculating machines)'.[4] By cross-lingual comparison,

[3] For example German "Papier" and "Zeitung", French "papier" and "journal" and Japanese "kami" and "shimbun."

[4] These are the glosses of the two corresponding senses from WordNet 3.1.

we might identify these two different senses if in some language two different words are available for each of these two senses. If we compare German and English, however, we will find out that both languages use the same word for these two senses, i.e. 'computer' in English and 'Rechner' in German. Furthermore, it is often the case that languages make distinctions that are not considered fundamental to the native speaker of a language. German distinguishes for instance different types of 'going', using 'gehen' in the case of moving under one's own power and 'fahren' when using mechanical assistance (e.g. a bicycle). This distinction seems unnatural to a native English speaker.[5] Following the tradition of linguistic relativism, it has been argued that concepts are language-specific, counterfeiting the assumption that cross-linguistic comparison can help us to establish a universal set of senses. It thus seems that relying on cross-lingual commonalities and differences as a basis to build a catalogue of senses will lead to extreme fragmentation and to overly specific senses that are not relevant in the context of a given application.

It has been indeed argued by some researchers that a small set of senses per word might suffice for practical applications. Ide and Wilks [18] for instance propose that the meaning of 'paper' can be captured with only three senses: (i) as a material, (ii) as a written article or document and (iii) as a newspaper. They argue that other senses such as 'publisher of the newspaper' identified in some dictionaries are unnecessary and can in fact be derived from background knowledge (see Sect. 4.4 on this). While distinguishing such core senses of a word might be enough for general purposes, for certain domains we might need a much more fine-granular and domain-specific set of senses. Leon Araúz et al. [20] have for example argued that in their application, they need to distinguish three different senses of 'accretion', as (i) accretion of snow flakes in the atmosphere, (ii) the accretion of ground in a tectonic plate and (iii) the accretion of sand in the formation of coastal bars. While these three senses have a similar basic meaning, i.e. the one of accumulation of materials, it is necessary in this domain to distinguish them. However, such fine-grained distinctions will certainly not be included in a domain-independent lexicon such as WordNet, for reasons that should be clear.

At the other extreme are approaches that claim that any approach postulating a finite set of senses is unsatisfactory from a theoretical point of view. This view is connected to the assumption that there are as many senses as there are different contexts in which a specific word is used and is rooted in and supported by insights from philosophy and linguistic study. Wittgenstein famously claims that "for a large class of cases – though not for all – ...the meaning of a word is its use in the language" [29]. Cruse states that the meaning of a word form is different in every distinct context in which it occurs [9]. Cruse and Croft even maintain that word senses are created *at the moment of use*, in what they consider a dynamic approach [10].

While theoretically appealing, approaches which assume an infinite inventory of senses – one for each usage context – are less useful from an NLP point of view

[5] At least from the opinion of the native English-speaking author of this chapter.

as automatic processing requires an inventory of senses that generalizes beyond specific examples and contexts observed. From an NLP perspective, it is crucial to have (i) either a finite set of senses, or (ii) a specification of the core meaning of a word together with a set of generative processes that allow to derive new meanings from this core. The latter is essentially the underlying idea of the *Generative Lexicon* of Pustejovsky [25]. According to Pustejovsky, lexical items have a semantic representation of their conventionally assumed meaning, which is accessed in language understanding and production and can produce context-specific interpretations (senses) due to certain constraints that activate one sense or the other. An essential aspect of this theory is that lexical meaning is not decomposed into individual senses, but instead that different context-specific interpretations (roughly corresponding to senses) are activated on demand out of an underspecified lexical semantic representation that Pustejovsky refers to as *Qualia Structure*.

Our standpoint is that there is no universal set of senses for a word that will suit all purposes and applications. In some applications, it might suffice to interpret both 'synagogue' and 'mosque' as `ReligiousBuilding`. Other domains might require very specific senses as in the case of 'accretion' discussed above. In other domains, some of the senses that are generally distinguished might not be relevant. For instance, in the domain of scientific publishing, the material sense of 'paper' might not be relevant. Overall, it is thus clear that there is no set of universal senses that are valid independently of the domain and application. It is thus legitimate to assume that the senses of a word are specific for a given ontology that models the domain in question.

4.3 Senses in the Ontology-Lexicon Interface

As stated in the introduction, we regard a sense as a three-faceted entity with three roles. We elaborate on these roles in this section. In the following, we will assume that there is a given ontology $O = (\Lambda_O, V_O)$ expressed in logical language Λ_O and vocabulary V_O consisting of a set of concepts C_O, a set of relations R_O and a set of individuals I_O as well as a lexicon L.

In this chapter we focus on tasks where natural language needs to be interpreted with respect to a given task and ontology such as the question answering task illustrated in Fig. 4.1. Thus, we consider that for a given task we need to find a mapping to an ontology-based representation, which we consider to be a formula in some task language $\mathscr{T}$ which uses the symbols of the vocabulary and the language of the ontology as well as some additional task-specific symbols. The interpretation of natural language with respect to task language $\mathscr{T}$ is given by the following function $[\cdot]_T$[6]:

[6]Without loss of generality we simplify to the case where there is only a single result for the task.

$$[\cdot]_T : \mathscr{L} \rightarrow \mathscr{T}$$

where $\mathscr{L}$ is the natural language in question.

4.3.1 Senses as Reification

We denote a *sense* as $\sigma^{(l,c)}$, where $l \in L$ is a lexical entry and $c \in V_O$ is the ontological concept or *reference*, and we define the *ontologically interpretable words* in the sentence λ as $W_\lambda \subseteq \{l_i : l \text{ is the } i^{th} \text{ word of } \lambda\}$. Furthermore, we define the *meaning of a word in sentence* λ as a function

$$[\cdot]_T^\lambda : W_\lambda \rightarrow V_O$$

And we assume that this function satisfies the *compositionality principle* given as:

$$\forall l_i \in W_\lambda : [l_i]_T^\lambda = c \Rightarrow c \in [\lambda]_T$$

This means that if the lexical entry l_i is interpreted as c in sentence λ, then c should be part of the interpretation of λ with respect to task T.

Finally, we can define a sense, $\sigma^{(l,c)}$ to be *valid* with respect to a meaning function if the following holds:

$$\exists i \in \mathbb{N}, \lambda \in \mathscr{L} : l_i \in W_\lambda \wedge [l_i]_T^\lambda = c$$

With this definition, we consider the sense to be a reified pair capturing the cases under which it is valid to interpret l as having meaning c, where we understand validity to mean that the sense is used in at least one interpretation for the given task.

Consider the question in our introduction: Who painted the Mona Lisa? In this case: $\lambda =$ *Who painted the Mona Lisa* and:

$$[\textit{painted}]_T^\lambda = \text{dbprop:artist}$$
$$[\textit{Mona Lisa}]_T^\lambda = \text{dbpedia:Mona\_Lisa}$$

Hence we have the following valid senses for our example sentence:

$$\sigma^{paint,dbprop:artist}$$
$$\sigma^{Mona\ Lisa,dbpedia:Mona\_Lisa}$$

In this role, we can understand a sense as the 'glue' between a pair of lexical entry and ontology concept, and also as the container for those pragmatic features (usage, register, etc.) whose role is neither purely ontological nor lexical.

4.3.2 Sense as Subset of Uses

As we have already argued above, the semantic distinctions made in the ontology provide a principled basis for defining a partition of the uses of a certain lexical entry. We define U – the usage set of a lexical entry l – as

$$U(l) = \{(l_i, \lambda) : l_i \in W_\lambda, \lambda \in \mathscr{L}\}$$

For each sense $\sigma^{(l,c)}$ we assume that the usage set u_l^c is defined as:

$$u_l^c = \{(l_i, \lambda) : l_i \in W_\lambda, \lambda \in \mathscr{L}, [l_i]_T^\lambda = c\}$$

For a set of senses Σ, let $\Sigma|_l$ denote all senses whose lexical entry is l. We say that a set of senses is *complete* for a lexical entry l if its usage sets for a task T satisfy:

$$U(l) = \bigcup_{\sigma^{(l,c)} \in \Sigma|_l} u_l^c$$

The existence of a sense linking the lexical entry l and the concept c implies that the lexical entry can be used with this meaning, which is supported by at least one interpretation in the context of the given task. It then follows that a complete set of senses for a lexical entry constitutes a (non-disjoint) partition of all the uses ($U(l)$). By specifying which ontological distinctions are relevant in a given domain, the ontology thus provides a principled criterion to define the senses or meanings of a lexical entry in relation to the given ontology. Thus, the sense represents a subset of the uses of the lexical entry l for which l can be understood as meaning concept c.

4.3.3 Sense as a Subconcept

The sense can also be understood as an implicit concept that captures further aspects of the meaning of the lexical entry that cannot be captured by the ontology. In the following, we try to formalize this idea, borrowing several notions from Guarino [14], including the notion of a conceptualization and an ontological commitment.

Definition. A conceptualization is a triple $\mathscr{C} = (D, W, R)$ with D a universe of discourse, W a set of possible worlds and R a set of conceptual relations on the domain space $\langle D, W \rangle$, where a conceptual relation ρ on $\langle D, W \rangle$ is a function $\rho : W \rightarrow D^*$ from the set W into D^*, the set of all n-ary (extensional) relations on D.

We define the lexical extension of a lexical entry l as a mapping from worlds to its extension,[7] i.e.

$$lex(l) : W \rightarrow D^*$$

We now consider an ontology $O = (\Lambda_O, V_O)$ where the vocabulary can be further divided into $V_O = I_O \cup C_O \cup R_O$ consisting of a set of individuals/instances I_O, a set of concepts C_O and a set of relations R_O. In line with Guarino [14] we also consider an ontological commitment K for an ontology O and conceptualization $\mathscr{C} = (D, W, R)$ as a pair $K = (O, \mathscr{I})$ where $\mathscr{I}$ is a function $\mathscr{I} : V \rightarrow D \cup R$, i.e. $\mathscr{I}$ is an interpretation function that *interprets* the vocabulary of the ontology with respect to the vocabulary of the conceptualization.

We shall now assume that $c \in C_O$ and for each sense $\sigma^{(l,c)}$ define its *ontological projection* as follows:

$$\pi_c^l = lex(l) \cap \mathscr{I}(c)$$

Here, $lex(l)$ and $\mathscr{I}(c)$ are functions from possible worlds to D^*.[8] From this it follows that:

$$\forall w : \ \pi_c^l(w) \subseteq \mathscr{I}(c)(w)$$

If we then add to our ontology a concept $c_{\pi_c^l}$ and extend the ontological commitment such that:

$$\mathscr{I}(c_{\pi_c^l}) = \pi_c^l$$

then we get that:

$$O \models_M \forall x \ (c_{\pi_c^l}(x) \Rightarrow c(x))$$

where M is an *intended model* in the sense of Guarino [14]. For the intended models of the ontology, it follows that $c_{\pi_c^l}$ is thus a subclass of c. This result is similar for the case that $c \in R_O$, in that we derive a similar $c_{\pi_c^l}$ that is a subproperty of c and for $c \in I_O$ we obtain that $c_{\pi_c^l} = c$ (as there is no sub-division of an individual, naturally). These projected or hypothetical concepts $c_{\pi_c^l}$ thus represent the full lexical meaning of entry l when interpreted as concept c.

[7] We adopt an intensional stance here in the sense that the extension of words depends on a certain state of the world.

[8] We denote the intersection of two functions f and g here as $(f \cap g)(w) = f(w) \cap g(w)$.

4.3.4 The Three Facets

As such, we have defined the sense in terms of three closely related entities or roles:

1. $\sigma^{(l,c)}$: The pair representing a correspondence between a lexical entry and vocabulary item in the ontology that it can be interpreted as in a given task.
2. u_l^c: The set of uses of a particular lexical entry l when used as referring to c.
3. π_c^l: a hypothetical concept representing the full lexical meaning of l when interpreted as c.

4.4 Systematic Polysemy in the Ontology-Lexicon Interface

In the previous sections we have argued that there is no universal set of senses for a word, but that the meanings of a word are specific for a given task and domain as described by a given ontology. We think that this is compatible with the view of Ide and Wilks [18] who propose to use a small set of senses that suit a given task. Ide and Wilks for example propose to use three senses for 'paper': (i) material, (ii) daily newspaper and (iii) article, thus excluding the sense of 'publisher of a newspaper'. However, this raises the question how we would interpret the following sentence:

The paper was sued by the Workers Revolutionary Party.

A reasonable interpretation for this sentence might look as follows:

$$\exists x \; sued(WorkersRevolutionaryParty', x) \wedge NewspaperPublisher(x)$$

Now, if *paper* does not have the sense of `NewspaperPublisher`, where does then the part of the above formula – $NewspaperPublisher(x)$ – come from? The meaning of *paper* in the above sentence might be derived as a result of a generative process which brings to the foreground the knowledge that a *newspaper* always has a publisher which publishes it, thus yielding the following meaning:

$$\exists x \; \exists y \; sued(WorkersRevolutionaryParty', y) \wedge Newspaper(x) \wedge publisher(x, y)$$

The meaning of *paper* in the above sentence can thus be approximated by $\lambda y \; \exists x \; Newspaper(x) \wedge publisher(x, y)$. Now where does this meaning come from? Following Hobbs et al. [16] we understand that this can be obtained by a principle of abduction in that we can introduce the property `publisher` as we have axioms within the ontology which state that the range of the `sued` property must be of a type `IndividualOrOrganization`, which is a disjoint class with `Newspaper`. Furthermore, we suppose that the property `publisher` has domain and range of `Publication` and `Organization` which are super/sub classes of `Newspaper` and `IndividualOrOrganization` respectively. Further, we

assume the existence of an axiom stating that every newspaper has a publisher: $\forall x\ Newspaper(x) \rightarrow \exists y\ publisher(x, y)$. Hence, from an ontological point of view, the above interpretation is thus plausible and can thus be deduced abductively by a system performing the interpretation task.

In the linguistics literature, different examples of such a systematic relation between different meanings of a word has been studied under the label of *systematic polysemy*. In the following we give some classical examples of word classes that have systematically related senses (cf. [1] or [24])[9]:

- Animal/meat (The lamb is running in the field vs. John had lamb for dinner)
- Plant/food (Mary watered the fig in the garden vs. Mary ate a fig)
- Producer/product (The newspaper fired its editor vs. John spilled coffee over the newspaper)
- Institution/building (The university became established in the early medieval ages. Versus the university is close to the capitol)

As an example of the institution/building class, consider the following uses of 'school' based on [4]:

- "Daddy drove me to school this morning". ⇒ Daddy drove me to the location of the school building this morning.
- "They painted the school over the holidays" ⇒ They painted the walls of the building which hosts the school.
- "The school was built in 1950." ⇒ The building which hosts the school (as institution) was built in 1950.
- "The school decided to fire the teacher." ⇒ The executive board of the school (as institution) decided to fire the teacher.

Logically, the meanings of the above sentences can be expressed as follows[10]:

- "Daddy drove me to school this morning": $\exists x, l, s\ drive(daddy, x, l) \wedge speaker(x) \wedge location(l) \wedge located\_at(s, l) \wedge educational\_institution(s)$
- "They painted the school over the holidays": $\exists p, b, s\ paint(they, p) \wedge has\_part(b, p) \wedge building(b) \wedge hosted\_in(s, b) \wedge educational\_institution(s)$
- "The school was built in 1950.": $\exists s, e, b\ built(e, b) \wedge happensAt(e, t) \wedge year(t, 1950) \wedge building(b) \wedge hosted\_in(s, b) \wedge educational\_institution(s)$.
- "The school decided to fire the teacher.": $\exists b, t, s\ fire(b, t) \wedge teacher(t) \wedge has\_board(s, b) \wedge educational\_institution(s)$

The above paraphrases suggest that there are (at least) the following different senses of school: (i) address where the school building is located, (ii) building which hosts the school as institution, (iii) walls of the building in which the school (as

[9] Buitelaar [5] gives an overview of many systematic polysemy classes derived from WordNet 1.6.

[10] For the sake of simplicity, we do not represent the temporal adverbials and we model definites through existential quantifiers.

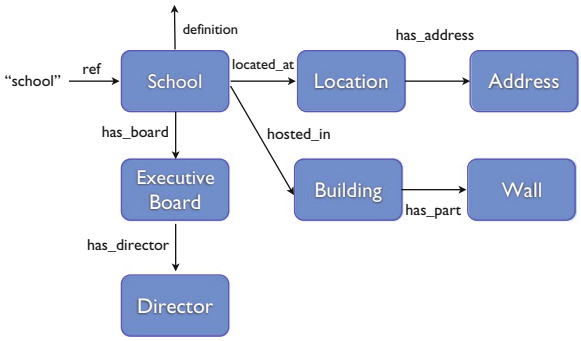

Fig. 4.2 Concept of 'school' in the ontology

institution) is located, (iv) executive board of the school (as institution). Thus, the meanings of 'school' in the above sentences could be formalized as follows:

1. $\lambda l\ location(l) \wedge located\_at(s, l) \wedge educational\_institution(s)$
2. $\lambda p\ has\_part(b, p) \wedge building(b) \wedge hosted\_in(s, b) \wedge educational\_institution(s)$
3. $\lambda b\ building(b) \wedge hosted\_in(s, b) \wedge educational\_institution(s)$
4. $\lambda b\ has\_board(s, b) \wedge educational\_institution(s)$

Now, would we include all of the above senses in a lexicon? Definitely not, for good reasons. In fact, it seems that all of the above mentioned meanings are related in the ontology to the concept *educational_institution* and can thus be generated when interpreting the corresponding sentences by some process of abduction that exploits the ontological neighbourhood of *educational_institution* to bring to the foreground additional – systematically related – meaning aspects as required by the linguistic context.

In fact, we argue that *educational_institution* is the primary meaning of 'school' and that the different linguistic contexts above select one particular or related aspect of the primary meaning of school, emphasizing the building in which it is located, the executive board, the activities that are typically offered at school, etc. If all of the aspects that are relevant for a school are modelled within an ontology (as sketched in Fig. 4.2), most of the above systematically related concepts can be derived through appropriate coercion operations that traverse the ontology to find an entity that is related and that fits the linguistic and semantic context of the sentence in question.

Such a viewpoint is still in agreement with our definition of sense given in Sect. 4.3 as it follows that the senses are still complete. In fact, although not every usage of a word is directly interpreted as *educational_institution*, every interpretation uses the symbol *educational_institution* and as such this sense is complete for the examples above. It is in this way that we argue that systematic polysemy is not a phenomenon that needs to be modeled in the context of the ontology-lexicon interface but exclusively at the ontological level.[11]

[11]Of course, we admit that for a certain application, specifying some of the senses that are derived from the primary one may improveperformance.

As such, from the perspective of the lexicon-ontology interface, the role of a sense is essentially to provide a hook into the ontology graph, reifying the fact that a word evokes a certain concept in the ontology. In our case, we merely need a link between the word 'school' and the concept *educational_institution*.

Thus, from the point of the lexicon-ontology interface, we can thus distinguish two types of lexical ambiguities:

- **Systematic Polysemy:** This case corresponds to the case where a word has different meanings that are systematically related through the ontology. In this case, the different interpretations of the lexical entry can be obtained by a process of abduction based on the axioms present in the ontology.
- **Homonymy:** According to our understanding, homonymy refers to the case where the different meanings of a word are not related through axioms in the ontology, so that two different senses are indeed required.

A consequence, however, is that whether a lexical ambiguity is systematic or irregular depends ultimately on the specific ontology. From a NLP perspective, we think that this represents indeed a principled approach, reducing the number of senses of a word to a minimum while being able to generate systematically related meanings exploiting background knowledge captured in the domain ontology.

4.5 Senses in the Ontology-Lexicon Model Lemon

In earlier work [22], we have proposed *lemon* as a model for representing and sharing ontology lexica using Semantic Web standards such as OWL and RDF. In *lemon*, senses as considered in this chapter are implemented through the class `LexicalSense`, which reifies the link between a `LexicalEntry` and some entity in the ontology (see Fig. 4.3). This link is established via two properties (`reference` and `isSenseOf`) which are specified as functional properties, such that it can be inferred that the sense is unique to the pair (l, c) as in the definition. As a simple example of the representation of senses in *lemon*, we consider the case of translation. In most cases, a certain lexical entry l_1 is not a translation of some other lexical entry l_2 in all contexts. Rather, the "translation property" is dependent on the meaning of lexical entry l_1. Thus, it might be that l_2 is only a translation of l_1 when l_1 is interpreted as concept c. Translation is thus a multi-valued function $trans : L_O \times V_O \rightarrow \mathscr{P}(L_O \times V_O)$, i.e. defined on pairs of lexical entry and concept. For example, the German word 'Krebs' is translated into English as 'cancer' when referring to the illness and as 'crab' when referring to the animal. In this example, it holds that *(cancer,illness)* $\in$ *trans((Krebs,illness))* but (crab,illness) $\notin$ trans((Krebs,illness)). Instead, it holds that (crab,animal) $\in$ *trans((Krebs,animal))*. The property of being a translation is thus a property between senses and not between lexical entries. This is formally stated in *lemon* as follows:

```
:Krebs lemon:sense [ lemon:reference dbpedia:Cancer ] ;
```

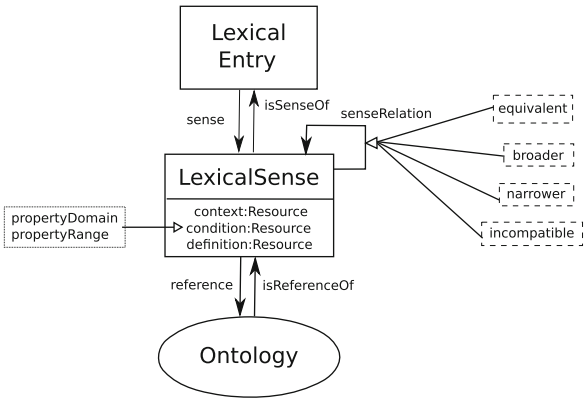

Fig. 4.3 The modelling of senses within the *lemon* ontology-lexicon model

```
    lemon:sense [ lemon:reference dbpedia:Crab ] .

:Cancer lemon:sense [ lemon:reference dbpedia:Cancer ] .

:Cangrejo lemon:sense [ lemon:reference dbpedia:Crab ] .
```

As a sense is a reified pair of the lexical entry and ontology concept, we do not require an explicit translation link to capture the *trans* function defined above (although the schema does not preclude the inclusion of such a link if desired).

4.5.1 Sense Properties

lemon allows additional properties to be attached to sense objects that can be used to describe aspects related to the usage of this lexical entry. In this sense, the reification of sense is crucial to express (i) certain pragmatic implications of using a certain lexical entry to refer to the concept in question and (ii) to state conditions under which it is legitimate to interpret the word as referring to the concept. An example of this might be the subjective or emotional associations that a certain language and culture makes when using a certain lexical entry to refer to a concept, i.e., connotations. Consider the noun 'retardation', which was earlier used to refer to people with learning and developmental difficulties. However, this use is considered extremely pejorative in modern usage.[12] However, according to the current 4th Edition of 'Diagnostic and Statistical Manual of Mental Disorders' (DSM-IV),[13] there is a pathology called "Mental Retardation." The senses that we might distinguish for 'retardation' are the following ones:

[12] As a corollary many charities have changed their original name, for example the "Association for the Help of Retarded Children" is now just the "AHRC", which officially is not an intialism.

[13] Standard international reference for mental health disorders.

- A meaning of developmentally disabled that is used primarily in texts before the 1970s.
- A meaning of unintelligent that is pejorative
- A reference to DSM-IV disorders 317, 318 or 319.

The associated aspects of the usage of the word ‘retardation’ can neither be attached to the lexical entry itself nor to the corresponding classes in an ontology as they describe the pair of lexical entry and concept and in particular constrain (i) in which cases and under which conditions the lexical entry can refer to the concept in question, but also (ii) what connotations the use of this lexical entry has when referring to the concept. In the case of ‘retardation’ as ‘developmentally disabled’, we would attach to the pair the information that this interpretation was mainly valid before the 1970s. With respect to the fact that ‘retardation’ is considered offensive in some contexts, this is neither a property of the lexical entry, as the word can be used in a non-offensive manner, nor of the actual concept `DevelopmentallyDisabled`, as the concept can be expressed in a non-offensive manner, but of the lexical entry when referring to the concept. A reification of the pair of lexical entry and concept is thus needed to express the pragmatic connotations that the word has when interpreted as a certain concept. We model this in *lemon* as follows:

```
:Retardation_entry
   lemon:sense [
     lemon:reference dbpedia:Developmental_Disorder ;
     lexinfo:dating lexinfo:old
   ] ;
   lemon:sense [
     lemon:reference dbpedia:Stupidity ;
     lexinfo:register lexinfo:perjorative
   ] ;
   lemon:sense [
     lemon:reference dsmiv:317
   ] .
```

As can be seen the *lemon* model requires an explicit sense object in its graph as otherwise there would be no sensible place to attach the properties required.

4.5.2 Contexts and Conditions

In order to specify contextual conditions and constraints under which it is legitimate to interpret a lexical entry as referring to a given concept, *lemon* allows to model such contextual conditions using two properties: `context` and `condition`. The property `context` constraints the domains under which the interpretation of the lexical entry as the concept in question is permissible. For example, for the case of ‘retardation’ discussed above, the interpretation as referring to a disorder from the DSM-IV is valid in the medical domain. Two further properties called `dating` and

register are subproperties of context and allow to constrain the time (e.g. before 1970) or register (e.g. informal, colloquial, ...) as conditions under which the lexical entry can be interpreted as referring to the concept in question.

The second property condition is used to state an evaluable expression describing the circumstances that need to be fulffilled such that the lexical entry can be interpreted as the ontological concept in question. The property is abstract and specific properties instantiating it need to be defined. The *lemon* model has two built-in subproperties of condition: propertyDomain and propertyRange. They restrict the usage of the lexical entry, requiring that the domain or range of the ontological property is of a specify type. For example, we could model that the verb 'essen', when interpreted as eat, requires the eater to be human, while 'fressen', when interpreted as eat, requires the eater to be an animal.[14]

The semantics of 'fressen' and 'essen' are thus modelled in lemon as follows:

```
:essen a lemon:LexicalEntry;
   lemon:canonicalForm [  lemon:writtenRep "essen"@de];
   lemon:synBehaviour  [  a lexinfo:TransitiveFrame;
                          lemon:subject :essen_subj;
                          lemon:object :essen_obj];
   isocat:partOfSpeech lexinfo:Verb;
   lemon:sense [ lemon:reference myOntology:eat;
                 lemon:subjOfProp :essen_subj;
                 lemon:objOfProp :essen_obj;
                 lemon:propertyDomain myOntology:Human].

:fressen a lemon:LexicalEntry;
   lemon:canonicalForm [  lemon:writtenRep "fressen"@de];
   lemon:synBehaviour  [  a lexinfo:TransitiveFrame;
                          lemon:subject :fressen_subj;
                          lemon:object :fressen_obj];
   isocat:partOfSpeech lexinfo:Verb;
   lemon:sense [ lemon:reference myOntology:eat;
                 lemon:subjOfProp :essen_subj;
                 lemon:objOfProp :essen_obj;
                 lemon:propertyDomain myOntology:Animal].
```

While *lemon* provides these two built-in properties, many other properties that model contextual conditions are possible. However, these need to be introduced by taking into account specific tasks and have thus not been included in the general model.

[14] *lemon* actually allows to model the corresponding (subcategorization) frames of these verbs and their mapping to ontological properties. This aspect of the model is however not discussed in the present chapter. The interested reader is referred to the *lemon* cookbook [21].

4.5.3 Sense Relations

lemon also has properties for the representation of relationships between senses that are defined based on the facets as defined above. In particular the properties are defined as follows based on the usages u_l^c and the projection π_c^l.

- `equivalent`: The usages of the two senses are equal and the projections are equal.
- `broader`: The usages of the first sense is a superset of the second sense's usage and projections are similarly a superset.
- `narrower`: The usages of the first sense is a subset of the second sense's usage and projections are similarly a subset.
- `incompatible`: The usages of the two senses are disjoint and the projections are disjoint.

These properties have a very different status compared to the properties in the ontology. The properties in the ontology are defined between concepts, while the properties considered here are defined between senses as three-faceted entities introduced in this chapter. These sense relations thus model (lexical) meaning aspects that are not included in the ontology but might nevertheless be important to model for a number of reasons. For example, one might be able to establish relations between different ontologies with different conceptualizations at the sense level if they are difficult to align at the conceptual level (e.g. because they vary substantially in granularity and modelling detail). Consider the property of antonymy for instance. Antonymy is typically a property between words that is not to be confused with the disjointness property between concepts used in many ontology languages. *lemon* introduces the property `antonymy` at the sense level as a subproperty of `incompatible`. The only ontological consequence is that the two projections π_c^l of the senses are regarded as being (ontologically) disjoint.

We can also use the *lemon* properties to capture the relationships between particular interpretations of lexical entries. For example, we consider the example of the French words 'rivière' and 'fleuve', which may be mapped to an ontology that only contains a concept corresponding to the English word 'river', while still ensuring that the terms are considered as not interchangeable. This can be achieved in *lemon* by mapping both words to the same ontology class but indicating that they are incompatible:

```
:Riviere lemon:sense :Riviere_sense .
:Riviere_sense lemon:reference dbpedia:River .

:Fleuve lemon:sense :Fleuve_sense .
:Fleuve_sense lemon:reference dbpedia:River .

:Riviere_sense lemon:incompatible :Fleuve_sense .
:Fleuve_sense lemon:incompatible :Riviere_sense .
```

While sense relations described above do strictly speaking affect neither the actual ontology nor its conceptualization, they are crucial for NLP applications. Take the example of natural language generation and assume that we want to describe a given `river` in French. We might choose to generate the lexicalization 'rivière', but then we should remain consistent and not refer to the same river as 'fleuve'. A NLP system thus needs to know that both senses are incompatible.

From a more general perspective, sense relations allow to represent cultural and linguistic differences in terminology and meaning granularity to be encoded in the lexicon.

4.6 Conclusions

In this chapter, we have revisited the notion of *sense* in the context of the ontology-lexicon interface and argued that the senses that a word has are specific to a task and domain as modelled by a given ontology. Following a principle we call *semantics by reference*, the goal of an ontology lexicon is to define the meaning of a lexical entry relative to the meaning distinctions made in a given ontology. We have argued that in the context of the ontology-lexicon interface, the intrinsic subjectiveness of the answer to the question of which and how many senses a certain word has, can be overcome in a principled way by resorting to the meaning distinctions in the ontology. We have then discussed whether, under this assumption, senses are still meaningful entities in the context of the ontology-lexicon interface. We argue that the notion of sense is necessary in the context of ontology-lexicon interface and that in this context senses can be understood as a three-faceted entity that has the following roles: firstly, a sense can be understood as a *reification of a pair of lexical entry and its corresponding reference in the ontology (concept)*. This is useful to state conditions under which it is permissible to interpret the word as referring to the concept in question. Second, the senses represent *a set of disambiguated uses of an entry when used as referring to a certain concept* in a given interpretation task. Third and finally, a sense represents also a *hypothetical concept* that, if added to the ontology, would be a subclass of the evoked concept. We have further discussed what implications this has for the traditional notion of systematic polysemy, arguing that this is a phenomenon that should be resolved by means of abduction on the axioms in the ontology instead of by recording all possible contextual senses in the lexicon. From this perspective, the role of a sense is to provide a hook to a concept in the ontology, providing an access route to other (systematically related) aspects of the meaning of a word. This hook can then be exploited in the process of interpretation of a sentence in order to bring additional meaning aspects into the foreground as required by the linguistic context and to yield a well-defined interpretation. Finally, we have provided formal definitions of what it means for a sense to exist in the ontology-lexicon as well as details of how this understanding is implemented in the ontology-lexicon model *lemon*.

Acknowledgements This work was developed in the context of the Monnet project, which is funded by the European Union FP7 program under grant number 248458, the CITEC excellence initiative funded by the DFG (Deutsche Forschungsgemeinschaft), the Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion2), and the Spanish national project BabeLData (TIN2010-17550).

References

1. Apresjan, J.: Regular polysemy. Linguistics **142**, 5–32 (1974)
2. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: a nucleus for a web of open data. In: Semantic Web, Busan, vol. 4825, pp. 722–735. Springer (2007)
3. Berlin, B., Kay, P.: Basic Color Terms: Their Universality and Evolution. University of California Press, Berkeley (1969)
4. Bierwisch, M.: Semantische und konzeptuelle Repräsentation lexikalischer Einheiten. In: Ruzicka, R., Motsch, W. (eds.) Untersuchungen zur Semantik. Akademie, Berlin (1982)
5. Buitelaar, P.: CoreLex: systematic polysemy and underspecification. Ph.D. thesis, Brandeis University (1998)
6. Buitelaar, P.: Semantic lexicons: between terminology and ontology. In: Simov, K., Kiryakov, A. (eds.) Ontologies and Lexical Knowledge Bases. Proceedings of the First International OntoLex Workshop. OntoText Lab, Sofia (2000)
7. Buitelaar, P.: Ontology-based Semantic lexicons: mapping between terms and object descriptions. In: Huang, C.-R., Calzolari, N., Gangemi, A., Oltramari, A., Lenci, A., Prevot, L. (eds.) Ontology and the Lexicon: A Natural Language Processing Perspective. Cambridge Studies in Natural Language Processing. Cambridge University Press, Cambridge/New York (2010)
8. Carpuat, M., Wu, D.: Improving statistical machine translation using word sense disambiguation. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 61–72. ACL, Stroudsburg (2007)
9. Cruse, D. A.: Lexical Semantics. Cambridge University Press, Cambridge/New York (1986)
10. Croft, W., Cruse, D. A.: Cognitive Linguistics. Cambridge University Press, Cambridge/New York (2004)
11. Cimiano, P., Buitelaar, P., McCrae, J., Sintek, M.: LexInfo: a declarative model for the lexicon-ontology interface. J. Web Semant. **9**(1), 29–51 (2011)
12. Dagan, I., Itai, B.: Word sense disambiguation using a second language monolingual corpus. Computat. Linguist. **20**(4), 563–596 (1994)
13. Edmonds, P., Kilgariff, A.: Introduction to the special issues on evaluating word sense disambiguation systems. J. Nat. Lang. Eng. **8**(4), 279–291 (2002)
14. Guarino, N., Oberle, D., Staab, S.: What is an Ontology? In: Staab, S., Studer, R. (eds.) Handbook on Ontologies. International Handbook on Information Systems, pp. 1–17. Springer, Berlin (2009)
15. Hanks, P.: Do word meanings exist? Comput. Humanit. **34**(2), 205–215 (2000)
16. Hobbs, J.R., Stickel, M.E., Appelt, D.E., Martin, P.: Interpretation as abduction. Artif. Intell. **63**(1), 69–142 (1993). Elsevier
17. Ide, N., Véronis, J.: Mapping dictionaries: a spreading activation approach. In: Proceedings of the 6th Annual Conference of the Centre for the New Oxford English Dictionary, pp. 52–64. UW Centre for the New OED and Text Research, Waterloo (1990)
18. Ide, N., Wilks, Y.: Making sense about sense. In: Agirre, E., Edmonds, P.G. (eds.) Word Sense Disambiguation, pp. 47–73. Springer, Dordrecht (2006)
19. Kilgarriff, A.: I don't believe in word senses. Comput. Humanit. **31**(2), 91–113 (1997)
20. Leon Araúz, P., Faber, P., Magaña Redondo, P.J.: Linking domain-specific knowledge to encyclopedic knowledge: an initial approach to linked data. In: Proceedings of the 2nd International Workshop on the Multilingual Semantic Web, Koblenz, pp. 68–73 (2011)

21. McCrae, J., et al.: The Lemon Cookbook. http://lexinfo.net/lemon-cookbook.pdf (2010)
22. McCrae, J., Aguado-de-Cea, G., Buitelaar, P., Cimiano, P., Declerck, T., Gomez-Perez, A., Gracia, J., Hollink, L., Montiel-Ponsoda, E., Spohr, D., Wunner, T.: Interchanging lexical resources on the Semantic Web. Lang. Resour. Eval. (accepted for publication)
23. Miller, G.A.: WordNet: a lexical database for english. Commun. ACM **38**(11), 39 (1995)
24. Nunberg, G.: The non-uniqueness of semantic solutions: polysemy. Linguist. Philos. **3.1**, 143–184 (1979)
25. Pustejovsky, J.: The Generative Lexicon. MIT, Cambridge (1995)
26. Unger, C., Cimiano, P.: Pythia: compositional meaning construction for ontology-based question answering on the semantic web. In: Proceedings of the 16th International Conference on Applications of Natural Language to Information Systems (NLDB), pp. 153–160. Springer, Heidelberg (2011)
27. Wierzbicka, A.: Semantic primitives and lexical universals. Quademi di Semantica **X**(1), 96–111 (1989)
28. Wierzbicka, A.: The semantics of color: a new paradigm. In: Pitchford, N.J., Biggam, C.P. (eds.), Progress in Colour Studies: Volume I. Language and Culture, pp. 1–24. J. Benjamins Pub., Amsterdam/Philadelphia (2006)
29. Wittgenstein, L.: Philosophische Untersuchungen. Blackwell, Oxford (1953)

Part II
Event Analysis from Text and Multimedia

Chapter 5
KYOTO: A Knowledge-Rich Approach to the Interoperable Mining of Events from Text

Piek Vossen, Eneko Agirre, German Rigau, and Aitor Soroa

Abstract To automatically understand text, a crucial step is to extract events and their participants. The same event can be *packaged* in many different ways in a language. Capturing all these ways with sufficient precision is a major challenge. This becomes even more complex, when we consider texts in different languages on the same topic. We describe a knowledge-rich event-mining system developed for the Asian-European project KYOTO that can extract events in a uniform and interoperable way, regardless of the way they are expressed and in which language. To achieve this, we developed an open text representation format, semantic processing modules and a central ontology that is shared across seven languages. We implemented a semantic tagging approach that performs off-line reasoning and a module for detecting semantic and linguistic patterns in the tagged data to extract events from a large variety of expressions. The system can efficiently handle large volumes of documents and is not restricted to a specific domain. We applied the system to an English text on estuaries.

5.1 Introduction

Information Extraction (IE) can be described as the task of filling template information from previously unseen text which belongs to a predefined domain [18]. Standard IE systems are based on language-specific pattern matching [13], where

P. Vossen (✉)
VU University Amsterdam, De Boelelaan 1105, 1081HV, Amsterdam, The Netherlands
e-mail: piek.vossen@vu.nl

E. Agirre · G. Rigau · A. Soroa
University of the Basque Country, M. de Lardizabal Pasealekua 1, 20018, Donostia, Spain
e-mail: e.agirre@ehu.es; german.rigau@ehu.es; a.soroa@ehu.es

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_5,

each pattern consists of a regular expression and an associated mapping from syntactic to logical form. The use of ontologies in IE is an emerging field [3]: linking text instances with elements belonging to the ontology, instead of consulting flat gazetteers. IE can be considered as a knowledge-rich approach to filter information from text, mostly using very specific background models. They focus on satisfying precise, narrow, pre-specified requests (e.g. to extract *all directors of movies*) and are able to only detect precise matches (e.g. from web documents) while they do not need to understand the remainder of the text.

This approach does not extend well for event mining, since this latter problem demands complex analysis of different semantic components: the events, their participants and their semantic roles, that can be expressed in many different ways or left implicit. Furthermore, existing semantic paradigms for modeling events such as FrameNet [2] and TimeML [19] are built upon specifications of events that often contradict each other, and no unitary framework for the analysis of events, relations and event participants over time has been applied to document processing so far.

We present a knowledge-rich approach to mining events from text that can handle a large amount of expressions of event information and can be applied to many different languages. It uses an open text representation system and a central ontology that is shared across languages. Ontological implications are inserted in the text through off-line reasoning and ontological tagging. We built a flexible pattern-matching module that searches for ontological and shallow linguistic event structures defined through simple XML profiles. We show that a rich ontology linked to large vocabularies can be used to extract event data from a wide variety of expressions from different languages in an interoperable way. It represents a first step towards the semantic modeling of events in text on a large scale and involving a wide variety of deeper ontological knowledge. The system is developed in the Asian-European project KYOTO[1] and tested for the environment domain.

In the next section, we first explain in more detail the large variety of ways in which event-data can be packaged in languages. In Sect. 5.3, we describe the general architecture of the KYOTO system and in Sect. 5.4 the knowledge structure used. Sect. 5.5 explains the off-line reasoning and ontological tagging process. In Sect. 5.6, we describe the module for mining knowledge from the text that is enriched with ontological statements. Finally in Sect. 5.7, we describe the results of applying the system to text on environmental issues for large estuaries.

5.2 Packaging of Events

People use a large variety of ways to refer to events in language. Whereas *things* such as *fish* can only be referred to by nouns and names in most languages, words in any part of speech can refer to events, e.g. *migration* (noun), *migrate* (verb),

[1] www.kyoto-project.eu

migratory (adjective) or *The Migration Period* (named event). Consequently, event mentions in text exhibit a large variety of syntactic structures as illustrated by the following examples taken from the Internet (italics and bold face added):

- *Adjectival reference:*

  1. In Europe, most **migratory fish species** *completing their cycle between the sea and the river* are currently in danger.
  2. Dams, culverts and other barriers currently block *the movement* of **migratory fish** *to spawning grounds*.

- *Nominal reference:*

  3. Downstream **migration of juvenile fishes** is an adaptation aimed at finding *habitat* and new areas *for feeding*, thereby expanding the feeding areas of the species.
  4. Historically, local economies flourished from the *annual shad run in the spring*, when the **fishes' upriver migration** begins.
  5. Species such as salmon, sturgeon, lampreys and various Cyprinids all *have* **anadromous migration** *patterns*, while Eels *have* **catadromous migration** *patterns*.

- *Verbal reference:*

  6. **Eel migrate** in the opposite sense they spend the longest time of their life in the *river* and *spawn in the sea*.
  7. **Menhaden migrate** *into Chesapeake Bay*

A number of issues are illustrated by these examples. First of all, the syntactic structures vary widely and cannot easily be covered through patterns. In the case of adjectival usage the noun that it modifies (*fish*) is the participant doing the migration. In the case of nominal usage, it can be the following *of*-phrase that holds the participant but also the possessive construction (*fishes'*) that proceeds it. More extreme is the sentential construction in which participants *have* patterns of *migration*, from which the reader needs to infer that the *fish* actually participate in the event. In the case of the verbal expression of *migrate*, it is the subject noun that refers to the participant.

In addition to references to events and the participants, we also find references to other events that are somehow semantically related to *migration*. *Cycles between the sea and the river* (example 1) actually co-refer with the migration process, where species travel from *sea* to *river* and back, and fill in details. In other cases, reference is made to events that have an effect on *migration*, e.g. *barriers block* (example 2), or represent the reasons, e.g. *finding habitat and new areas for feeding* (example 3). We see here that the event *migration* is packaged in many different ways and that the sentence includes aspects of the events (italics phrases) that are either directly related to it (repeating the event and filling in other elements) or that have some causal relation to it.

In some cases the same event is referred to without using the word *migrate* or any of its derived forms. These are called conceptual references, as opposed to the previous lexical references:

- *Conceptual reference:*

  1. Some measures were taken in the late 1880s to provide access for anadromous fishes around dams by construction of rudimentary fishways, or by stocking fish into habitats that historically supported large runs.
  2. The allis shad used to be found in the large rivers but is now extinct in the Netherlands.

Only through our knowledge that *anadromous fish* is a type of *migratory fish* and *allis shad* is a type of *anadromous fish*, we can interpret the rest of these sentences in relation to the *fish migration* process.

Packaging of events is a well-known phenomenon in cognitive science and cognitive linguistics literature. For example, Majid et al. [14] argue that events in language are always packaged through the choice of semantic roles. Within computational approaches this is less commonly accepted as a starting point. A computer program that tries to reconstruct the *migration* event from any of these texts faces a major challenge. It not only needs to deal with the different syntactic structures but also needs to have access to knowledge about migration and decide on the interpretation of the different phrases in relation to the event. The above examples are all in English, but events could be extracted from text in different languages, requiring the following capabilities:

1. Handle a large variety of syntactic structures to express events and (causal) relations between events.
2. Have a semantic typing of the words in the text: what words refer to events and what words can refer to the participants.
3. Know what participants an event takes and what their roles are.
4. Have rich knowledge about the type of event or process to understand causal relations with other events and conditions.
5. Have a large and rich database of semantic relations to inherit properties to more specific words and concepts.
6. Use a uniform and interoperable approach across different languages.

To solve this problem completely, large amounts of deep background knowledge need to be paired with knowledge about the way reference can be made to events and participants in and across languages. In this article, we describe a first step in tackling these problems using a knowledge-rich approach that is interoperable across different languages. Our solution includes the following elements:

- The structure of text is represented in a uniform way across different languages.
- All textual elements are converted into ontological elements in the same way across these languages.
- We use an ontological model that is designed to model events and relations between events.

- The vocabularies of the different languages are mapped to the same shared ontology.
- We use an event extraction module that pairs any textual and structural property with ontological properties.

In the next sections, we will explain each element in more detail.

5.3 KYOTO Overview

The KYOTO system is designed to exploit rich semantic background knowledge packaged in many different linguistic expressions. Because background knowledge plays a major role, KYOTO allows communities to model terms and concepts in their domain, which helps to extract events from text. As such, KYOTO follows a knowledge-rich approach to interpret text that can be extended, tuned and maintained for specific domains. Nevertheless, the architecture of KYOTO is set up as a generic system that can model event structures in any text and any domain.

Figure 5.1 shows an overview of the process in which documents are processed through a pipeline of modules. The knowledge cycle starts with a set of source documents (at the left top side), which are converted to HTML format if necessary. Next, linguistic processors apply tokenization, segmentation, morpho-syntactic analysis and semantic processing to the text in different languages. In the current system, there are processors for English, Dutch, Italian, Spanish, Basque, Chinese and Japanese. The output of the linguistic processors is stored in an XML annotation format that is the same for all the languages, called the KYOTO Annotation Format (KAF, [4]). KAF incorporates proposals for standardized linguistic annotation of text and represents them in a layered structure, compatible with the Linguistic Annotation Framework (LAF, [11]). Once the text is represented in KAF, a series of semantic processing modules is launched that take KAF as input and produce KAF as output with a new conceptual interpration.The semantic processing involves the detection of multiword expressions, named-entities (persons, organizations, places, time-expressions), determining the most-likely synsets of words according to a given wordnet [6] and assigning ontological labels to textual units through the wordnet synsets. The result is that every element in the textual representation will get a corresponding semantic representation in terms of synsets and the associated ontological classes that apply to each synset. For the semantic processing of KAF, the system uses a knowledge base that contains wordnets in seven languages and a shared central ontology.

The KYOTO system then proceeds in two cycles (see Fig. 5.1). In the 1st cycle, we extract potentially relevant terms from the documents represented in KAF, such as *migratory fish* and *anadromous species*. Terms are normalized (sequences of) words that have sufficient frequency and/or many semantics relations with other terms in a set of documents for a domain. The terms are organized as a structured hierarchy and, wherever possible, related to existing concepts in the given knowledge base, i.e. wordnets for each language. For example in the case of

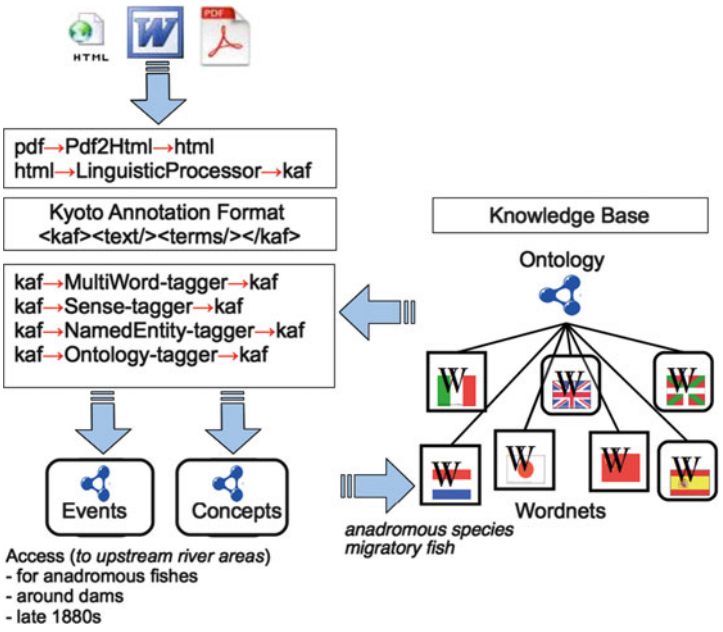

Fig. 5.1 Overview of the KYOTO architecture

migratory fish, the word-sense-disambiguation (WSD) module [1] will determine the most-likely sense of *fish* in the sequence and likewise determine the hypernym synset to which the new term will be connected. Since each wordnet is mapped to the central ontology, also the new terms are ultimately mapped to the ontology. The extended knowledge base is then used for processing new text, adding more precision to the interpretation: while *fish* and *migratory* have two meanings in the general WordNet, *migratory fish* will only have one in the extended WordNet. Customization and tuning of the processing can thus be done by adding more specific knowledge.

From the same KAF with semantic information, we also extract events in the 2nd cycle by so-called Kybots (Knowledge Yielding Robots). Kybots use a collection of profiles that represent patterns of information of interest. In the profile, conceptual relations are expressed using ontological and morpho-syntactic linguistic patterns, e.g. a noun with the ontology class *species* is followed by a verb with the class *change-of-location*. When a match is detected, the instantiation of the pattern is saved in a formal representation. Since the wordnets in different languages are mapped to the same ontology and the text in these languages is represented in the same KAF, similar patterns can easily be applied to multiple languages.

KAF plays an important role in the architecture of the system. In KAF, words, terms, constituents and syntactic dependencies are stored in separate layers with references across the structures. This makes it easier to harmonize the output of linguistic processors for different languages and to add new semantic layers to the

```
<KAF>
<text>
    <wf page="29" sent="770" wid="w10963">the</wf>
    <wf page="29" sent="770" wid="w10964">passage</wf>
    <wf page="29" sent="770" wid="w10965">of</wf>
    <wf page="29" sent="770" wid="w10966">migratory</wf>
    <wf page="29" sent="770" wid="w10967">fish</wf>
<text/>
<terms>
<term lemma="passage" pos="N" tid="t9032" type="open">
   <externalReferences>
   <externalRef conf="0.52" ref="eng-30-03895293-n" res="wneng3.0">
*        <externalRef ref="eng-30-00021939-n"  reftype="baseConcept" res="wn30g"/>
*        <externalRef ref="CommonSenseMapping.owl#geographical-object" reftype="sc_domainOf" res="ontology">
**            <externalRef reftype="SubClassOf" ref="CommonSenseMapping.owl#physical-place"/>
**            <externalRef reftype="SubClassOf" ref="ExtendedDnS.owl#non-agentive-physical-object"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#physical-object"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#physical-endurant"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#endurant"/>
*        </externalRef>
*        <externalRef ref="Kyoto#connect" reftype="sc_participantOf" res="ontology"/>
*        <externalRef ref="Kyoto#has-path" reftype="sc_playRole" res="ontology"/>
    </externalRef>
   <externalRef conf="0.061" ref="eng-30-07310642-n" res="wneng3.0">
*        <externalRef ref="eng-30-07283608-n"  reftype="baseConcept" res="wn30g"/>
*        <externalRef ref="Kyoto#natural_event-eng-3.0-07283608-n" reftype="sc_subClassOf" res="ontology">
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#event"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#perdurant"/>
*        </externalRef>
    </externalRef>
</externalReferences>
</term>
<!-- etc. -->
<term lemma="migratory fish" pos="N" tid="t9035mw" type="open">
<externalReferences>
<externalRef conf="0.409837" ref="dw-eng-30-343-n" res="wneng3.0">
*        <externalRef ref="eng-30-02512053-n"  reftype="baseConcept" res="wn30g"/>
*        <externalRef ref="Kyoto#fish-eng-3.0-02512053-n" reftype="sc_domainOf" res="ontology">
**            <externalRef reftype="SubClassOf" ref="Kyoto#animal-eng-3.0-00015388-n"/>
**            <!-- etc.. -->
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#physical-endurant"/>
*        </externalRef>
*        <externalRef ref="Kyoto#migration" reftype="sc_participantOf" res="ontology">
**            <externalRef reftype="SubClassOf" ref="Kyoto#active-change-of-location"/>
**            <externalRef reftype="Kyoto#done-by" ref="Collections.owl#physical-plurality"/>
**            <externalRef reftype="SubClassOf" ref="Kyoto#change_of_location-eng-3.0-00280586-n"/>
**            <externalRef reftype="Kyoto#has-source" ref="DOLCE-Lite.owl#particular"/>
**            <externalRef reftype="Kyoto#has-path" ref="DOLCE-Lite.owl#particular"/>
**            <externalRef reftype="Kyoto#has-destination" ref="DOLCE-Lite.owl#particular"/>
**            <externalRef reftype="SubClassOf" ref="Kyoto#change-eng-3.0-00191142-n"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#accomplishment"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#event"/>
**            <externalRef reftype="SubClassOf" ref="DOLCE-Lite.owl#perdurant"/>
*        </externalRef>
*        <externalRef ref="Kyoto#done-by" reftype="sc_playRole" res="ontology">
**            <externalRef reftype="InverseObjectProperties" ref="Kyoto#active-participant-in"/>
**            <externalRef reftype="SubObjectPropertyOf" ref="Kyoto#done-by"/>
**            <externalRef reftype="SubObjectPropertyOf"
**              ref="FunctionalParticipation.owl#functional-participant"/>
**            <externalRef reftype="SubObjectPropertyOf" ref="DOLCE-Lite.owl#participant"/>
*        </externalRef>
    </externalRef>
</externalReferences>
</term>
</terms>
<!-- Additional layers (chunking, dependencies, ...) -->
</KAF>
```

Fig. 5.2 Terms in KAF (in *blue*) expanded with ontological tags. Ontological classes from direct mappings are marked with '*' and implied ontological classes are marked with '**' and in *red*

basic output, when needed. All semantic modules for interpreting textual elements into conceptual structures draw their input from this structure. This means that these modules are the same for all the involved languages, resulting in further interoperability. Figure 5.2 shows in blue and without the prefix '*' a shortened example of a KAF structure, representing a text and a term layer. The text layer shows five sequential word tokens (*the passage of migratory fish*) and the term layer shows four corresponding *terms*. Terms have attributes such as lemma,

part-of-speech and a unique identifier. Furthermore, they have elements (*span*) that refer back to the word tokens that make up the terms and references to external sources (*externalReferences*) which represent the semantic interpretation of the textual elements. In the case of *passage*, we see 2 out of a list of 10 wordnet synsets representing its different meanings, where the *conf* attribute indicates the score of the WSD [1]. External references can be nested and here we show the mappings to ontological classes for the first two senses only, prefixed with a '*'. The ontology and the mapping relations are explained in more detail below. In the case of the multiword *migratory fish*, we have a single term that refers back to two word tokens and there is only a single meaning, thanks to the acquisition of concepts in the 1st cycle, which led to the extension of wordnet with the concept *migratory fish*.

5.4 Ontological and Lexical Background Knowledge

Defining terms and concepts in a domain is an important step towards the disclosure of knowledge. In many cases, communities already have large quantities of (semi-) structured vocabularies and thesauri. Modeling these terms and concepts is a huge integration task, possibly involving millions of concepts and relations. To cope with these different types of knowledge, we designed a three-layered knowledge model [21] along the notion of the division of labor [20]. According to this model, we assume that domain experts know how to distinguish rigid and disjoint types of things (as defined by Guarino and Welty [8]) in their domain. There is no need to define the identity criteria for fishes such as *Alosa sapidissima* and *Brevoortia tyrannus* for computers. The simple fact that these are subclasses of an ontological type (e.g. fish) is sufficient to know that they are disjoint, each with a unique set of properties: *Alosa sapidissima* will never become *Brevoortia tyrannus*. Instead, it is more important to model the actual processes and states in which these rigid types of fish can be involved: e.g. being *invasive*, *endangered*. Specialists can consult encyclopedia or text books to find static knowledge about types of species but they urgently need to access textual sources to learn about new trends and environmental changes in local areas over time. We thus argue that software that supports such specialists needs to know what these processes and states are to mine informative events from text. Following these observations, we distinguished three knowledge layers:

1. Domain and background vocabularies in different languages
2. Wordnets in different languages
3. A central ontology shared by all languages

The first layer consists of large volumes of background knowledge and new terms learned from text collections in the domain. This layer is automatically linked to wordnets in different languages. All the wordnets are linked to the English WordNet. The wordnets represent the 2nd layer of knowledge, which is linked to the 3rd layer: the central ontology. Each of these layers has an internal semantic structure, connecting specific concepts to more general concepts and it has specific

mapping relations to the next layer. In this model, it is not necessary to have a mapping relation between all the concepts across the resources, since we can use the internal relations in each resource to find a more general concept with a mapping. Whenever we come across a term such as *Ethmidium maculatum* which is not in WordNet, we traverse the relations in a species database[2] until we find a more general concept (*Brevoortia*) that is matched to WordNet. Next, we traverse the hypernym relations in WordNet until we find a synset (*fish genus*) that is matched to the ontology. When combining vocabularies, we assume the principle that all concepts related to more general concepts are rigid-subtypes unless there is evidence to the contrary. Consequently, we need a specification for non-rigid terms, such as *alien invasive fish* and *migratory fish* to explain (1) that they are **not** rigid types of fish and (2) what their role is in vital processes and conditions. In the next sections, we describe the ontology and the formal model for these relations in more detail.

5.4.1 Ontology

The ontology consists of around 2,000 classes divided over three layers [9]. The top layer is based on DOLCE[3] [15] and OntoWordNet [7]. The second layer are the Base Concepts[4] which cover an intermediate level of abstraction for all nominal and verbal WordNet synsets [12]. Base concepts are hypernym synsets that have relatively many relations to other synsets and cover all different branches of the wordnet hierarchy. Examples of Base Concepts are: *building, vehicle, animal, plant, change, move, size, weight*. They provide an interface from the ontology to a complete wordnet. A third layer consists of domain classes introduced for detecting events and qualities in a particular domain (i.e. environment).

A mapping for every synset in the English WordNet is provided to the ontology, where the so-called Base Concepts guarantee that there is such a mapping through the hyponymy relations: 114,016 mappings to the Base Concepts, 185,666 mappings to the central ontology together with 30,000 mappings from ontology labels to implications in the ontology.[5] The word-to-concept mapping also harmonize predicate information across different parts-of-speech. For instance, *migratory events* are represented by different synsets such as the verb *migrate*, the noun *migration* and the adjective *migratory*, which all inherit the same ontological information corresponding to the *active-change-of-location* class. Furthermore, through the equivalence relations of wordnets in other languages to the English WordNet, this semantic framework can also be applied to other languages.

---

[2] http://www.sp2000.org/

[3] DOLCE-Lite-Plus version 3.9.7

[4] http://adimen.si.ehu.es/web/BLC

[5] This knowledge model is freely available through the KYOTO website as open-source data.

Table 5.1 Rigid and non-rigid synset to ontology mappings

wn:allis shad	hypernym	wn:shad
wn:shad	hypernym	wn:fish
wn:fish	sc_equivalenceOf	ont:fish
wn: anadromous fish	hypernym	wn:migratory fish
wn:migratory fish	hypernym	wn:fish
	sc_domainOf	ont:fish
	sc_playRole	ont:done-by
	sc_participantOf	ont:migration
wn:fish migration	sc_subcassOf	ont:migration (perdurant)
	sc_hasParticipant	ont:fish
	sc_hasRole	ont:done-by
wn:air pollution	sc_subcassOf	ont:pollution (perdurant)
	sc_hasParticipant	ont:air
	sc_hasRole	ont:patient
wn:nitrogen pollution	sc_subcassOf	ont:pollution (perdurant)
	sc_hasParticipant	ont:nitrogen
	sc_hasRole	ont:done-by

5.4.2 Wordnet to Ontology Mappings

Relations from wordnet synsets to the ontology are used to differentiate between rigid and non-rigid concepts. This is done in the following way, where the prefix sc_ stands for synset-to-class:

sc_equivalenceOf: the synset is fully equivalent to the ontological class and inherits all properties; the synset is Rigid;

sc_ subclassOf: the synset is a proper subclass of the ontological class and inherits all properties; the synset is Rigid;

sc_domainOf: the synset is not a proper subclass of the ontological class and is not disjoint (therefore orthogonal) with other synsets that are mapped to the same class; the synset is therefore non-Rigid but still inherits all properties of the target ontology class; the synset is also related to a Role with a sc_playRole relation;

sc_playRole: the synset denotes instances for which the context of the Role applies for some period of time but this is not essential for the existence of the instances, i.e. if the context ceases to exist then the instances may still exist [16];

sc_participantOf: instances of the concept (denoted by the synset) participate in some perdurant class of the ontology, where the specific role relation is indicated by a sc_playRole mapping;

Table 5.1 shows some examples. Using these relations, we can express that the synset *alis shad* is a proper subclassOf the ontological type *fish* because it is related to the synset *shad* as a hypernym, which is related to the syset *fish* as a hypernym, where the latter has an sc_equivalenceOf mapping with the ontological type. For newly acquired non-rigid concepts, such as *anadromous fish* and *migratory fish*,

we create internal wordnet hypernym relations but also an explicit mapping to the ontology to indicate their non-rigid status. This mapping indicates that the synset for *migratory fish* is used to refer to instances of *fish* (not subclasses!), where the domain is restricted to *fish*. Furthermore, these instances participate in the process of migration in the role of done-by. The fact that *anadromous fish* is a hyponym of *migratory fish* implies that it is also non-rigid by definition, whereas the fact that *migratory fish* is a hyponym of *fish* does not imply that the former is rigid. Rigidity is not transitive along hypernym relations but non-rigidity is. The properties of the process migration are further defined in the ontology. As a subclass of *active-change-of-location*, it involves an endurant as a *done-by* participant and it has further roles *has-source*, *has-path* and *has-destination*.[6]

Ideally, all processes and states that can be applied to endurants should be defined in the ontology. This may hold for most verbs and adjectives in languages, which do not tend to extend in specific domains and are part of the general vocabulary. However, domain specific text contain many new nominal terms that refer to domain-specific processes and states, e.g. *fish migration*, *air pollution* or *nitrogen pollution*. These terms are equally relevant as their counter-parts that refer to endurants involved in similar processes, e.g. *migratory fish*, *polluted air*, *polluting nitrogen*. As shown in Table 5.1, we therefore use the reverse participant and role mappings to define such processes as subclasses of more general processes involving specific participants in a specified role.

Our model extends other existing WordNet to ontology mappings. For instance in the SUMO to Wordnet mapping [17], only sc_equivalenceOf and sc_subclassOf relations are used, represented by the symbols $=$ and $+$ respectively. The SUMO-Wordnet mapping likewise does not systematically distinguish rigid from non-rigid synsets. Through the mapping relations, we keep the ontology relatively small and compact whereas we can still define the richness of the vocabularies of languages in a precise way. To summarize, event relations can be derived in the following ways in KYOTO:

1. Wordnet relations between synsets that express role relations between events and participants. These are still rare in the English WordNet.
2. Wordnet to ontology mappings from event synsets to ontological participants and from participant synsets to ontological events
3. Ontological axioms that express role relations between events and participants
4. Inheritance in Wordnet of relations through hyponymy relations and in the ontology through subclass relations

In the next sections, we will explain how we exploit these options for inserting the semantic information in the KAF representations and to use these for extracting events and event relations in texts.

[6]The mapping relations from wordnet to the ontology, need to satisfy the constraints of the ontology, i.e. only roles can be expressed that are compatible with the role-schema of the process in which they participate.

5.5 Off-Line Reasoning and Ontological Tagging

The ontological tagging represents the last phase in the KYOTO annotation pipeline described in Sect. 5.2. It consists of a three-step module to enrich the KAF documents with knowledge derived from the ontology. For each synset connected to a term, we first add the Base Concepts to which the synset is related through the wordnet hypernym relations. Next, through the synset to ontology mapping, we add the corresponding ontology type with appropriate relations. Once each synset is annotated with its ontology type, we finally insert the full set of ontological implications that follow from the ontology. The ontological implications are extracted from the OWL representation of the ontology and stored in a static table for all ontological classes. The main purpose is to optimize the performance of the mining module over large quantities of documents, but it is also very useful for debugging.

Figure 5.2 shows, in red and prefixed with '*' and '**', a fragment of the result of onto-tagging for the correct meanings of *passage* and *migratory fish*. Compared to the blue parts we see an additional reference to a Base Concept and the ontological mappings have been expanded with a series of implications (marked with '**') resulting from the offline reasoning. For example, the implications reflect the subclass hierarchy of the ontology and indicate that the first sense of *passage* is an *endurant* and the second sense is a *perdurant*. In the case of *migratory fish*, we see that the mapping as a participant to the ontology class *Kyoto#migration* gives us the implied information that this event also involves the roles *has-source*, *has-destination* and *has-path*.

There are a number of advantages for expanding the KAF representation with ontological implications. First of all, we can now formulate patterns of ontological classes or base concepts instead of looking for sequences of words or synsets. We thus need less patterns to capture more event structures. It is relatively easy to experiment with patterns at different levels of specificity to find the optimal balance between precision and recall (e.g. searching either for *perdurants*, *accomplishment* or *changes of locations*). Secondly by making the implicit ontological statements explicit, we can find the same relations in many different expressions with different surface realizations: *fish migration*, *migratory fish*, *migration of fish*, *fishes that migrate* etc. Since these expressions share the same ontological implications, we can apply similar patterns for the extraction of events. Thirdly, event-participant relations that are not overtly expressed but are semantically implied are still available for matching and can be used to create relations with surrounding expressions, e.g. *passage* can fill the has-path role of *Kyoto#migration* that is implied by *migratory fish*. The same implication will also be represented for terms such as *anadromous fish* as a hypernym of *migratory fish*. Furthermore, the implications will be represented in the same way across different languages, thus facilitating cross-lingual extraction of events. Finally, onto-tagging is a kind of off-line ontological reasoning through which the pattern matching can be relatively easy, fast and robust. There is one big disadvantage to this approach in that the size of the KAF files is expanded by a factor of 20.

5.6 Event Extraction

Kybots (Knowledge Yielding Robots) are programs that find sequences of concepts to extract instances of events, participants and relations in KAF documents. The Kybot server loads a set of profiles that express patterns of such sequences and compiles them into Kybots that scan enriched documents in KAF for matches. In case of a match, the Kybot server will output elements from the text into a a specified output format. Due to our ontology insertion method, these KAF files include all possible implications of all word meanings of the text, which can all be used for matching in the profiles. The Kybot module uses two different methods to find event-participant relations:

1. Profiles that represent sequences of terms exhibiting event-participant relations
2. Complex terms that exhibit an event-participant relation as part of their meaning

The Kybot profiles have a declarative XML format, which describes general morpho-syntactic patterns and semantic conditions on sequences of terms. Linguistic patterns can include morphological and lexical constraints but also semantic conditions that must hold for terms. Kybot are thus able to search for term lemmas or part-of-speech tags but also for terms linked to ontological process and states using the mappings described in before. Figure 5.3 presents an example of a profile. The profiles consist of three main parts:

- Variable declaration (`<variables>` element): defines the search entities e.g.: **x** (denoting terms whose part-of-speech is noun and lemma is not *people*), **y** (which are terms whose lemma is *move*, *migrate* or *travel*), **p** (which are the prepositions *into* or *to*) and **z** (terms linked through one of its synsets to a subclass of the ontological class *CommonSenseMapping.owl#geographical-object*).
- Relations among variables (`<rel>` element): specifies the relations among the previously defined variables e.g.: **y** is the main pivot, variable **x** must precede variable **y** in the same sentence, variable **p** follows **y** and variable **z** must follow variable **p**. Thus, this relation declares patterns like '$\mathbf{x} \rightarrow \mathbf{y} \rightarrow \mathbf{p} \rightarrow \mathbf{z}$' in a sentence.
- Output template: describes the output to be produced for every matching structure e.g.: each match generates a new event targeting term **y**, which becomes the main term of the event with two roles: the 'done-by' role filled by term **x** and 'destination-of' role, filled by **z**.

The profile in Fig. 5.3 would match a sentence such as *Menhaden migrate into Chesapeake Bay* and output the structure of Fig. 5.4. This example shows that we can directly use any ontological class that is inserted in KAF to constraint the variables. Likewise, we can formulate patterns that capture any ontological feature that is either directly or indirectly associated with a word meaning in the text, to express either an event, a participant or a relation. We can therefore replace lexical constraints such as the disjunction *move or migrate or travel* by a more

```
<kprofile>
 <variables>
  <var name="x" type="term" pos="N" lemma="! people"/>
  <var name="y" type="term"
       lemma="move | migrate | travel"/>
  <var name="p" type="term"  pos="P" lemma="into | to"/>
  <var name="z" type="term"
       ref="CommonSenseMapping.owl#geographical-object"
       reftype="SubClassOf"/>
 </variables>
 <relations>
  <root span="y"/>
  <rel span="x" pivot="y" direction="preceding"/>
  <rel span="p" pivot="y" direction="following"/>
  <rel span="z" pivot="p" direction="following"/>
 </relations>
 <events>
  <event target="$y/@tid" lemma="$y/@lemma" pos="$y/@pos"/>
  <role target="$x/@tid" rtype="done-by" lemma="$x/@lemma"/>
  <role target="$z/@tid" rtype="destination-of" lemma="$z/@lemma"/>
 </events>
</kprofile>
```

Fig. 5.3 Example of a Kybot profile

```
<event eid="e97" target="t9643" lemma="migrate" pos="V"
  synset="eng-30-01857093-v" rank="0.5"/>
<role rid="r191" event="e97" target="t9646mw"
  lemma="chesapeake bay" pos="N" rtype="destination-of"
     synset="eng-30-09243405-n" rank="1.0""/>
<role rid="r84" event="e97" lemma="menhaden"
    target="t9642" rtype="Kyoto#done-by"/>
```

Fig. 5.4 Output structure resulting from a Kybot profile

powerful ontological constraint such as the class *Kyoto#active-change-of-location*. Similarly, we can replace the exclusion of the lemma *people* by the ontology class *Kyoto#person-eng-3.0-00007846-n*, which captures all words and expressions in wordnet that relate to this class. The resulting profile would not only match many more expressions in English but, after adapting the prepositions, would also work for many other languages linked to the same ontology through their wordnet. Through closure of the ontology and wordnets by the Base Concepts, i.e. every synset in wordnet is linked to a Base Concept and every Base Concept is mapped to the ontology, we can thus guarantee maximal coverage of the profiles. It is therefore possible to detect similar event information within and across documents even if expressed differently and in different languages.

One drawback of the profiles is that they can only relate sequences of distinct terms that represent events and participants. In many cases, the event and a participant are both implied by a single term. For example, role-denoting terms, such as *migrant*, *prey*, *predator*, refer to participants and implicitly also to the event in which they are involved. Similarly, event-denoting terms such as *migration* already imply participants even when they are not explicitly mentioned in the surroundings of the term. Actually, one of the effects of acquiring terms for a specific domain

```
<event eid="e3"
       lemma="Kyoto\#change\_of\_location-eng-3.0-00280586-n"
       target="t8570mw" profile_id="complex_term"/>
<role rid="r3" event="e3" lemma="migratory fish"
       target="t8570mw" rtype="Kyoto\#done-by"
       profile_id="complex\_term"/>

<event eid="e28" lemma="spawn"
     target="t8575" profile_id="complex_term"/>
 <role rid="r42" event="e28"
       lemma="Kyoto\#fish-eng-3.0-02512053-n"
       target="t8575" rtype="Kyoto\#done-by"
       profile_id="complex\_term"/>
```

Fig. 5.5 Events and participants extracted from the terms *migratory fish* and *spawn*

is that many multiword expressions, such as *migratory fish*, *murky water* and *crab exploitation*, become single terms in our KAF representation and likewise cannot be matched through the Kybot profiles. Whereas the domain acquisition adds semantics and precision for these words, we loose the possibility to detect the sequence of elements. To be able to still exploit the semantic richness of such terms (both generic and domain specific), we defined special kybots which extract event-participant relations that are implicit. The so-called complex-term process works in two ways:

1. Search for terms that are events (subclasses of perdurant) and look for any role that is defined within the same set of ontological implications related to the same word meaning;
2. Search for terms that are potential participants (endurants) and look for roles and events expressed within the same set of ontological implications related to the same word meaning;

In the first case, the Kybots will output an event represented by the term and a role by the ontological class of the role that is defined. In the second case, the Kybots output an event represented by the ontological class whereas the participant is represented by the term. Figure 5.5 shows the event representation for two such terms *migratory fish* and *spawn*. In the case of the domain term *migratory fish*, the term is the lemma for the role *done-by* and the ontological class *Kyoto#change_of_location-eng-3.0-00280586-n* is given as the lemma for the event. Both the role and the event have the same term identifier as the target. In the case of the generic verb *spawn*, we see that the verb is the lemma for the event and that the ontological class "Kyoto#fish-eng-3.0-02512053-n" is the lemma for the role. Again, both the event and the role have the same target term identifier.

The representation of the implied event and the implied role is important because they do not only capture relations outside the scope of the profiles but can also connect to other elements in the text. In the surrounding of *migratory fish*, we may find concepts for the source, path or destination of the *migration*. In the surroundings of the verb *spawn*, we can expect other concepts related to *fish* even when these *fish* are not explicitly mentioned.

5.7 Experimental Results

To evaluate the platform, we carried out an in-depth evaluation on a single document and we applied the system to a large volume of documents. Finally, we applied the same system to another domain (medical) and to another language (Dutch).

5.7.1 In-Depth Evaluation

The event structure in KYOTO is rather specific and events can be complex. To be able to compare our results with other systems and gold-standards, we defined a more neutral triplet format

```
<R, E, P>
```

where R is a relation, E is a set of word tokens representing the event and P is a set of word tokens representing a participant. If an event has multiple participants, a separate triplet is created for each event-participant pair. The triplet identifier is used to mark which triplets relate to the same event. The Kybot output shown in Fig. 5.4 is then converted to the following two triplets, where the target term identifiers are converted to word token identifiers:

```
<triplet id="941" profile_id="" relation="destination-of">
    <eventids comment="migrate">
        <event id="w11698"/>
    </eventids>
    <participantids comment="Chesapeake Bay">
        <participant id="w11700"/><participant id="w11701"/>
    </participantids>
</triplet>
<triplet id="941" profile_id="" relation="done-by">
    <eventids comment="migrate">
        <event id="w11698"/>
    </eventids>
    <participantids comment="Menhaden">
        <participant id="w11697"/>
    </participantids>
</triplet>
```

A range of (possibly disjoint) token identifiers can be given in a triplet, as shown for *Chesapeake Bay*. Events and participants across triplets therefore match if at least one identifier overlaps, while the relation is the same. Abundance of identifiers is blocked. **P**recision, **R**ecall and **F**-measure are then calculated as follows, where C is the correct system triplets, N_{GS} is the total number of gold standard triplets and N_S is the total number of system triplets:

$$P = \frac{C}{N_S} \qquad R = \frac{C}{N_{GS}} \qquad F = \frac{2(PR)}{(P+R)}$$

Table 5.2 Document statistics

	Nouns	Verbs	Adjectives
Nr. of Terms	893	375	201
Sense tokens:	3,013	3,668	680
Average polysemy	3	10	3
Sense types:	1,065	1,007	353
Base concept tokens:	3,013	3,668	680
Base concept types:	144	223	75
Ontology tokens	14,530	24,763	2,717
Ontology types	573	484	160
Implied ontology tokens:	73,639	126,275	10,262
Implied ontology types:	524	480	214

Table 5.3 Synset to ontology mappings in the text

Mapping	Noun	Verb	Mapping	Noun	Verb
sc_domainOf	63		sc_resultOf	268	30
sc_hasParticipant	294	1,486	sc_simpleCauseOf	43	179
sc_participantOf	686	14	sc_hasCoParticipant	26	
sc_hasRole	251	1,154	sc_playCoRole	26	
sc_playRole	402	6	sc_equivalentOf	463	1,634
sc_subClassOf	3,978		Total	7,123	4,613

For the in-depth evaluation, we took a single document[7] about the Chesapeake Bay, a large estuary in the US. The document has 16,145 word tokens. We manually annotated 132 sentences (2,927 word tokens) from the document with events, participants and their roles. This resulted in 263 events and 470 triplets. We processed the text using the KYOTO system, where we used the generic English WordNet, the KYOTO ontology and a domain wordnet with 990 terms from the environment that have been mapped to the generic WordNet and to the ontology (including the term *migratory fish*). Table 5.2 shows some of the statistics for the document after processing it with the KYOTO system. Average polysemy for nouns and adjectives is three but ten for verbs. Consequently, almost three times as many nouns in the text yield the same number sense meanings and a similar amount of base concepts as the verbs. Furthermore, we see that the nouns result in 14 K mappings to ontology classes, the verbs in 24 K mappings and the adjectives in 2 K mappings. Even though verbs map to many more classes, in the end this boils down to the same proportion of distinct classes (about 500 different types, which is 25 % of the ontology). The ontology classes yield more implied ontology classes, which are classes resulting from the semantics in the ontology. For the verbs 126 K classes apply, which is 1.7 times the amount of classes that apply to the nouns. Table 5.3 shows the important synset-to-ontology mappings for events

[7] www.acb-online.org/pubs/BayBarometer2008Web.pdf

Table 5.4 Baseline and Kybot results on a gold standard of 132 sentences with 263 events and 470 triplets

	Baseline	Kprofiles	Cterms	Profiles-Cterms	Profiles-Wsd	Profiles-Wsd-Cterms
Nr. events	1762	773	32	795	719	741
Nr. correct	319	239	16	249	227	237
Precision	0.18	0.31	0.50	0.31	0.32	0.32
Recall	1.0	0.91	0.06	0.95	0.86	0.90
F-measure	0.32	0.46	0.11	0.47	0.46	0.47
Nr. triplets	4,688	644	19	663	511	530
Nr. correct	131	181	10	191	164	174
Precision	0.03	0.28	0.53	0.29	0.32	0.33
Recall	0.28	0.39	0.02	0.41	0.35	0.37
F-measure	0.05	0.32	0.04	0.34	0.33	0.35

and participants. Obviously, *sc_equivalentOf* and *sc_subClassOf* are most frequent (62 %) but the remainder mostly introduces event-role relations.

To evaluate our system, we created 261 profiles that were applied to the 132 sentences of the gold standard. The profiles consider all ontological classes associated with all meanings. However, these meanings were scored by the WSD system. We therefore considered two variants of the system: one considering all meanings and one considering only the meanings with the highest rank if there was a choice to be made between alternative interpretations of profiles (see [22] for more details on the role of WSD in the process of event extraction). In addition to the profiles, we also extracted relations through the complex-term approach. Combining these options, we get the following variants:

1. Kprofiles: applying the 261 profiles to all different meanings of words
2. Cterms: detecting event-participant relations implied by the meaning of a single term (possibly a multiword term)
3. Profiles-Cterms: combining the results of 1 and 2
4. Kprofiles-Wsd: applying profiles only to the meanings with the highest word-sense-disambiguation score if there is a choice between profiles
5. Profiles-Wsd-Cterms: combining 4 with 2

As a baseline, we created triplets for all heads of constituents in a single sentence according to the constituent representation of the text in KAF. The baseline generates 4,688 triplets for the annotated sentences. Since there is no relation predicted, we assume the most-frequent patient relation for all.

Table 5.4 shows the results of the baseline and the Kybot variants. The top part of the table shows the results for detecting the 263 gold standard events. The baseline and Kybot profiles have high recall (100 and 91 %). The baseline gives an extremely low precision, whereas the precision of the Kybot profiles is 31 %. Precision gets slightly higher when we apply WSD. The Cterms heuristics has low recall but higher precision (50 %). We get the best f-measure by combining profiles, WSD and Cterms. For the triplets in the lower part of the table, we see similar results even

Table 5.5 Baseline and Kybot results on four sentences containing *migration*, *migratory* and *migrate*, representing 20 events and 43 triplets in the gold-standard

	Baseline	Kprofiles	Cterms	Profiles-Cterms	Profiles-Wsd	Profiles-Wsd-Cterms
Nr. events	79	48	3	48	42	42
Nr. correct	20	17	3	17	15	15
Precision	0.25	0.35	1.00	0.35	0.36	0.36
Recall	1.00	0.85	0.15	0.85	0.75	0.75
F-measure	0.40	0.50	0.26	0.50	0.48	0.48
Nr. triplets	344	52	6	56	42	46
Nr. correct	5	8	3	10	8	10
Precision	0.01	0.15	0.50	0.18	0.19	0.22
Recall	0.12	0.19	0.07	0.23	0.19	0.23
F-measure	0.03	0.17	0.12	0.20	0.19	0.22

though the task is more difficult. Again, the Cterms approach has highest precision (53 %) and lowest recall and WSD adds precision to the profiles. We obtain the highest f-measure by combining them.

We also measured the results for four sentences (including examples 2, 4 and 7 from Sect. 5.2) that explicitly refer to migration using nominal, adjectival and verbal forms. The results are shown in Table 5.5. We see that the profiles perform slightly better on events but much worse on the triplets. The Cterms, on the other hand, perform much better on both events and triplets. Through the Cterms, the combined results recover a little but the best results still have a lower f-measure of 22 % compared to 35 %. This shows that the examples we considered are more complex than average compared to the gold-standard. It also shows that the Cterm approach can significantly contribute to the precision and recall of the system if sufficient terms are added to the knowledge base. In the current system, we added only 990 terms for the domain.

5.7.2 Large Scale Evaluation

The 291 English profiles have been optimized to extract the relations from a single document on environmental issues. After that they have been applied to almost 9,000 documents on environmental issue from various sources. We also applied the same profiles to another domain without adaptation: seven documents on medical breast cancer. These documents together contain about 25 million words and the profiles extracted 890 thousand events. Table 5.6 gives overview of the extracted data.

Table 5.6 Events extracted during the KYOTO project

	Docs	Word tokens	Events	Roles	Date refs	Dates	Places	Countries
Estuary	4,625	3,091,842	470,762	231,630	102,653	1,168	2,409	176
WWF International	1,174	1,966,914	264,743	331,391	38,057	711	1,224	146
Journal of Environmental Biology	791	3,440,611	23,406	27,782	51,188	696	2,306	82
European Environment Agency	713	4,814,647	47,355	58,628	105,952	662	1,348	93
Hydrology and Earth System Sciences	1,355	11,228,175	71,781	85,276	157,057	2,380	4,407	116
Medical breast cancer protocols	7	110,501	8,416	15,984				
Total	8,758	24,695,387	890,558	757,553	463,025			

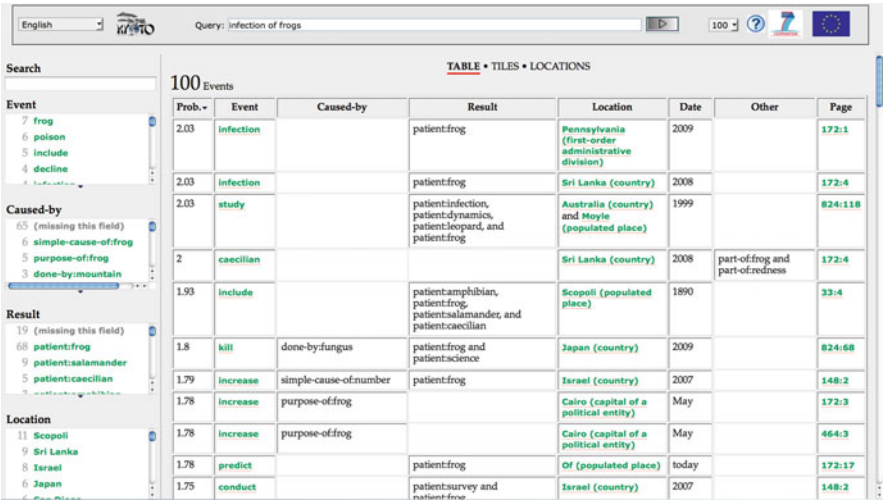

Fig. 5.6 Search results in table form for the query *infection of frogs*

All the data and the Kybot profiles can be downloaded from the KYOTO website. They are available in the KYOTO output format and in RDF format. Since most events are placed in time and space, we can consider them as (partial) descriptions of facts. As suggested above, we can compare events within the same region and time-frame. To illustrate this, we developed a semantic search system that uses a multi-lingual index on Kybot facts. We index the facts by the lemmas, the synset-ids and the synset-ids of the hypernyms for each event and role.[8]

To search the indexed facts, a client program was developed through the Exhibit API.[9] Exhibit [10] consists of Java-script packages that provide advanced functionality to display structured data. The structured data can be published by any server (e.g. as Google spreadsheets) and are loaded in the browser of the user together with the Java-script. The local database of the user is accessed to further present the data. For KYOTO, the retrieved data are converted to a Json structure.

Queries are first lemmatized and sent to a word-sense-disambiguation server to obtain the most likely concepts associated with the words. The client receives the facts that have been indexed, orders them by the strength of the matches, and displays the 100 best facts. The databases mentioned in Table 5.6 can be searched through the demo that is available on the Kyoto website.[10] Figure 5.6 shows a

[8]For cross-lingual retrieval, the lemmas have been translated to all the other languages in KYOTO, using the equivalences in the wordnets. The databases can be searched in any of the languages and the results are rendered in the query languages, regardless of the source language of the information.

[9]http://simile-widgets.org/wiki/Exhibit

[10]Follow the next URL to search in the Estuary database. Login with any name and any password: http://kyoto.irion.nl/kyoto/web/init.do?project=estuary_en&database=2&queryLg=en&query=

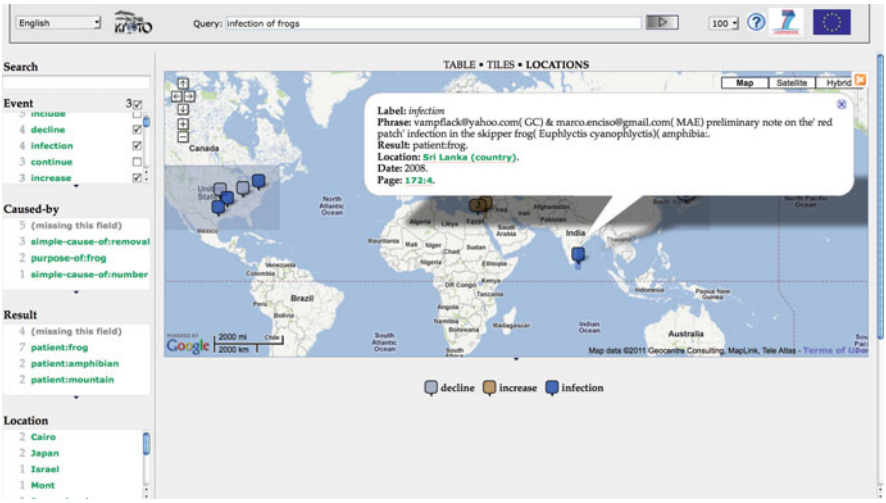

Fig. 5.7 Search results on Google map for filtered results of the query *infection of frogs*

screen dump of the results when searching for *infection of frogs* in the Estuary database. The results are shown in a structured table with columns for the probability (matching score), the lemma for the event, causal roles, result roles, the location, date, other roles and finally a list of pointers to the sources. The table can be sorted by each column and you can click on the cell values to obtain further details. At the left side, filter tables are given for each column. They list the unique values with their frequency. By clicking on these values, a selection from the full table is shown.

The Exhibit API lets you display the Json structure in different ways, among which on Google maps. This is shown in Fig. 5.7, where the results have been filtered by selecting the events *infection*, *decline* and *increase*. The events are depicted on the map and by clicking an event information from the source is shown as for the event located in Sri Lanka.

We carried out a user-evaluation on three different retrieval systems:

1. A standard text search with a Google-like result list;
2. A mashup system that converts the results from the standard text search into similar Exhibit tables;
3. The semantic search on the Kybot output;

Sixteen students and six environment professionals participated in the study. The participants had to answer six questions per system, after a short introduction and practice with each system. Different groups answered different questions with different systems and in different order. Across the different system, we could not measure any significant difference in the quality of the answers and the time to find the answers. We also asked the users to provide feedback through the SUS-tool (a tool that measures usability; [5]). The feedback showed that out of 20 subjects,

most users preferred the benchmark tool over the semantic search. The standard text search system scored best on usability and learnability, probably because it matches the experience all users have with Google. The system acts in the way they expect, matching phrases and presenting the results with snippets. These same users are confused by the semantic search which finds matches through concepts rather than phrases. However, another (smaller) group of subjects disliked the benchmark because it did not enable them to refine their search term or search very effectively. They preferred the semantic search because of its extra functionality.

We believe that semantic search is disliked by *conservative* users, who wish to be able to use a tool immediately, and who prefer the presentation to be familiar, so that they do not have to spend time learning to use the tool. However, it is liked by more *adventurous* users, who will invest time to investigate the extra functionality if they believe it will help them to search more effectively in the end, and to find better information. Presumably, there is also a *middle* group which could be persuaded to adopt the tool if its user-friendliness were improved, and/or if they were shown its potential and how to use it by fellow workers.

5.7.3 Transferring to Another Language

An important aspect of the KYOTO system is the sharing of the central ontology and the possibility to extract semantic relations in different languages in a uniform way. To test the feasibility of sharing the same semantic backbone and transferring Kybot profiles, we carried out a transfer experiment from English to Dutch. We collected 93 Dutch documents on a Dutch estuary (the *Westerschelde*) and related topics. We created KAF files using the Dutch parser Alpino[11] and applied WSD to these KAF files using the Dutch wordnet.

To apply the profiles to the Dutch KAF documents, we need to apply the ontology tagger to the Dutch KAF. However, the tables map the English WordNet to the ontology and not the Dutch wordnet. We therefore generated Dutch variants of the tables on the basis of the equivalence relations between the Dutch wordnet and the English wordnet. For each Dutch synset, we looked up all the equivalent synsets in English, next we looked up the English synset in the ontology tag tables. If there was a match, we created an entry for the Dutch synset in the new table with the same mapping. Likewise, we created tables that match every Dutch synset to the English Base Concepts and to the ontology. Some Dutch synsets have no equivalence and some have multiple equivalences. We generated 145,189 Dutch synset to English Base Concept mappings (for comparison for English we have 114,477 mappings) and 326,667 Dutch synset to ontology mappings (186,383 for English). These ontology tag tables were used to insert the ontological implications into the Dutch KAF files.

[11] http://www.let.rug.nl/vannoord/alp/Alpino/

Table 5.7 Roles related to the noun *toename* (increase) and the verb *stijgen* (increase)

toename (increase)			*stijgen* (to increase)		
Lemma	Role	Freq.	Lemma	Role	Freq.
Aantal (number)	Patient	1	Bodem (ground)	Patient	1
Activiteit (activity)	Patient	1	Zeespiegel (sea level)	Patient	3
Consumptie (consumption)	Patient	16	Zeespiegel (sea level)	Done-by	1
Vervuiling (pollution)	Patient	16	Aarde (earth)	Simple-cause-of	4
Introductie (introduction)	Done-by	16	Aarde (earth)	Patient	4
Atmosfeer (atmosphere)	Generic-location	2			
Handel (trade)	Patient	4			
Druk (pressure)	Patient	4			

Finally, we adapted the 261 English Kybot profiles to replace all English specific elements by Dutch. This mainly involved:

- Replacing English prepositions and relative clause complementizers by Dutch
- Adapting the word order sequences for relative clauses in Dutch
- Adapting profiles including adverbials
- Eliminating profiles for multiword compounds which hardly occur in Dutch
- Eliminating profiles for explicit English structures that express causal relations

We kept all the ontological constraints exactly as they were for English. Only superficial syntactic properties were thus changed. It took us half-a-day to adapt the profiles for Dutch. From the original 261 English profiles, we obtained 134 Dutch profiles. We ran the profiles on the 93 Dutch KAF files (42,697 word tokens) and 65 profiles generated output. In terms of relations, we see a similar distribution as for English: the patient relation is most frequent, followed by relations such as generic-location, has-state and done-by. We did a preliminary inspection and the results look reasonable. For instance, two frequent words denoting events: the noun *toename* (increase) and the verb *stijgen* (increase) appear to have sensible patients, shown in Table 5.7.

5.8 Conclusion

We described a knowledge-rich approach to the interoperable extraction of event data from text, expressed in different ways and across different languages. We use a shared representation formats for seven different languages and shared modules for the semantic processing of the text. Ontological implications from a single shared ontology are inserted in the text using wordnets and WSD. We used a pattern-matching module to extract event-participant relations from text running over these

ontological statements. We evaluated the system on sentences of a single document, which showed promising results.

In the near future, we will extend the evaluation of the system to other types of text and more languages. We will also exploit many more options to use semantic constraints on interpreting sequences that have not been exploited yet. We will especially investigate more precise ways in which WSD can be combined with the task to come to an interpretation of the textual elements that makes sense. Finally, we will work on the more complex ways in which the different events fit together. So far we consider each event as separate but the examples in Sect. 5.2 showed that event descriptions overlap to a high degree.

Acknowledgements The KYOTO project is co-funded by EU – FP7 ICT Work Programme 2007 under Challenge 4 – Digital libraries and Content, Objective ICT-2007.4.2 (ICT-2007.4.4): Intelligent Content and Semantics (challenge 4.2). The Asian partners from Tapei and Kyoto are funded from national funds. This work has been also supported by Spanish project KNOW-2 (TIN2009-14715-C04-01).

References

1. Agirre, E., Soroa, A.: Personalizing PageRank for word sense disambiguation. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL '09), pp. 33–41. Association for Computational Linguistics, Stroudsburg (2009)
2. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The berkeley framenet project. In: Proceedings of the 17th International Conference on Computational Linguistics, COLING '98, vol. 1, pp. 86–90. Association for Computational Linguistics, Stroudsburg (1998)
3. Bontcheva, K., Wilks, Y.: Automatic report generation from ontologies: the MIAKT approach. In: Nineth International Conference on Applications of Natural Language to Information Systems (NLDB'2004). Manchester (2004)
4. Bosma, W.E., Vossen, P., Soroa, A., Rigau, G., Tesconi, M., Marchetti, A., Monachini, M., Aliprandi, C.: Kaf: a generic semantic annotation format. In: Proceedings of the GL2009 Workshop on Semantic Annotation, Pisa (2009)
5. Brooke, J.: SUS: a quick and dirty usability scale. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A., McClelland, A.L. (eds.) Usability Evaluation in Industry. CRC (1996)
6. Fellbaum, C.: WordNet: an Electronical Lexical Database. The MIT Press, Cambridge (1998)
7. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Interfacing WordNet with DOLCE: towards OntoWordNet. In: Ontology and the Lexicon, pp. 36–52. Cambridge University Press, Cambridge/New York (2010)
8. Guarino, N., Welty, C.: Evaluating ontological decisions with ontoclean. Commun. ACM **45**(2), 61–65 (2002)
9. Hicks, A., Herold, A.: Evaluating ontologies with rudify. In: Dietz J.L.G. (ed.) Proceedings of the 2nd International Conference on Knowledge Engineering and Ontology Development (KEOD'09), pp. 5–12. INSTICC Press (2009)
10. Huyhn, D., Karger, D., Miller, R.: Exhibit: lightweight structured data publishing. ACM 978-1-59593-654-7/07/0005. MIT Computer Science and Artificial Intelligence Laboratory (2007)
11. Ide, N., Romary, L.: Outline of the international standard linguistic annotation framework. In: Proceedings of the ACL 2003 Workshop on Linguistic Annotation: Getting the Model Right (LingAnnot 03), vol. 19, pp. 1–5. Association for Computational Linguistics, Stroudsburg (2003)

12. Izquierdo, R., Suarez, A., Rigau, G.: Exploring the automatic selection of basic level concepts. In: Angelova, G., Bontcheva, K., Mitkov, R., Nicolov, N., Nikolov, N. (eds.) International Conference Recent Advances in Natural Language Processing, Borovets, pp. 298–302 (2007)
13. Kaiser, K., Miksch, S.: Information extraction a survey. Tech. rep., Vienna University of Technology. Institute of Software Technology and Interactive Systems (2005)
14. Majid, A., Boster, J.S., Bowerman, M.: The cross-linguistic categorization of everyday events: a study of cutting and breaking. Cognition **109**, 235–250 (2008)
15. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: Wonderweb deliverable d18: Ontology library. Tech. rep., ITSC-CNR, Trento (2003)
16. Mizoguchi, R., Sunagawa, E., Kozaki, K., Kitamura, Y.: The model of roles within an ontology development tool: hozo. Appl. Ontol. **2**, 159–179 (2007)
17. Niles, I., Pease, A.: Linking lexicons and ontologies: mapping wordnet to the suggested upper merged ontology. In: Proceedings of the 2003 International Conference on Information and Knowledge Engineering (IKE 03), Las Vegas, pp. 23–26. CSREA Press, Las Vegas (2003)
18. Peshkin, L., Pfeffer, A.: Bayesian information extraction network. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03), pp. 421–426. Morgan Kaufmann, San Francisco (2003)
19. Pustejovsky, J., Lee, K., Bunt, H., Romary, L.: Iso-timeml: an international standard for semantic annotation. In: Chair, N.C.C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J. Piperidis, S., Rosner, M., Tapias, D. (eds.) Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta (2010)
20. Putnam, H.: The meaning of 'meaning'. Minn. Stud. Philos. Sci. **7**, 131–193 (1975)
21. Vossen, P., Rigau, G.: Division of semantic labor in the global wordnet grid. In: Proceedings of Global WordNet Conference (GWC'2010), Mumbay (2010)
22. Vossen, P., Agirre, E., Calzolari, N., Fellbaum, C., Shu-Kai Hsieh, Chu-Ren Huang, Isahara, H., Kanzaki, K., Marchetti, A., Monachini, M., Neri, F., Raffaelli, R., Rigau, G., Tesconi, M., VanGent, J.: KYOTO: a system for mining, structuring, and distributing knowledge across languages and cultures. In: Proceedings of the 4th Global WordNet Conference (GWC'08), University of Szeged. Szeged, Hungary (2008)

Chapter 6
Anchoring Background Knowledge to Rich Multimedia Contexts in the KNOWLEDGESTORE

R. Cattoni, F. Corcoglioniti, C. Girardi, B. Magnini, L. Serafini, and R. Zanoli

Abstract The recent achievements in Natural Language Processing in terms of scalability and performance, and the large availability of background knowledge within the Semantic Web and the Linked Open Data initiative, encourage researchers in doing a further step towards the creation of machines capable of understanding multimedia documents by exploiting background knowledge. To pursue this direction it turns out to be necessary to maintain a clear link between knowledge and the documents containing it. This is achieved in the KNOWLEDGESTORE, a scalable content management system that supports the tight integration and storage of multimedia resources and background and extracted knowledge. Integration is done by (i) identifying mentions of named entities in multimedia resources, (ii) establishing mention coreference and either (iii) linking mentions to entities in the background knowledge, or (iv) extending that knowledge with new entities. We present the KNOWLEDGESTORE and describe its use in creating a large scale repository of knowledge and multimedia resources in the Italian Trentino region, whose interlinking allows us to explore advanced tasks such as entity-based search and semantic enrichment.

R. Cattoni (✉) · C. Girardi · B. Magnini · L. Serafini · R. Zanoli
Fondazione Bruno Kessler, Via Sommarive 18, 38123, Trento, Italy
e-mail: cattoni@fbk.eu; cgirardi@fbk.eu; magnini@fbk.eu; serafini@fbk.eu,zanoli@fbk.eu

F. Corcoglioniti
Fondazione Bruno Kessler, Via Sommarive 18, 38123, Trento, Italy
e-mail: corcoglio@fbk.eu

DISI - University of Trento, Via Sommarive 14, 38123, Trento, Italy

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_6, © Springer-Verlag Berlin Heidelberg 2013

6.1 Introduction

The availability of multimedia digital documents is exponentially increasing, and natural language, speech, and image processing technologies are mature enough to support large scale extraction of knowledge about various kinds of entities (e.g., persons, organizations, locations). The Semantic Web and the Linked Open Data initiatives, on the other side, are making available an increasingly large amount of knowledge resources, about many disparate domains, under the form of populated ontologies, and a set of scalable systems that support a flexible access and efficient reasoning services on this knowledge. Recently, it became clear that joining these two important achievements will provide a great advantage in multimedia processing, i.e., that the usage of large available knowledge bases can sensibly improve the understanding and processing of various types of media. Some examples of initial steps in this direction are described in [3, 8, 14, 18].

In this chapter we propose a further step in this direction by presenting an architecture and a prototype called KNOWLEDGESTORE, especially designed to support multimedia knowledge extraction and integration with the help of existing knowledge. First notice that this complex task involves two types of knowledge: on the one hand, knowledge automatically extracted from multimedia resources, which we call *corpus-induced knowledge*, is the result of the process of searching, collecting, and clustering various fragments of evidences (a.k.a. mentions) of such knowledge that occur in multimedia documents; on the other hand, the knowledge already available under some ontological resource, which we call *background knowledge*, is the result of somebody publishing such resources in the format of the Semantic Web or Linked Open Data. Although integrating corpus-induced and background knowledge turns out to be the crucial point, so far, knowledge extraction and reasoning about knowledge have mainly been investigated separately, with different solutions developed in different fields. In the Natural Language Processing community (NLP), tagging systems have been developed to semantically annotate multimedia resources (i.e., text, images, videos) and extract corpus-induced knowledge, with the focus on the extraction process. In the Knowledge Representation (KR) and Semantic Web communities, knowledge bases have been developed to store and manage large amounts of knowledge, but with limited linguistic information. Therefore we need a novel architecture that supports the seamless integration of both kinds of knowledge. We believe, indeed, that exploiting the relation between the two types of knowledge is fundamental for future improvements, as it allows for novel applications and the exploiting of the growing amount of data being published on the (Semantic) Web.

We present our ongoing work in developing the KNOWLEDGESTORE, a scalable content management system that stores multimedia resources, background knowledge, and all the intermediate results produced by the NLP and KR tools used in the process of interpreting a resource content and linking it to background knowledge. The KNOWLEDGESTORE builds on state-of-the-art tagging systems for extracting corpus-induced knowledge, and permits to integrate it with background

knowledge consisting of annotated RDF entity descriptions. Integration is performed by (i) extracting mentions of named entities from resources, (ii) establishing mention coreference and either (iii) linking mentions to entities in the background knowledge, or (iv) extending that knowledge adding new entities.

The KNOWLEDGESTORE builds on the following inspiring principles:

- *Scalability*. As large multimedia collections and knowledge resources are becoming widespread and publicly accessible on the Web, scalability with respect to the size of managed contents is a crucial matter.
- *Traceability*. Through the use of rich metadata, stored information should be traced back to its location in the original information sources, so to guarantee the proper use and exploitation of information contents.
- *Incrementality*. At any moment, it should be possible to add (or remove) information sources and rely to the system for the proper merging of new contents with existing ones, without the need to re-process all the stored information.
- *Contextualization*. As information in multimedia resources (e.g., the fact that "Barack Obama" is the President of USA) may be valid only in the context of that resources (e.g., in 2012), it is crucial to detect these contexts and maintain and formalize them when extracting and integrating knowledge.

Thanks to the explicit representation and alignment of semantic information at different levels—from annotated multimedia resources to RDF entity descriptions—the design of the KNOWLEDGESTORE enables advanced applications combining knowledge and multimedia, and provides the ideal settings for the empirical investigation of several tasks which are difficult to experiment otherwise. One of those tasks is *ontology population*, for which the KNOWLEDGESTORE allows investigating the mechanisms underlying *knowledge crystallization*, i.e., the process through which information from a stream of multimedia documents is automatically extracted, compared, and finally integrated into the background knowledge. Knowledge crystallization is particularly challenging as it involves the temporal dimension, an aspect which is almost untouched in current research on ontology population. Another supported task is *knowledge fusion*, i.e., the merging of possibly contradicting information extracted from different sources (e.g., different Web sites or news articles). To that respect, the resolution and the explicit representation of cross-media (i.e., text, images, video) and cross-document coreferences allows the exploration of a number of computational strategies for knowledge fusion.

The KNOWLEDGESTORE has been concretely used in the LiveMemories project[1] to store, process and interlink large amounts of multimedia documents and ontological knowledge about the Italian Trentino region, building a comprehensive repository of the knowledge and the digital memories of this area. In this context, two applications that build on the interlinking of knowledge and multimedia in the KNOWLEDGESTORE have been investigated: *entity-based search* and *semantic enrichment*. The first exploits extracted knowledge to find entities matching some

[1] LiveMemories (Active Digital Memories of Collective Life—http://www.livememories.org).

criteria, and then returns the multimedia resources mentioning those entities. The second aims at discovering which additional relevant knowledge the system can add to a given mention so to ease a user's understanding of a multimedia resource; as such, it requires to explicitly take into consideration the contextual aspect, as only knowledge valid or relevant in the mention context should be included.

This chapter is organized as follows. Section 6.2 briefly describes the context and the initiatives related to our work. Section 6.3 presents the KNOWLEDGESTORE approach and overall architecture, while Sect. 6.4 focuses on the implementation. Section 6.5 reports on the LiveMemories experiences and Sect. 6.6 concludes.

6.2 State of the Art

Our research takes advantage of previous and ongoing experiences both in Open Information Extraction and in the Semantic Web area, including the Linked Open Data initiative.[2]

As for Open Information Extraction (OIE), although several linguistic taggers are available for a number of languages (e.g., OpenCalais,[3] Gate[4]) and standards have been developed to represent linguistic information (e.g., Conll[5] and ACE[6]), still there are no attempts to systematically integrate such annotated data with background knowledge. A significant example of state-of-art OIE system is TextRunner [7]. While it mines and stores annotated data on a very large-scale (actually, much larger than what we experimented with the KNOWLEDGESTORE), it does not address the crucial issue of linking such triples to existing background knowledge, which limits the re-usability of such data. As a matter of fact, much more attention has been paid on the processing side (e.g., several taggers are offered as web services) rather than on the storage side. The focus has been mainly on formats for single tasks, without a clear overall design, with the consequence that the interaction between linguistic, semantic, and world knowledge is still underspecified and poorly investigated. In this direction, the KNOWLEDGESTORE is intended to exploit the benefits of a common place for representing linguistic, semantic, and world knowledge on a large-scale.

As for architecture, a popular annotation framework is the Unstructured Information Management Architecture (UIMA)[7] developed by IBM and used as architectural basis for several NLP systems, including the recent Watson system. UIMA enables applications to be decomposed into components via XML descriptor

[2] http://linkeddata.org/

[3] http://www.opencalais.com/

[4] http://gate.ac.uk/

[5] http://www.cnts.ua.ac.be/conll2002/ner/

[6] http://www.itl.nist.gov/iad/mig/tests/ace/

[7] http://uima.apache.org/

files and provides a general metadata schema (called CAS), although there is no specific reference to the representation of large amounts of knowledge. A recent initiative which shares some of the motivations with the KNOWLEDGESTORE is the NLP Interchange Format (NIF).[8] NIF is an RDF/OWL-based format that aims at achieving interoperability between NLP tools, language resources and annotations. Its core consists of a vocabulary for representing strings as RDF resources. A special URI design is used to pinpoint annotations to a part of a document, which can be published on the Web as Linked Data and interchanged between different NLP tools and applications.

As for the Semantic Web community and the Linked Open Data initiative, a significant research effort has gone along the direction of supporting the interlinking of knowledge and text documents at the representation and publication levels. This includes the standardization of metadata ontologies for describing generic information resources (e.g., the Dublin Core Metadata Standard[9]) and multimedia contents [19], and the deployment of mechanisms such as GRDDL [6] and RDFa [15] for embedding RDF statements in XML and HTML documents, and HTTP content negotiation for relating the RDF description of an entity with its corresponding human-readable unstructured representation. Also relevant are the attempts to extract Semantic Web data from existing semi-structured resources, among which Wikipedia plays a central role. Both the DBPedia [2] and Yago [10] datasets are extracted from Wikipedia, but they focus on its structured part (categories and infoboxes) and largely ignore page texts. Aiming at bridging the gap between the Web of documents and the Web of Data, several web services have been developed for recognizing mentions of named entities in an input text and linking them to URIs in relevant Linked Data knowledge resources, such as DBPedia Spotlight [13], Zemanta[10] and AlchemyAPI.[11] Most of these services rely on the direct or indirect link between Web of Data entities and corresponding Wikipedia pages for disambiguation. Finally, it is worth noting that also in the Semantic Web community little attention has been paid to the storage of semantic data interlinked with multimedia resources. While triple stores have evolved into scalable solutions for storing, querying and reasoning with large amounts of knowledge, they currently provide only a limited support for integrating knowledge with multimedia, often consisting in simple full text search capabilities on RDF literals.

[8] http://nlp2rdf.org/nif-1-0

[9] http://dublincore.org/documents/dces/

[10] http://www.zemanta.com

[11] http://www.alchemyapi.com

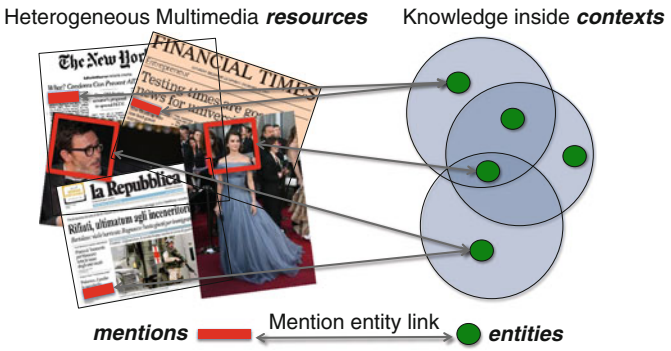

Fig. 6.1 Relating resources, mentions, entities and contexts in the KNOWLEDGESTORE

6.3 The KNOWLEDGESTORE Approach

This section presents the KNOWLEDGESTORE approach to knowledge extraction and integration. Central to the approach are the four key concepts illustrated in Fig. 6.1: *resource*, *mention*, *entity* and *context*. Each concept corresponds to a representation layer of the system, with the layer sequence mirroring the knowledge extraction process (from resources to mentions and then to entities and context). The representation layers are the focus of Sect. 6.3.1, while the KNOWLEDGESTORE approach for processing information contents is described in Sect. 6.3.2.

6.3.1 Representation Layers

The four KNOWLEDGESTORE representation layers—resources, mentions, entities and contexts—are shown in the UML class diagram of Fig. 6.2 and described next.

Resources. A resource is a multimedia physical file from which to extract knowledge, or evidences of knowledge. Examples of resources are texts, images, audios and videos (or portions of them), as well as the files derived from their processing, such as Automatic Speech Recognition transcriptions. Each resource is stored in its raw format and is described with a set of metadata attributes (see Fig. 6.2), most of which coming from the Dublin Core (DC) standard. Resources can be related one to another through several relations so that a graph of resources is actually stored. The relation *partOf* stores the relation between a complex resource and the parts in which it has been split (e.g., pictures which are parts of an article). The relation *captionOf* connects the text of the caption of a picture with the picture itself. The relation *from*, represents the fact that a certain resource has been obtained by preprocessing some other resource (e.g., the speech transcription from a video). Any other relation between resources is stored under the generic relation *relatedTo*.

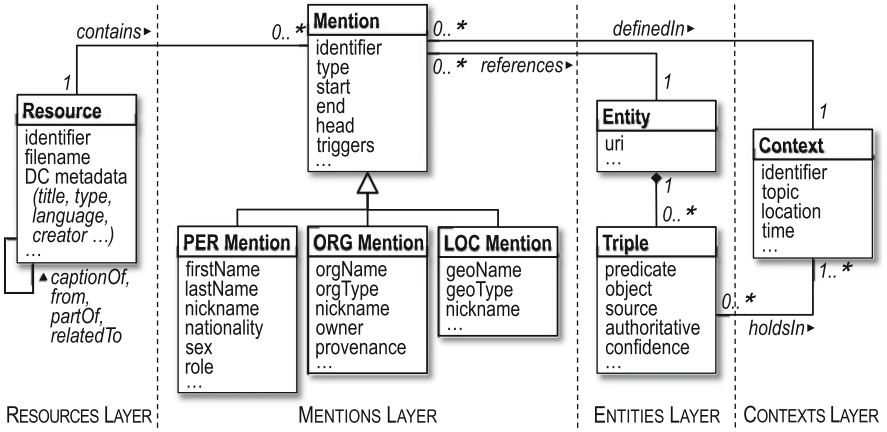

Fig. 6.2 The KNOWLEDGESTORE representation layers (only the most relevant attributes are shown)

Mentions. A mention is a portion of a resource referring to (i.e., mentioning) a real world object. Examples include mentions of persons (PER), organizations (ORG) and geo-political entities/locations (GPE/LOC). For instance, a PER mention in a text is a fragment such as "President Barack Obama", while a PER mention in a picture is the area of the picture where the person is depicted. Mentions are a kind of hybrid objects as, on the one side, they are portions of the original resource, while, on the other side, the text of each mention conveys some information (i.e., it is a knowledge evidence) about some property of the mentioned object. This knowledge is extracted from the text and, in particular, from certain kinds of words called *triggers* that appear immediately before or after a mention. As an example, for the mention "President Barack Obama" we may add the facts that it represents a *Person*, whose *firstName* is "Barack", whose *lastName* is "Obama" and whose *role* is "President" (trigger word). As shown in Fig. 6.2, mentions are described with metadata attributes, some of which are independent of the particular type of mention (e.g., attributes *start*, *end* and *head* that locate a mention in a text) while others are type-specific; the latter mainly encode extracted semantic information and are inspired to the ACE program. A mention is defined in the context of the resource it occurs in and, as a consequence, its semantic annotations hold only in that context. In order to ease coreference, we currently restrict to mentions of named entities, i.e., mentions denoting a proper name (e.g., "Barack Obama"); nevertheless, the KNOWLEDGESTORE model is general and can accommodate also for pronoun mentions.

Entities. An entity is a media-independent, abstract representation of an object of a certain type (PER, ORG, GPE/LOC). Following the Semantic Web approach, an entity is identified by a URI and described using an unrestricted set of ⟨*subject,*

predicate, object⟩[12] RDF triples, whose subject is the entity URI while the predicate and object define an atomic *fact* about the entity, consisting in an attribute (e.g., that the entity *firstName* is "Obama"), a category (e.g., that Obama is a politician) or a relationship with another entity (e.g., that Obama is president of the USA). A many-to-one relation holds between mentions and entities, as different mentions may refer to the same entity; for instance, mentions "Barack Obama" and "President Obama", although different, may denote the same PER entity. Under this view, entities are supposed to be resource-independent object representations, such that multiple occurrences of the same piece of information are encoded by a single triple, no matter how many resources express this information. In the KNOWLEDGESTORE, entities and their triples originate from trusted background knowledge or from knowledge extracted from mentions. For this reason, each triple can be associated to rich metadata covering aspects such as the triple provenance, authoritativeness and reliability (*source*, *authoritative* and *confidence* attributes in Fig. 6.2).

Contexts. A context delimits the circumstances within which some piece of knowledge holds. Contexts play a central role in the KNOWLEDGESTORE, as information in a resource may be valid only in that resource context, which needs to be considered when performing integration. For example, a 2012 news article may refer to Barack Obama as the USA President, while an article from 2007 may state that he is a Senator; on the other hand, background knowledge may provide time-independent information about Barack Obama, such as his birth date and place. Simply merging these pieces of knowledge without considering where they are stated is not sufficient, as it leads to loss of information and inconsistencies, which arise, for instance, by ignoring the two articles dates thus describing Barack Obama as both the President and a Senator of the USA. It follows that knowledge extraction and integration must explicitly take context into consideration. As shown in Fig. 6.2, this is achieved in the KNOWLEDGESTORE by explicitly representing contexts as ⟨*topic, location, time*⟩ tuples, following the *Contextualized Knowledge Repository* (CKR) model [11]. Contexts are then associated to entity triples, to denote their validity scope (e.g., to encode that Barack Obama is the USA President only in a specific time period), and to mentions, to denote the circumstances where information extracted from them hold. Based on the CKR model, contexts are implicitly organized in a broader-narrower hierarchy based on their tuple values, so that, for instance, context ⟨*Politics, World, 2008–2012*⟩ is broader than context ⟨*Politics, USA, 2012*⟩. This structure plays an important role in organizing and integrating knowledge, as it provides the basis for relating and propagating knowledge across contexts.

[12] *Subject*, *predicate* and *object* are the standard terms denoting the components of a triple in the Semantic Web literature: although they are named after the components of a natural language sentence, they convey no linguistic semantics.

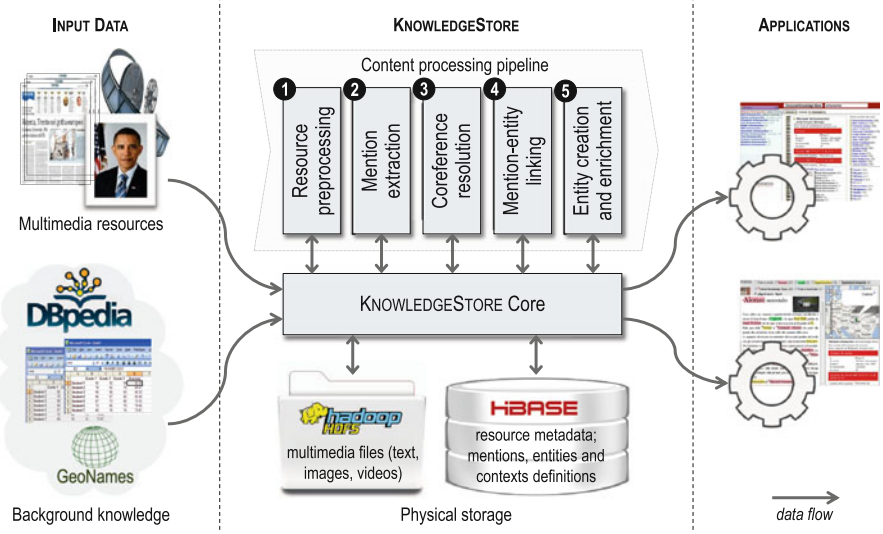

Fig. 6.3 The KNOWLEDGESTORE approach for processing information content

6.3.2 *Content Processing*

Resources, mentions, entities and contexts are stored and processed as shown in Fig. 6.3. The central part of the figure depicts the architecture of a KNOWLEDGESTORE instance, which consists of a *core* component and a *content processing pipeline*. The first provides a scalable, joint storage for multimedia and knowledge, building on established standards from NLP (e.g., Conll and ACE) and Semantic Web (RDF for entities and Named Graphs [5] for contexts).

The pipeline is composed of modules that interface with the core to extract and integrate knowledge from those contents. A KNOWLEDGESTORE instance is fed with raw multimedia resources and background knowledge (left side of Fig. 6.3), which populate the resources, entities and contexts representation layers of the system. The pipeline is then activated to process those contents, by calling its modules in cascade, each of them reading its inputs from the KNOWLEDGESTORE core and writing back its results to the same component. The following processing steps are performed:

1. *Resource preprocessing*. Resources are preprocessed so to ease or enable further elaboration; this involves conversion to common data formats, segmentation of complex resources, automatic speech recognition and linguistic tagging.
2. *Mention extraction*. Named entity recognition is performed to extract mentions of named entities together with their recognized contexts and semantic annotations.
3. *Coreference resolution*. Cross-document coreference resolution is performed on extracted mentions so to identify and cluster together mentions that refer to the same real world entity, e.g., to tell whether the "philosopher John Smith" in

a resource is the same as the "J. Smith" in another resource. Cross-document coreference resolution is a clustering task, made more difficult by the fact that coreferring mentions may appear in different—and often dissimilar—resources.

4. *Mention–entity linking*. Clusters of coreferring mentions are linked to the corresponding entities in the background knowledge, if any, thus effectively establishing a link between knowledge and multimedia. This central task suffers of the ambiguity problem, as multiple entities may have a name compatible with the surface form of a mention. Ambiguity is addressed in the KNOWLEDGESTORE leveraging on the contextual organization of knowledge, by matching the contexts of mentions and entities in addition to their respective attributes.
5. *Entity creation and enrichment*. New entities are created starting from unlinked mention clusters, as they denote entities whose existence is unknown in the background knowledge but can be inferred from resources stored in the KNOWLEDGESTORE. Information extracted from mentions is then used to enrich entity descriptions with new triples (this is currently restricted to entity name triples).

The process results in the extraction and storage of relevant knowledge from multimedia resources, in the storage of intermediate results produced by the tools in the pipeline and in the interlinking of resources and entities through mentions. Overall, this allows external applications (right side of Fig. 6.3) to effectively access stored contents, navigating from multimedia resources to corresponding knowledge and back.

6.4 System Implementation

The section presents the inside of the KNOWLEDGESTORE from the implementation point of view. Taking into account the overall view shown in Fig. 6.3, the main software components are described with some details. While the focus is on the processing of Italian texts, the presented components can be configured and/or trained to successfully work with other languages, as discussed next. Section 6.4.1 focuses on the KNOWLEDGESTORE *core*, while the five steps of the *content processing pipeline* are described in Sects. 6.4.2–6.4.6.

6.4.1 KNOWLEDGESTORE *Core*

The KNOWLEDGESTORE core provides the storage for the four representation layers of Sect. 6.3.1, building on top of the Hadoop[13] and Hbase[14] frameworks. Distributed computation on multiple nodes and fault tolerance with respect to single node failure

[13] http://hadoop.apache.org

[14] http://hbase.apache.org

are key features of such frameworks and provide the KNOWLEDGESTORE with massive scalability. The Hadoop distributed file system stores raw resources, while Hbase is used as a database to store the remaining information, with four specialized tables encoding resource metadata, mentions, entities and contexts.

For each representation layer of the system, the KNOWLEDGESTORE core exposes a set of web services to manage the corresponding data, implemented in Tomcat 5.5[15] on top of the Java Servlet framework. Web services include the standard CRUD (create, read, update, delete) operations and search/query capabilities. For each type of object (resource, mention, entity, context) the web services allow users to store, delete, update and retrieve its instances. The search web services allow users to perform queries—whose fields correspond to the object attributes—by means of a SQL-like language. Although Hbase is an appropriate backend for the KNOWLEDGESTORE, as it has been specifically designed for the management of huge data with sparse attribute values, it does not directly support SQL-like languages, so the HBQL package[16] has been used to overcome this limitation.

6.4.2 Resource Preprocessing

Resource preprocessing—the first step of the content processing pipeline—is implemented by transforming input resources with a number of pluggable modules and storing derived resources back in the KNOWLEDGESTORE. Preprocessing modules can be selected and configured based on the characteristics of the data loaded in a KNOWLEDGESTORE instance. Four types of modules are supported:

- *Format converters* encode resources in common data formats;
- *Segmenters* split complex resources in their components and extract inter-resource relationships among them, e.g., by separating individual stories in a news broadcast or by extracting text, figures and captions from a complex XML news article;
- *Automatic speech recognizers* extract annotated speech transcriptions from audio resources, producing text that can be processed more effectively in the pipeline;
- *Linguistic taggers* annotate resources with linguistic features that ease further processing; they include part of speech tagging, lemmatization, morphological analysis and temporal expression recognition and normalization, all based on the TextPro suite [17], and key concept extraction, based on the KX tool [16].

[15] http://tomcat.apache.org

[16] http://www.hbql.com

Table 6.1 TextPro annotation for the Italian text "Luca di Montezemolo" in the Conll format

Token	Start position	End of sentence	POS	Lemma	Entity
Luca	0	–	SPN	Luca	B-PER
di	5	–	E	di	I-PER
Montezemolo	8	<eos>	SPN	Montezemolo	I-PER

6.4.3 Mention Extraction

Mentions of PER, ORG and GPE/LOC named entities in textual resources are identified, classified and annotated using the TextPro Mention Detection module. Like other modules in TextPro, mention detection involves the supervised training of a statistical model. The training requires an annotated corpus and can be performed for any language for which such a corpus is available. The system comes pre-packaged with a model for the Italian language produced by training the system on the Italian Content Annotation Bank (I-CAB) [12]; concerning the accuracy with this model, the authors report a F1 value of 82 %.

Table 6.1 reports an example of TextPro output for the text "Luca di Montezemolo", in the Conll standard format. TextPro segments the text in sentences; for each sentence, mentions are marked as shown in the last column: the entity type is reported (e.g., PER) together with a letter flag distinguishing the first token ('B' as begin) from the other tokens ('I' as inside) of a mention.

TextPro has been also configured to extract mention *triggers*, which are then processed in order to annotate mentions with semantic attributes such as a person's role and sex. A rule based approach has been implemented to match the most extended trigger of a mention given a predefined list of possible trigger words, that for the Italian language consists of 8,267 roles, 510 nationalities, 87 political and 28 religious affiliations. For example, given the following sentence: "Oggi il presidente della Ferrari Luca di Montezemolo si è detto soddisfatto" ("Today, Ferrari president Luca di Montezemolo declared to be pleased"), the fragment "presidente della Ferrari" ("Ferrari president") is recognized to be a trigger by rule `[role trigger] [det. preposition] [ORG]`; based on that rule, two semantic annotations expressing the role and affiliation of the mentioned person are derived and stored.

6.4.4 Coreference Resolution

Cross-document coreference resolution is performed using two distinct, specialized systems, one for PER and ORG mentions and one for GPE/LOC mentions.

Mentions of PER and ORG named entities are coreferred using the JQT2 system [21], which is based on the Quality Threshold (QT) clustering algorithm [9]. The distinguishing feature of JQT2 is the use of a dynamic similarity threshold.

Table 6.2 Bcubed precision, recall, and F1 measure for different levels of ambiguity of a name: no ambiguity, medium ambiguity and high ambiguity. The *all-in-one* baseline of the NePS task consists in grouping all the mentions sharing the same superficial form in the same cluster

	All names			No ambiguity			Med. ambiguity			High ambiguity		
Algorithm	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
JQT2	0.89	0.97	0.93	1.00	0.99	0.99	0.89	0.95	0.92	0.71	0.96	0.82
ALL-IN-ONE	0.84	1.00	0.91	1.00	1.00	1.00	0.86	1.00	0.93	0.56	1.00	0.72

In coreference systems, the similarity threshold determines how close two elements—either the resources containing the coreferred mentions or two mention clusters—have to be so to be clustered together assuming coreference: the lower the value, the less the evidence required by the system to assume coreference. Although a global threshold value is commonly used, for optimal performances the threshold value should depend on the ambiguity of the coreferred name. For example, the Italian name "Luca Cordero di Montezemolo" is uncommon and non-ambiguous, hence a low threshold should be applied requiring less evidence for coreference; differently, "Paolo Rossi" is an Italian common and ambiguous name, so the chance that many different persons carry this name is high and a higher threshold should be used. This approach is implemented in JQT2, whose dynamic threshold adapts its value to the ambiguity of the coreferred name, estimated using a language-specific phone book. The improvement in accuracy is shown in Table 6.2, where JQT2 accuracy on PER names has been evaluated on the Cross-document Italian People Coreference (CRIPCO) corpus [1] as part of the News People Search (NePS) task at Evalita 2011.[17] Concerning the system annotation speed, common values are about 500 mentions/s. As JTQ2 is based on unsupervised clustering, it can be ported quite easily to different languages for which a phone book is available.

Mentions of GPE/LOC entities are coreferred with Geocoder [4], a system designed for the coreference of ambiguous toponyms (e.g., "Cambridge" in UK or USA or "Alabama" as a river or a state) by using geometric features. Coreference in Geocoder is performed by linking mentions to well-known and geo-referenced toponyms in external resources (the GeoNames[18] geographical database and the Google Maps geo-referencing service[19] are used), thus resulting in coreferring mentions being associated to the same toponym. The linking of a mention to a toponym in the database is performed by considering other non-ambiguous GPE/LOC mentions co-occurring in the text: the chosen toponym is the one resulting to be *geographically nearer* to the toponyms associated to co-occurring mentions, weighted according to their frequencies and distance in the text from the linked mention.

[17] http://www.evalita.it/2011/tasks/NePS

[18] http://www.geonames.org/

[19] http://code.google.com/apis/maps/

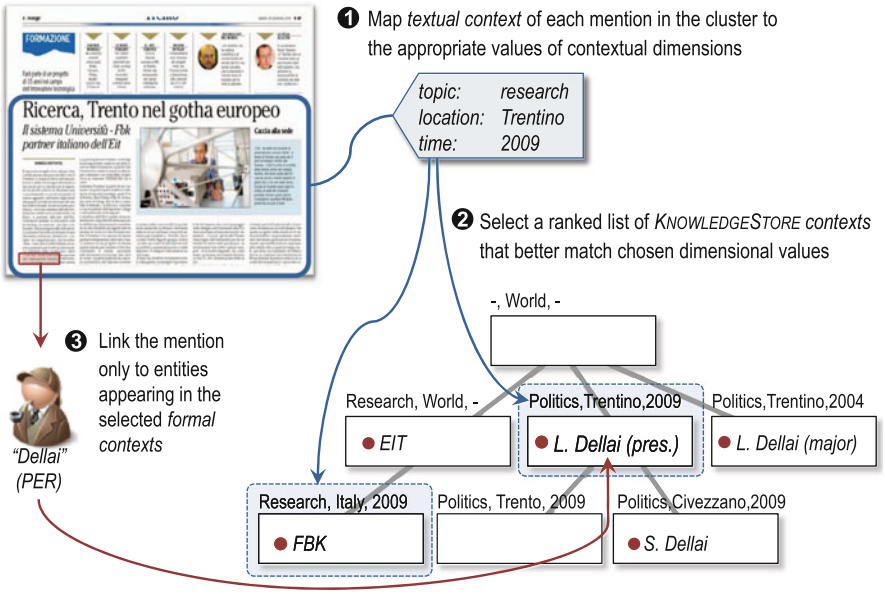

Fig. 6.4 Context-driven mention–entity linking algorithm

Both JQT2 and Geocoder allow for two operations: clustering the whole dataset from scratch and incremental clustering, which updates only the clusters affected by new data, coherently with the incrementality principle stated in Sect. 6.1.

6.4.5 Mention–Entity Linking

Mention–Entity linking is performed in different ways depending on how coreference has been resolved. In particular, linking of GPE/LOC mention clusters is performed indirectly by Geocoder, as it links mention clusters to GeoNames toponyms that, in turn, can be reasonably expected to be aligned to the GPE/LOC entities in the background knowledge (e.g., via `owl:sameAs` triples). Linking of remaining PER and ORG mention clusters is performed using a *context-driven linking algorithm* [20] that leverages the contextual organization of knowledge to address ambiguity, as shown in Fig. 6.4. The algorithm associates each mention in an input cluster to the values of the *topic*, *time* and *location* contextual dimensions that more closely reflect its *textual context*. The topic value is chosen based on the keywords automatically extracted from the mention texts, while time and location values are currently chosen based on resource metadata (e.g., creation time), although the use of automatically extracted spatial and temporal expressions is foreseen. Chosen dimensional values are used to select a ranked list of compatible

KNOWLEDGESTORE contexts. Only entities in these contexts are considered for linking, thus reducing the number of candidates and helping with disambiguation. Linking is performed by searching the selected contexts one by one for entities which are compatible with the surface forms, the textual contexts and the metadata associated to mentions. The process ends with success as soon as a matching entity is found, or with failure after having considered all the matching contexts.

The linking algorithm has been evaluated against a gold standard derived from the one used in Evalita 2011 for cross-document coreference. It consists of 21,273 documents with 22,511 PER mentions grouped in 298 clusters, 73 of which manually associated to corresponding background knowledge entities. The algorithm scored 84.5 % accuracy (i.e., the fraction of correctly linked or non-linked clusters).

The linking of mention clusters instead of single mentions permits to increase the accuracy, as more input data is available to the linking service. On the down side, the approach is vulnerable to coreference errors, which may be detected when an entity matching all the cluster mentions cannot be identified. For this reason, a more elaborated workflow is under investigation, with coreference and linking rearranged in a loop where the output of linking is used to refine the coreference decision.

6.4.6 Entity Creation and Enrichment

For each unlinked cluster a new entity is created and stored in the entity layer of the KNOWLEDGESTORE. A triple is stored to denote the name of the new entity, which is chosen by a *naming algorithm* based on the superficial forms of the mentions in the clusters; longer and frequently mentioned names are preferred.

In the future, we plan to aggregate mention attributes so to derive new triples, both for corpus-induced and background knowledge entities, investigating suitable algorithms for knowledge fusion.

6.5 Experiments and Results

In this section we report on our experience in using and experimenting with the KNOWLEDGESTORE in the scope of the *LiveMemories* project.

Within the project, a large scale KNOWLEDGESTORE instance has been created by collecting multimedia news and background knowledge relevant to the Italian Trentino region, effectively building a real multimodal archive of digital memories and public knowledge in Trentino. Section 6.5.1 describes the creation of the instance, demonstrating the effectiveness and scalability of the proposed approach.

The created instance has been used to experiment with a number of tasks and applications that build on the interlinking of multimedia and knowledge. Among them, we have focused on two interaction mechanisms that ease the users'

Table 6.3 Resources statistics (RTTR news reported after segmentation and speech transcription)

News provider		News	Images	Captions	Videos
l'Adige	http://www.ladige.it/	733,738	21,525	21,327	–
VitaTrentina	http://www.vitatrentina.it/	33,403	14,198	7,516	–
RTTR	http://www.rttr.it/	2,455	–	–	120 h
Fed. Coop.	http://www.ftcoop.it	1,402	–	–	–
Total		770,998	35,723	28,843	120 h

access to large content repositories: *entity-based search* and *contextualized semantic enrichment*. Sections 6.5.2 and 6.5.3 describe the two mechanisms and show the benefits of applying knowledge extraction and interlinking with multimedia on a large scale.

It is worth noting here that although the language of the multimedia resources is Italian, the techniques and methods implemented in the content processing pipeline work successfully with other languages (e.g., English), as discussed in Sect. 6.4.

6.5.1 KNOWLEDGESTORE *Population*

Following the approach described in Sect. 6.3.2, the KNOWLEDGESTORE instance has been initially loaded with ∼770 K multimedia resources and with background knowledge consisting in ∼30 K entities described by ∼350 K triples. Multimedia resources consist of written news and images from three daily and weekly local newspapers—l'Adige, VitaTrentina, Federazione Trentina della Cooperazione (Fed. Coop.)—as well as videos of daily television news from the local television RTTR, covering a time period from 1999 to 2011 overall. Background knowledge has been manually collected from several Web sources, including the Italian Wikipedia, sport-related community sites and the official Web sites of local and national-level public administrations and economic and government bodies (e.g., the Italian Parliament). Statistics about loaded resources are reported in Table 6.3, while Table 6.4 reports on the number of loaded contexts, entities and triples of background knowledge, aggregated along top-level topics.

The results of processing loaded contents with the KNOWLEDGESTORE pipeline are summarized in Table 6.5, with the detail on the number of mentions by type and news provider reported in Table 6.6. Mention extraction resulted in the identification of ∼12 M mentions, corresponding to ∼10 % of all the words in processed textual resources. Starting from these mentions, cross-document coreference resolution identified about 400 K distinct clusters. Only 10.74 % of these clusters were linked to entities in the background knowledge, a percentage that increases to 31.03 % in terms of mentions, indicating altogether that the most popular (and thus frequently mentioned) entities are present in the background knowledge. The large number of unlinked clusters gave rise to the creation of ∼390 K corpus-induced entities, corresponding to 92.76 % of all stored entities. This large percentage highlights

Table 6.4 Background knowledge statistics (when computing totals, the same triples and entities that appear in different contexts are counted only once)

Main topic	Contexts	PER entities	ORG entities	Avg. predicates per entity	Total triples
Sport	136	8,570	191	3.81	192,115
Culture	20	9,785	1	2.00	33,236
Justice	7	354	10	2.16	1,575
Economy	7	49	1,203	4.47	11,147
Education	6	850	82	2.35	3,573
Politics	535	8,402	319	4.64	98,780
Religion	3	1,391	0	1.67	12,855
Total	714	28,687	1,806	3.64	352,244

Table 6.5 Processing statistics

Entity type	Recognized mentions	Coreference clusters	Linked mentions (%)	Linked clusters (%)	Induced entities	Total entities
PER	5,566,174	340,147	22.36	5.03	323,026	351,713
ORG	3,230,007	16,649	12.02	7.96	15,323	17,129
GPE/LOC	3,224,539	52,478	65.04	48.64	52,478	52,478
Total	12,020,720	409,274	31.03	10.74	390,827	421,320

Table 6.6 Mentions statistics detailed by type and news provider

News provider	PER mentions	ORG mentions	GPE/LOC mentions	Total mentions
l'Adige	5,387,994	3,100,994	3,052,011	11,540,999
VitaTrentina	144,486	100,789	136,611	381,886
RTTR	19,290	15,493	27,404	62,187
Fed. Coop.	14,404	12,731	8,513	35,648

the limits of manually acquired background knowledge, which can only cover the most popular entities. It also suggests that there is a large potential for applying knowledge base population techniques on stored multimedia resources, in order to (semi-) automatically populate the remaining long tail of less popular entities. Concerning accuracy, the measures reported in Sects. 6.4.3–6.4.5 are roughly indicative of the performances of mention extraction, coreference resolution and mention-entity linking on the dataset considered here, as the evaluation data they derive from is a subset of the considered dataset.

6.5.2 Entity-Based Search

The linking of multimedia and knowledge in the KNOWLEDGESTORE enables powerful presentation mechanisms that improve the fruition of contents by users. The first mechanism we investigated is *entity-based search*, which permits to retrieve the multimedia resources related to a query entity, e.g., to retrieve (and rank)

Fig. 6.5 Example of entity-based search for query "Schumacher" in the *TrentinoMedia* application

all the documents mentioning person "Michael Schumacher". This differs from normal keyword based search as the query is an entity name and the user is provided with suitable mechanisms to disambiguate that name with respect to homonymous entities known to the system, leading to better precision and recall.

Figure 6.5 shows the implementation of this mechanism in the *TrentinoMedia* demonstrator, a Web application realized within the *LiveMemories* project and based on the KNOWLEDGESTORE instance presented in this section. Users submit queries consisting in an entity name and are presented with the list of matching entities from the KNOWLEDGESTORE, organized by type and distinguished with short labels generated from stored data; by selecting an entity, the user is presented with the list of associated documents and with a card reporting all the information about the entity known to the system, with highlighted the distinction between information coming from background knowledge (upper part) and information extracted from multimedia resources and thus possibly less reliable (lower part).

6.5.3 Contextualized Semantic Enrichment

Another content presentation mechanism enabled by the KNOWLEDGESTORE is the *contextualized semantic enrichment* of entity mentions. This mechanism aims at

Fig. 6.6 Example of enrichments of mentions "Michael Schumacher" and "Ungheria" (Hungary) in a news article with context ⟨*World, 02 Aug 2010, Formula 1*⟩, from the *TrentinoMedia* application

improving the users' understanding of information resources, by allowing them to expand an entity mention in a displayed resource and gain access to related available information. The enrichment is contextualized as only the knowledge about an entity which is valid in the particular *context* of the resource is presented, limiting the risk of information overload that may arise if all the available information is shown.

Figure 6.6 presents a couple of examples of enrichment from the *TrentinoMedia* application. While reading a 2010 news article about Formula 1, the mentions of named entities known by the system are highlighted. By selecting a PER mention as "Michael Schumacher", the user is presented with a card reporting all the facts about Schumacher that were true in the context of the article, e.g., that he was a pilot of team Mercedes in 2010. By selecting a GPE/LOC entity, the geographical coordinates stored in the KNOWLEDGESTORE are used to locate the entity on a map. In addition, links to external resources stored in the system, such as the Wikipedia page associated to the entity, are exploited to present additional information to users.

Entity-based search and contextualized semantic enrichment are complementary mechanisms that exploit the interlinking between knowledge and multimedia: the first helps users in finding contents by navigating KNOWLEDGESTORE links from knowledge (entities) to multimedia resources; the second helps in understanding the

contents found by navigating links in the opposite direction, i.e., from multimedia resources to knowledge.

6.6 Conclusions and Future Work

We presented the KNOWLEDGESTORE, a system for storing and interlinking heterogeneous multimedia resources with automatically extracted semantic information and contextualized background knowledge. The KNOWLEDGESTORE builds on top of two ingredients: (i) a scalable and metadata-rich *storage model*, organized along four representation layers (i.e. resources, mentions, entities and contexts); and (ii) a *content processing pipeline* with state-of-the-art tools orchestrated into a workflow for extracting knowledge from multimedia and integrating it with background knowledge. Combined together in the KNOWLEDGESTORE, they enable the construction of large-scale linguistico-semantic resources, which provide an ideal setting for developing knowledge extraction and fusion techniques, for investigating the use of background knowledge in NLP tasks and for experimenting with solutions centered on the interlinking of knowledge and multimedia.

Future work will address both the storage model and the content processing pipeline. On the storage side, we plan to strengthen the integration with the Semantic Web—currently limited to the import and export of RDF/OWL background knowledge—by providing suitable mechanisms to expose data in the KNOWLEDGESTORE as Linked Data and to link it to existing Linked Data resources. In particular, we are currently experimenting with the use of the WikiMachine[20] tool to link KNOWLEDGESTORE entities to Wikipedia pages and therefore to DBPedia resources. On the content processing side, the goal is to improve the pipeline along three directions. First, we plan to extend the knowledge extraction capabilities of the KNOWLEDGESTORE—currently limited to the induction and labeling of new entities—by supporting the extraction and crystallization of knowledge triples from attributes extracted at mention level. Second, we plan to rearrange coreference and mention–entity linking in a loop, so that the output of one can be used to refine the decision of the other. Finally, we aim at completing the support for the incremental processing of data, currently restricted to few components of the pipeline, so that multimedia resources and background knowledge can be added to (or retracted from) the KNOWLEDGESTORE dynamically and efficiently.

Acknowledgements This work was supported by the LiveMemories project (Active Digital Memories of Collective Life) funded by Autonomous Province of Trento (Italy).

[20] http://thewikimachine.fbk.eu

References

1. Bentivogli, L., Girardi, C., Pianta, E.: Creating a gold standard for person cross-document coreference resolution in Italian news. In: Proceedings of LREC '08 Workshop on Resources and Evaluation for Identity Matching, Entity Resolution and Entity Management, Marrakech (2008)
2. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia – a crystallization point for the web of data. J. Web Semant. **7**(3), 154–165 (2009)
3. Bryl, V., Giuliano, C., Serafini, L., Tymoshenko, K.: Using background knowledge to support coreference resolution. In: Proceedings of 19th European: Conference on Artificial Intelligence, ECAI '10, pp. 759–764. IOS Press, Amsterdam (2010)
4. Buscaldi, D., Magnini, B.: Grounding toponyms in an Italian local news corpus. In: Proceedings of 6th Workshop on Geographic Information Retrieval, GIR '10, Zurich, pp. 15:1–15:5. ACM, New York (2010)
5. Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs, provenance and trust. In: Proceedings of 14th International Conference on World Wide Web, WWW '05, Chiba, pp. 613–622. ACM, New York (2005)
6. Connolly, D.: Gleaning resource descriptions from dialects of languages (GRDDL). W3C recommendation, W3C (2007). http://www.w3.org/TR/2007/REC-grddl-20070911/
7. Etzioni, O., Fader, A., Christensen, J., Soderland Mausam, S.: Open information extraction: the second generation. In: Proceedings of 22nd International Joint Conference on Artificial Intelligence, Barcelona, Lisbon, Portugal, pp. 3–10. IJCAI'11, Menlo Park (2011)
8. Ghosh, S., Shankar, N., Owre, S.: Machine reading using Markov logic networks for collective probabilistic inference. In: Proceedings of ECML-PKDD Workshop on Collective Learning and Inference on Structured Data, CoLISD '11 (2011). http://www.cse.iitm.ac.in/CoLISD/2011/CoLISD.html
9. Heyer, L.J., Kruglyak, S., Yooseph, S.: Exploring expression data: identification and analysis of coexpressed genes. Genome Res. **9**(11), 1106–1115 (1999)
10. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: a spatially and temporally enhanced knowledge base from Wikipedia. Artificial Intelligence – Special Issue: Artificial Intelligence, Wikipedia and Semi-Structured Resources (2012)
11. Homola, M., Serafini, L.: Contextualized knowledge repositories for the semantic web. Web Semant. Sci. Serv. Agents Worldw. Web **12**, 64–87 (2012)
12. Magnini, B., Pianta, E., Girardi, C., Negri, M., Romano, L., Speranza, M., Bartalesi Lenzi, V., Sprugnoli, R.: I-CAB: the Italian content annotation bank. In: Proceedings of 5th International Conference on Language Resources and Evaluation, LREC '06, Genova (2006)
13. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBPedia Spotlight: shedding light on the web of documents. In: Proceedings of 7th International Conference on Semantic Systems, pp. 1–8. ACM, New York (2011)
14. Oltramari, A., Lebiere, C.: Extending cognitive architectures with semantic resources. In: Proceedings of 4th International Conference on Artificial General Intelligence, AGI '11, Mountain View, pp. 222–231. Springer, Berlin (2011)
15. Pemberton, S., Adida, B., McCarron, S., Birbeck, M.: RDFa in XHTML: syntax and processing. W3C recommendation (2008). http://www.w3.org/TR/2008/REC-rdfa-syntax-20081014
16. Pianta, E., Tonelli, S.: KX: a flexible system for keyphrase extraction. In: Proceedings of 5th International Workshop on Semantic Evaluation, SemEval '10, pp. 170–173, Uppsala (2010)
17. Pianta, E., Girardi, C., Zanoli, R.: The TextPro tool suite. In: Proceedings of 6th International Conference on Language Resources and Evaluation, LREC '08. ELRA, Marrakech (2008)
18. Rahman, A., Ng, V.: Coreference resolution with world knowledge. In: Proceedings of 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies – Volume 1, ACL-HLT '11, Portland, pp. 814–824. ACL, Portland (2011)

19. Suãrez-Figueroa, M.C., Atemezing, G.A., Corcho, O.: The landscape of multimedia ontologies in the last decade. Multimed. Tools Appl. **55**(3), 1–23 (2011). http://link.springer.com/article/10.1007/s11042-011-0905-z?null
20. Tamilin, A., Magnini, B., Serafini, L.: Leveraging entity linking by contextualized background knowledge: a case study for news domain in Italian. In: Proceedings of 6th Workshop on Semantic Web Applications and Perspectives, SWAP '10 (2010). http://www.inf.unibz.it/krdb/events/swap2010/page10/page10.html
21. Zanoli, R., Corcoglioniti, F., Girardi, C.: Exploiting background knowledge for clustering person names. In: Proceedings of Evalita 2011 – Evaluation of NLP and Speech Tools for Italian (2012). Springer, Berlin. http://www.evalita.it/2011/information_about_publications

Chapter 7
Lexical Mediation for Ontology-Based Annotation of Multimedia

Mario Cataldi, Rossana Damiano, Vincenzo Lombardo, and Antonio Pizzo

Abstract In the last decade, the annotation of multimedia has evolved toward the use of ontologies, as a way to bridge the semantic gap between low level features of media objects and high level concepts. In many cases, the annotation terms refer to structured ontologies. Such ontologies, however, are often light scale domain oriented knowledge bases, whereas the employment of wide, commonsense ontologies would improve interoperability and knowledge sharing, with beneficial effects on search and navigation. In this chapter, we present an approach to the semantic annotation of media objects through a meaning negotiation approach that requires natural language lexical terms as interface and employs large scale commonsense ontologies. As a test case, we apply the annotation to narrative media objects, using a meta–ontology, called Drammar, to describe their structure. We present the annotation schema, the software architecture for integrating several large scale ontologies, and the lexical interface for negotiating the ontological term. We also describe an evaluation of the proposed approach, conducted through experiments with annotators.

7.1 Introduction

The huge amount of available multimedia resources requires novel forms of content indexing that is oriented toward re–use and retrieval. Beside the recent trend of user–generated annotations, structured semantic annotation has been proposed as a means to develop advanced search and retrieval tools, that rely upon both textual descriptions of the resource and signal content.In the last decade, thanks to the

M. Cataldi (✉) · R. Damiano · V. Lombardo · A. Pizzo
Università di Torino, Torino, Italy
e-mail: cataldi@di.unito.it; rossana@di.unito.it; vincenzo@di.unito.it; antonio.pizzo@unito.it

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_7,

standards developed by the Semantic Web project, metadata can be expressed by reference to ontologies, thus guaranteeing the use of a shared, machine-readable format that goes beyond the limitations of keyword annotation. Ontologies are essential to represent and reason about shared meanings [20, 21] and allow the systems to describe the same resources with the same concepts belonging to a shared knowledge base [6].

Most approaches to the semantic annotation of multimedia content, aimed at bridging the so–called semantic gap by mapping low–level features onto semantic concepts, refer to specific sets of semantic descriptors, developed for specific content types and tasks. For example, consider the the MediaMill set of 101 semantic descriptors, suited for the MediaMill repository [46], or light ontologies such as LSCOM (a few thousands of concepts), specifically designed for a corpus of broadcast news[33]. Such approaches work for limited scale ontologies, where declarative rules and indexing algorithms directly refer to ontology nodes. On the contrary, when dealing with commonsense knowledge, the size and complexity of the ontologies make the mapping between low level features and ontology nodes hard. In order to support the use of large–scale commonsense ontologies in semantic annotation, we claim that the manual or semiautomatic generation of annotations is a crucial step: it provides training data for knowledge acquisition and learning [34] and ground truth data for evaluation purposes.

This paper presents a Wordnet–based lexical interface to the annotation, i.e., a system that permits a human user to access – via the lexical knowledge incorporated in WordNet – vast ontological knowledge bases for annotation purposes. Ontology concepts are selected by inserting natural language terms in a web-based system that helps the user visualize the multimedia documents and "negotiate" an ontological concept through a presentation of the glosses associated. This "meaning negotiation" process relies on the lexical knowledge–bases MultiWordNet [39] and WordNet [32]. The large–scale commonsense ontologies are the Suggested Upper Merged Ontology (SUMO, [37]) and Yet Another Great Ontology (YAGO, [49]), merged into YAGOSUMO project [31]. YAGOSUMO incorporates almost 80 millions of entities from YAGO (which is based on Wikipedia and WordNet) into SUMO, a highly axiomatized formal upper ontology. Thus, within the proposed framework, taking as input the word senses, the system queries YAGOSUMO in order to retrieve the ontological concepts that best match a set of ontological conditions imposed through YAGOSUMO properties. The description of situations, processes, and events require the connection of several concepts in a single relation. For this annotation, we rely upon the frame notion provided by the knowledge base FrameNet [2].

The lexicon–based approach described here is part of the CADMOS project, aiming at a Character–centred Annotation of Dramatic Media ObjectS (i.e., media objects having as their content character-enacted stories). We present the software architecture of the CADMOS project and the result of a test over an experimental corpus of narrative media (cf. [8]), where stories are presented in audiovisual and textual form. We believe that narrative media provide a valid test bed for the use

of commonsense knowledge: notwithstanding the constraints posed by media and genres, they take as their object the real world, suitable to test the use of large commonsense ontologies.

The paper is organized as follows: in Sect. 7.2 we survey the relevant literature on the use of ontologies for multimedia content and the semantic annotation. Then, after the introduction of our case study, namely the annotation of dramatic media objects and the meta–ontology of narrative features, in Sect. 7.4.1 we introduce the architecture of the proposed framework and describe the methods and modules for implementing the lexicon–based method for the selection of the ontological concepts (Sect. 7.4.2). Finally, in Sect. 7.5 we report the experimental test, with user studies and analyses on the lexicon–based method for accessing the ontological knowledge base.

7.2 Related Work

In this section, we consider video annotation as a paradigmatic case for media annotation, both for the interest it has raised in the multimedia community [25] and for its relevance to the case study we describe in this paper.

Semantic annotation of video is generally performed by classifying content elements according to some ontology that represents its typical content [4]. Standardized metadata vocabularies, such as the LSCOM initiative [33], have been created to make the representation of video content interoperable, together with specialized vocabularies for videos related to various domains.

The annotation process implies a mapping of the individual elements of the video onto the terms of the reference ontology. The detection of the individual elements can be performed manually or automatically, through software systems for image and video analysis. The mapping of individual elements onto ontology concepts can be accomplished by simple pre-defined correspondences or through the definition of rules that establish relationships between the annotated terms to specify more abstract concepts. In this case, the terms of the ontology are mapped onto appropriate knowledge models that encode the spatio–temporal combination of low– or intermediate–level features [5, 15, 27]. The Video Event Representation Language (VERL) models events in the form of changes of state [17], following the paradigm of the event calculus [26]. This language introduces a compositional approach, yielding complex events from from primitive concepts. It gives prominence to perceived objects and events, allowing for sequences or multi–threaded compositions, connected to the video through the beginning and end keyframes of the event. The VERL approach does not refer to large–scale domain ontologies or to acknowledged patterns to provide a structure to the event models. Ballan et al. use the hierarchical linguistic relations over lexical entries encoded in WordNet to learn and refine rules that detect complex events from simple ones [3]. An ontology-based approach to the detection and annotation of events is video is pursued also by the

Mind's Eye project [10]. In this work, the events detected in video are described as "verbs", described in terms of a spatial model of motion. This approach relies on the paradigm of Ontological Realism, according to which the representation of the universals shared by different domain descriptions and applications is kept distinct from the representation of domain–specific templates. Beside the annotation of event, there is a growing interest for the representation of actions carried out by humans in a video (see, e.g., [51]). The representation of actions can be useful in the annotation of complex events, and can address many practical tasks, such as video surveillance.

An important limitation of current approaches is that they generally manage a limited range of concepts because of the inability to automatically recognize a wide range of elements from videos. In order to avoid these problems – and enable the use of a wider range of terms –, some annotation tools (as in [44]) allow the user to manually map a term with a specific ontological concept. The importance of the lexicon design for the task of recognition has been also pointed out by Hauptmann [22]: according to [22], the key to the creation of general–purpose content annotation and retrieval tools is the identification of a large lexicons and taxonomic classification schemes. The use of large-scale ontologies, however, introduces a new problem: the access to the data is, for the user, an extremely hard task, because of the size and the complexity of the considered data (cf. [33] and successive developments). An approach to improve the interoperability of the annotations is to constrain the scope of the semantic model: for example, the Lode ontology [28] describes the concept of public event (concert, performance, etc.), its structure, and properties, by abstracting on the descriptions provided by different directories.

A number of research projects directly address the problem of efficiently annotating video resources through large, shared, knowledge bases. Among all, the Advène project [42] addresses the annotation of digital video fragments by proposing a system that leverage free textual description of the content, cross-segment links, transcribed speech, etc. This information can be exploited to provide advanced visualization and navigation tools for the video. As a result of the annotation, the video becomes available in hypertext format. The annotation is therefore independent from the video data and is contained in a separate package that can be exchanged on the net.

A media independent project is provided by the OntoMedia ontology [23], exploited across different projects (such as the Contextus Project [24]) to annotate the narrative content of different media documents, ranging from written literature to comics and tv fiction. The OntoMedia ontology mainly focuses on the representation of events and the order in which they are exposed according to a time line, rather then to the specific features of the single medium (video, text, etc.). Rather than being tailored to event detection or annotation, OntoMedia lends itself to the comparison of cross-media versions of the same story (for example, a novel and its filmic adaptation), where the story is rearranged according to different timelines in the different realizations of the story.

7.3 Case Study: Annotating Stories in Video

Narrative annotation requires the use of a semantic model to structure the description of stories. In order to make the annotated data interoperable and shareable among different projects, this models should abstract from the specific medium by which the story is conveyed and from the constraints posed by the conventions of specific genres or formats. In the Cadmos project, the annotation model is provided by the Drammar ontology.

Written in the Ontology Web Language [30], Drammar is not exclusively aimed at video, but relies on the concept of 'dramatic media', i.e., media displaying live action [16], that assign the character a primary role in the exposition of content. According to [43], in fact, media are more and more exploiting the power of narrative. With respect to the approaches presented in the previous section, Drammar shares with them the basic assumption that a media object can be segmented into meaningful units. However, it replaces the previous definition of units, respectively based on production (Answer project), thematic (Advène project) and structuralist concepts (OntoMedia ontology), with a segmentation methodology that relies on the identification characters' actions.

In order to describe the behavior of characters, Drammar borrows the definition of agents from the BDI (Belief Desire Intention) model [11, 41], inspired by the framework of bounded rationality [7]. According to this model, agents devise plans (i.e., intentions) to achieve their desires, given their subjective beliefs about the current state of the world. This model, widely used in computational storytelling [1, 35], in Drammar is augmented with the notions of emotional states and moral values [13, 14, 38], to address the specific commitment of drama towards these notions.

Notice that the semantic model only describes the universe of discourse of drama. However, since the drama elements are also physical and abstract entities such as characters, institutions, objects, and so on, the annotation process needs a vocabulary for describing the real world counterparts of these elements. The paradigm of linked data [6] offers a way to link external semantic resources when describing some entity in an ontology. In the World Wide Web, classes, properties and individual of any ontology can be referred anywhere by using URIs to identify them. Thanks to this mechanism, in semantic annotation an external ontology can be employed as a terminological base without requiring an explicit integration of it in the annotation model. Cadmos relies on the paradigm of linked data to refer to individuals that belong to different datasets. For example, for describing the type of the objects that appear in a story, the Drammar ontology employs the *type* property. In each triple where this property is employed (<object,type,URI>), its value (the third element of the triple) is the URI of a concept in another ontology that corresponds to the type of that object. So, if the object is a car, the type property of this object will take as its value the URI of the concept of "car" in the external ontology that provides the vocabulary for the annotation of object types.

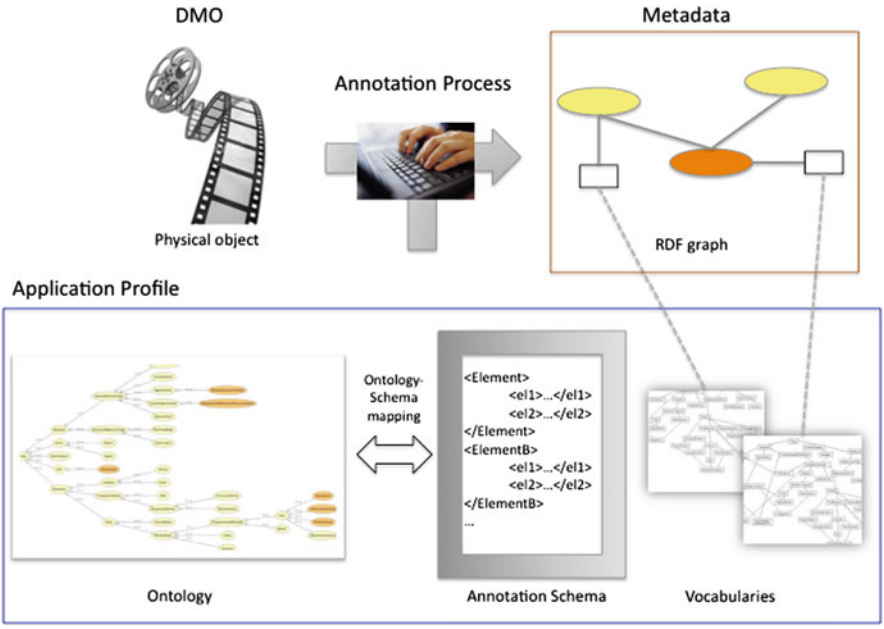

Fig. 7.1 The annotation process framework. The annotation system incorporates the semantic model (Application Profile, below) and the external vocabularies, thus enforcing the correctness of the metadata encoded by hand by the human annotators and their translation into a formal language

The schema depicted in Fig. 7.1 represents the elements involved in a semantic annotation framework. The input to the process is given by the resource to be annotated (in Cadmos, a dramatic media object, or DMO) and the Application Profile. The Application profiles include the semantic model (in Cadmos, the Drammar ontology), the annotation schema and a set of vocabularies (i.e., external ontologies). The annotation schema is a hierarchical structure of descriptors, mapped onto the concepts represented in the semantic model; the values for the descriptors are given by the entries in the vocabularies.

The annotation process is accomplished by a human annotator with the help of a software tool that incorporates the Application Profile. Through this software, the annotator fills the annotation schema, selecting values for the descriptors from the vocabularies. Once the annotation schema has been filled, the system maps it onto the appropriate concepts and relations in the model, creating the right instances of the ontology classes and relations. The creation of the ontology instances is carried out by the system in a transparent way to the user: the output of this process is the metadata of the input DMO, encoded as an RDF graph. Also, in our framework, the selection of the values for the descriptors is not carried out by the annotator by direct access to the vocabularies (i.e., browsing the external ontologies) but is mediated by the natural language, as described in Sect. 7.4.

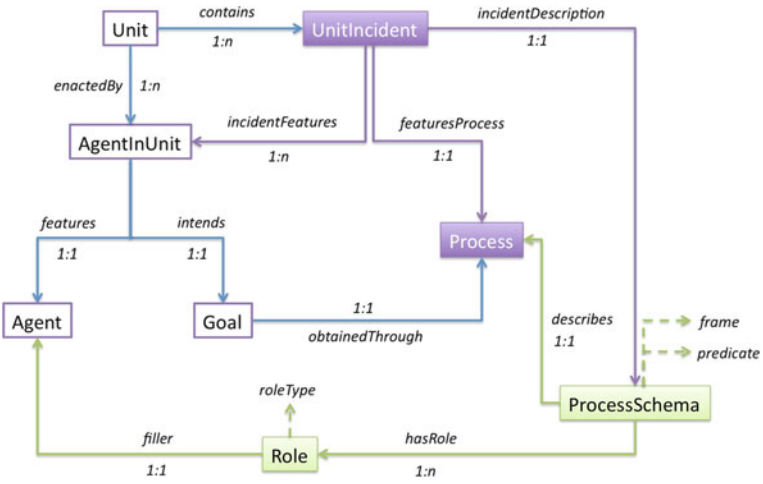

Fig. 7.2 The template for annotating story incidents within the Cadmos system; the incident is described by an ontological concept, a semantic frame and its participants

The top level of the Drammar ontology consists of five disjoint classes: *Unit*, *Dynamics*, *Entity*, *Relation* and *DescriptionTemplate*. The notions of unit, dynamics and entity generalize over a tripartite model of drama composed of plot tree (*Unit*), story advancement (*Dynamics*) and character (*Entity*), respectively [9]. According to this model, a story is segmented into units; units feature entities, involved in actions and events, i.e. the incidents that occur in units; units are arranged at different levels of detail, forming a tree structure. The *Dynamics* class contains the basic concepts for modeling the advancement of drama as a sequence of states interconnected by incidents. Finally, the *Relation* class subsumes the concepts that describe the properties of drama entities in a certain unit, such as the characters' goals and the conflicts among them. As stated before, agents are described according to paradigm of intelligent agents, following the Belief Desire Intention (BDI) model, as operationalized in several agent architectures [36,40], and enriched with emotions and moral values [12, 38].

The annotation process centers on the description of the story units: a unit is enacted by certain characters, who perform actions in it, and/or contains certain naturally occurring events. As a result of these actions and events (collectively named incidents), the unit brings the world state from an initial state to a final state. In a situation calculus perspective [29], a unit can be seen as an operator characterized by preconditions and effects, that bridges the story world from a state in which the preconditions hold to one in which the effects hold. So, in Drammar, the unit is modeled as having preconditions and effects. The relation between the unit and the world state (before and after that unit) is modeled by the *hasPrecondition* and *hasEffect* properties, that connect the Unit with a StoryState.

As represented by the Fig. 7.2, a Unit contains (*containsEvent*) some *UnitIncident* (an agent's action or an event) and is *enactedBy* some Agents. The *UnitIncident*

class (inspired by the Time Indexed Situation and the Time Indexed Participation patterns defined by Gangemi and Presutti [18] and Gangemi et al. [19]) connects the occurrence of an event (no matter if it is an agent's action or a naturally occurring event) with the entities (agents and objects) which participate to it, and to set it into the time extent provided by a unit. Similarly to the *UnitIncident* class, the *StoryState* class connects the occurrence of a state (be it a mental state or a state of affairs) with the entities (agents and objects) which participate to the state, and to set this event in relation to a unit. The linguistic description of the incident, then, is attached to the *ProcessSchema* (or *StateSchema*) class, which in turn is connected to the entities which play a role in the incident through the *Role* class. The *Process* class is connected to the *ProcessSchema* class through the *incidentDescription property* (Fig. 7.2, below).

The *ProcessSchema* class describes the process through the following properties:

- The *predicate* data property links the *ProcessSchema* to a single concept represented into an external ontology of processes.
- The *frame* data property links the *ProcessSchema* to a single linguistic frame.
- The *hasRole* object property links the *ProcessSchema* to a thematic role (an instance of the *Role* class) belonging to the linguistic description of the process. Since a process normally encompasses multiple roles, an instance of *ProcessSchema* normally has multiple instances of the *hasRole* property.

The *Role* class represents a thematic role in the description of the process and can be filled by a drama entity through the *filler* property. The *roleType* property of the *Role* class provides a label for the type of role. By using this schema, the description of the process is entirely delegated to external ontological and linguistic resources, lifting Drammar from the responsibility of modeling common sense knowledge with which it is not concerned.

In order to better understand the final output of the RDF annotation, here we also provide a short example related to Act I, Scene 2, of Shakespeare's *Romeo and Juliet*. In particular, with our annotation system it is possible to describe the scene, where Romeo is entering, unseen, the garden of the Capulet's villa to find out where is Juliet. The movie fragment shows Romeo in the act of entering the garden and approaching the indoor balcony by the poolside in order to find Juliet's room.

```
:romeo rdf:type :Agent, owl:NamedIndividual;
       :age "18"^^xsd:int;
       :name "Romeo Montague"^^xsd:string;
       :gender "male"^^xsd:anyURI.

:goalOfRomeoInUnit1 rdf:type :Goal,
       :obtainedThrough :processRomeo;
       :hasStatus :goalRomeoStatus;

:processRomeo rdf:type :Process, owl:NamedIndividual;
```

```
:schemaRomeo rdf:type :ProcessSchema, owl:NamedIndividual
       :describes :processRomeo
       :predicate "finding"^^xsd:anyURI,
       :frame "Arriving"^^xsd:string;

:goalRomeoStatus rdf:type :GoalStatus,
                          owl:NamedIndividual;
       :goalState "active"^^xsd:string.
```

7.4 Accessing Large Scale Commonsense Knowledge Through a Lexical Interface

In this section, we describe the CADMOS system for the annotation of story and characters in video and the mechanism for accessing large ontologies from lexical knowledge it encompasses. In the CADMOS system, in fact, the NLP-mediated approach to large ontologies that we propose is employed to help human annotators identifying the appropriate concepts when describing what characters do in video, their motivations and the environment in which the action takes place.

7.4.1 The Architecture of CADMOS

The architecture of CADMOS, illustrated in Fig. 7.3, includes six main modules:

- The User Interface;
- The Annotation Manager;
- The Ontology Framework;
- The Ontology Mashup;
- The NL-to-Onto module;
- The Video Repository.

The system works as follows: the textual and multimedia documents to be annotated (also called media objects in this chapter) are stored and indexed within a repository, called Media Repository. In particular, video documents can be uploaded and visualized through a web-based User Interface, which is also the front-end for the annotation process. The Media Repository relies on a multimedia database to archive the video in the repository and a storage server to stream the requested video to the Annotation manager. The entire annotation work flow is led by the Annotation Manager which communicates with the Media Repository and the Ontology Framework, guiding the user within the annotation process.

The Ontology Framework carries out the reasoning services requested by the Annotation Manager and bridges the gap between the natural language input of the user and the ontological knowledge (Ontology Mashup). Currently, within the

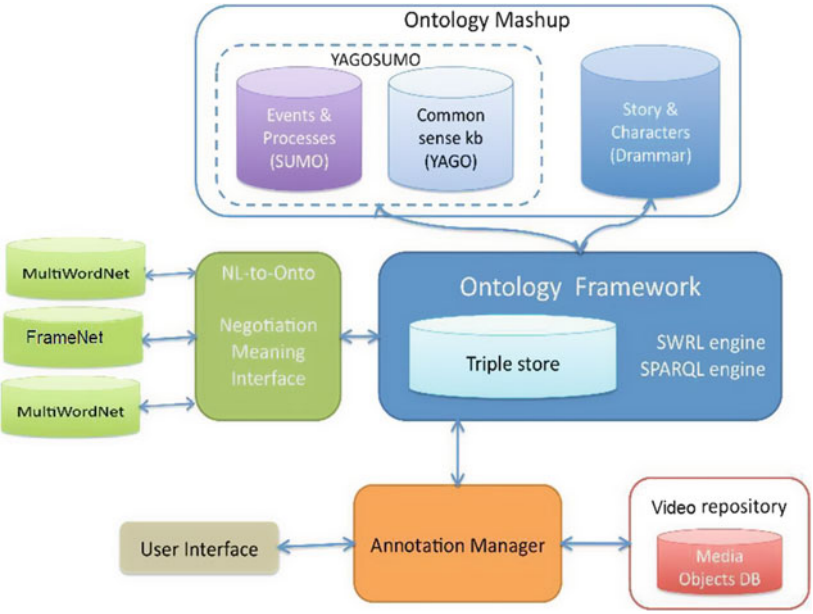

Fig. 7.3 The architecture of the Cadmos system

proposed architecture, the Ontology Framework is provided by Jena[1] and it has been integrated with the Pellet reasoner[2].

This mediation between natural language input and ontologies is possible through the use of the NL-to-Onto module: as explained in detail in Sect. 7.4, given a user input, expressed in one of the available languages, this module first permits to disambiguate the sense of the inserted term (in the selected language) by proposing to the user its different possible meanings; then, it associates in a transparent way the selected meaning to a unique sense in English. Moreover, when it is necessary, it permits to associate the selected sense to a semantic frame and to a set of thematic roles (therefore permitting a better contextualization of the annotated situation).

Currently, the Ontology Mashup contains two well-known ontologies: the Suggested Upper Merged Ontology (SUMO [37][3]) and Yet Another Great Ontology (YAGO [49][4]), merged into YAGOSUMO [31][5]. This combined ontology provides a very detailed information about millions of entities, such as people, cities, organizations, and companies and can be positively used not only for annotation

[1] http://jena.sourceforge.net/

[2] http://clarkparsia.com/pellet/

[3] http://www.ontologyportal.org/

[4] http://www.mpi-inf.mpg.de/yago-naga/yago/

[5] http://www.mpi-inf.mpg.de/~gdemelo/yagosumo.html

purposes, but also for automated knowledge processing and reasoning. The univocal mapping between a sense and an ontological concept is also possible thanks to the integration of WordNet in YAGOSUMO[6]. The Ontology Mashup module also contains the annotation model (expressed by the Drammar ontology, described in the previous section), that provides the elements and properties employed to annotate the media objects within the system.

It is important to note that the current architecture also support user queries on the annotated objects through the User Interface; in this case, the Ontology Framework translates the user request into a SPARQL query and performs the requested operation on the triple store (which contains the annotated information). The result is returned to the Annotation Manager that retrieves the relevant associated media objects and presents them to the user through the User Interface.

7.4.2 The Meaning Negotiation Process

In order to fill the schema for describing story incidents with terms from external ontologies, our approach proposes a guided access to the ontology concepts based on natural language expressions. For this, we designed and implemented a tool that helps the user access the commonsense knowledge through a linguistic-based disambiguation process. The high-level schema of the entire work flow is shown in Fig. 7.4).

In detail, the first part of this negotiation process can be described as a word sense disambiguation step aimed at associating to each natural language term/expression a unique definition which makes it distinguishable from any other possible meaning. In particular, for each element in the annotation schema, the system implements the following steps:

- The annotator initially expresses the content as a word (or a minimal set of words) in one of the available languages (English, Italian, Spanish, Portuguese, Hebrew and Romanian): the keyword-based query is forwarded to the NL-to-Onto module and the possible meanings of the query are shown by using the related glosses;

[6]Each synset of WordNet becomes a class of YAGO [48]. They only exclude the proper nouns known to WordNet, which in fact would be individuals (Albert Einstein, e.g., is also known to WordNet, but excluded). Moreover, there are roughly 15,000 cases, in which an entity is contributed by both WordNet and Wikipedia (i.e. a WordNet synset contains a common noun that is the name of a Wikipedia page). In some of these cases, the Wikipedia page describes an individual that bears a common noun as its name. In the overwhelming majority of the cases, however, the Wikipedia page is simply about the common noun (e.g. the Wikipedia page Physicist is about physicists). To be on the safe side, they always give preference to WordNet and discard the Wikipedia individual in case of a conflict. This way, they can lose information about individuals that bear a common noun as name, but it ensures that all common nouns are classes and no entity is duplicated.

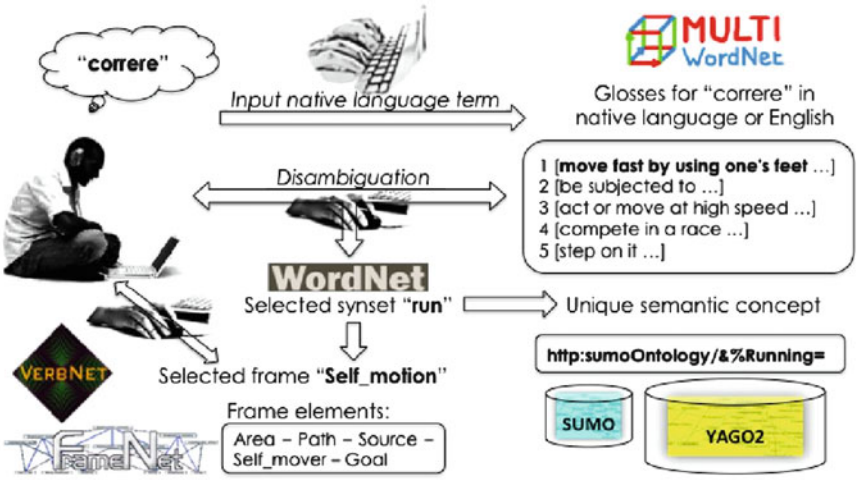

Fig. 7.4 The disambiguation model proposed in this chapter includes several knowledge bases: MultiWordNet, WordNet, FrameNet and YAGOSUMO

- The annotator disambiguates the meaning of her query by selecting the gloss that best matches her/his request;
- Each gloss is then automatically and univocally mapped to a representative English WordNet synset;
- Finally, the system queries the YAGOSUMO knowledge base to retrieve the ontological concept that can positively represent the retrieved English synset.

More in detail, for each user query, the system retrieves the related definitions by querying the MultiWordNet data base searching for the synsets that are associated to the inserted query. In fact, within MultiWordNet, each synset (for each language) is represented as a tuple with four attributes:

- *id*: the WordNet synset identifier;
- *word*: the lemmas that can be associated to the considered WordNet synset;
- *phrase*: a elocutionary expression that can represent the considered synset;
- *gloss*: a formal definition, as in a dictionary, expressed in natural language (with real examples), of the WordNet synset.

Thus, given the user's query, the system retrieves the related definitions by querying the NL-to-Onto module searching for the glosses which related "word" contains (also partially) the inserted term. This operation is initially performed on the table related to the user language. However, if related glosses are not available (in fact, except for the English table, it is not guaranteed a 1:1 mapping between each synset and a gloss), the system leverages the *id*s to retrieve, on the English table, the related English glosses (which are always guaranteed). At this step, the retrieved glosses are reported to the user (through the User Interface module) in

her language (when available), or in English otherwise. The user then reads and analyzes the reported definitions in order to select the most suitable one.

Then, the system leverages again the related synset id to retrieve additional information about the disambiguated sense. In particular, it is possible to query the YAGOSUMO knowledge base to retrieve the related ontological concept; this is possible by using the ontological property *hasSynsetId*, represented within YAGOSUMO, which links an ontological concept to its related id in WordNet. In fact, the considered knowledge base has been constructed by merging, with an unsupervised method, the information expressed by Wikipedia (each article is represented as a class or an individual) and the linguistic hierarchical knowledge provided by WordNet. In fact, the information contained in Wikipedia is organized and structured based on its categorizations (that provides a basic hierarchical structure among the classes) refined and re-organized through the hyponyms/hypernym hierarchy provided by WordNet (i.e., they are converted into ontological high-level internal nodes). More in detail, YAGO has been automatically derived from Wikipedia and WordNet by including the taxonomic Is-A hierarchy as well as semantic relations between entities. The facts for YAGO have been extracted from the category system and the infoboxes of Wikipedia and have been combined with taxonomic relations from WordNet.

Note that YAGOSUMO, for each ontological concept, does not always associate the same id stored within MultiWordNet (this is because of data integration problems). Therefore, in order to avoid this problem, we leverage the related gloss to retrieve the correlated YAGOSUMO concept. This can be achieved through the ontological property *hasGloss* which links each single ontological concept to a unique formal definition extracted by WordNet. Note that, again, MultiWordNet and YAGOSUMO do not always associate the same gloss to each WordNet synset. In fact, they has been developed based on different WordNet versions and are therefore not completely aligned. Thus, when also this mapping system fails, our framework uses the related lemmas associated within MultiWordNet (stored in the "word" attribute) to retrieve the YAGOSUMO ontological concepts. This is possible through the use of the ontological property called "hasMeaning", which links an ontological concept to all the terms (expressed as strings) than can be represented by the concept. Note that, using the associated lemmas, it could be possible to retrieve multiple concepts for each single selected definition. If this is the case, another negotiation step is required (i.e, the user needs to manually select the most suitable ontological concept).

Once the relevant concept has been retrieved, if the user is annotating a situation/event/action, the system can help the user in the annotation process by also proposing to the annotator the frame structure related to that concept (which can help describe the situation/event/action that needs to be annotated). Let consider for example the "Questioning" frame; it requires the specification of the elements "Message" (the exact wording of the questions), "Topic", "Addressee" and "Speaker". Using the information related to the frames, complex situations or events can be easily understood and annotated. The mapping between an ontological concept and a semantic frame is possible through the MapNet project [50] and

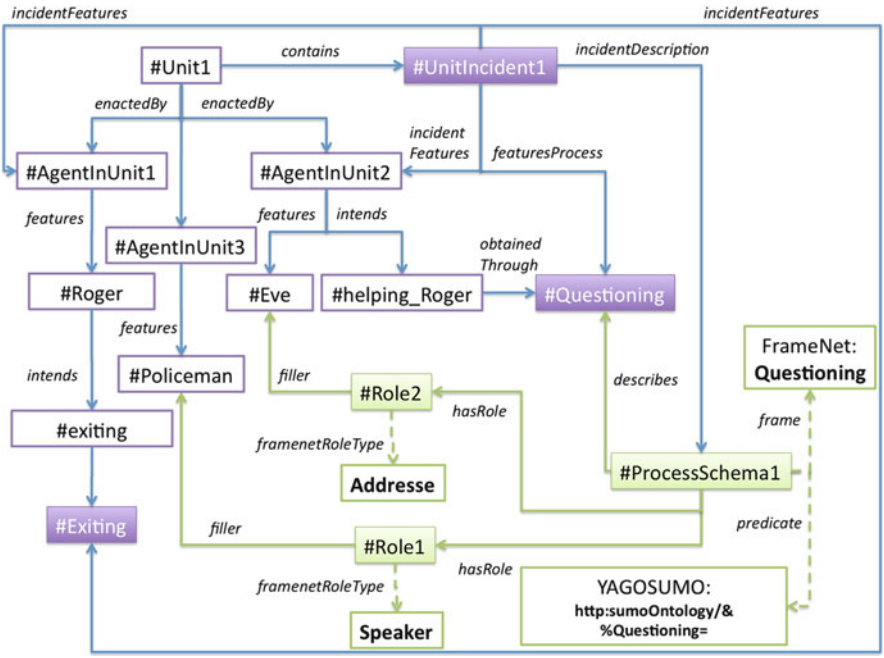

Fig. 7.5 The annotation of an example incident from North By Northwest (a policeman questions Eve about Roger)

FrameNet itself. When no frame is found (since the mapping is not yet complete), a generic frame is proposed to the annotator, accompanied by the set of 23 roles taken from the VerbNet project [45].

As an example of annotation, consider a scene from the classic Hollywood film "North by Northwest" by Alfred Hitchcock (1959). In this scene, the protagonist of the film, Roger Thornhill, gets off the train with Eve, disguised as a porter. The policeman who is pursuing Roger questions Eve about Roger and she answers that she has not seen him. Figure 7.5 illustrates the annotation of the incident. In the figure, *#UnitIncident1* features three agents (Roger, Eve and the Policeman, all instances of the *Agent* class). These agents are also the fillers (through the *filler* property) of the roles attached to the linguistic frame which describes the action featured in the incident (*#Questioning*). The role labels are provided by the *framenetRoleType* property, Speaker (Policeman, *#Role1*), Adressee (Eve, *#Role2*), Topic (Roger *#Role3*, not shown in the figure). The action is described by a SUMO concept ("Questioning", see the *predicate* property) and by the 'Questioning" frame (*frame* property) The annotation also represents the characters' goals (*#Exiting* for Roger, *#Helping_Roger* for Eve). In Cadmos, the propositional content of goals, be it a state or a process, is also described through a situation schema, although this part has been omitted in the figure for space reasons.

7.5 Annotation Test and Discussion

In this section we report about an annotation test on a small–size corpus of narrative media objects. We analyze the numbers involved in our knowledge bases and the mappings between the lexical knowledge base and the ontological concepts and frames, respectively; then we report on the behavior of annotators through the "meaning negotiation" process and a comparison with free tagging.

7.5.1 Experimental Setting

The annotation of video through the ontological knowledge base challenges the sharing of the ontological concepts, by potentially introducing a large variety of terms both inter–language and inter–annotator, thus preventing interoperability. Thus, preliminarily, we measured the amount of positive mappings between the linguistic knowledge base terms (the terms stored in the NL–to–Onto module) and the ontology knowledge base YAGOSUMO. In particular, we tested how many terms contained within the lexical knowledge base (MultiWordnet) can be positively mapped to a concept within YAGOSUMO. In Table 7.1(a) and (b) we report the results related to two different languages contained within MultiWordNet: English and Italian. As it is shown in Table 7.1(a) and (b), 92.86 % of the English terms reported in MultiWordNet (80.03 % for Italian terms) are directly linked to an ontological concept. In fact, the presented system provides a guided access to ontological concepts related to ∼95 % of the English verbs, ∼86 % of the English nouns, ∼90 % of the English adjectives and ∼97 % of the English adverbs. The user can therefore leverage the expressiveness of the ontological knowledge base for a very significant percentage of natural language terms. Considering the Italian language, the percentage of terms that can be successfully linked to some ontological concept lowers a little (event if it remains higher than 75 % for all the considered parts of speech); in fact, the considered ontologies (YAGOSUMO) are expressed in English and the system needs to find the correspondent concepts by also starting from glosses (or lemmas) in different languages. Thus, the data integration problems affect the mappings and lowers the percentage of terms in other languages associable to some ontological concept.

We also tested the mapping of MultiWordNet terms onto Framenet frames, that are employed for the annotation of situations/events/actions. Thus, we measured the percentage of natural language terms that can be positively mapped to a frame structure in FrameNet. As it is shown in Table 7.2(a) and (b), nouns, adjectives and adverbs resulted in a very low percentage of positive mappings; as expected, verbs are more commonly considered for describing situations and events. In fact, for the verbs, our test reports a significantly higher percentage of positive mappings (60 % for English and 70 % for Italian). On the other hand, as explained in Section 7.4, when the system is not able to provide a mapping to a frame, it resorts to a general frame with high-level frame elements (taken from the knowledge base VerbNet).

Table 7.1 Mappings among terms (English(a) and Italian (b)) in MultiWordnet and the considered large-scale knowledge base (YAGOSUMO)

	Total # Synsets	# Verbs	# Nouns	# Adjective	# Adverbs
(a) English terms					
Total # in MultiWordNet	102,101	12,144	68,465	17,917	3,575
Total # of Mappings in YAGOSUMO	94,817	10,452	64,831	16,062	3,472
Percentage	92.86	86.06	94.69	89.64	97.11
(b) Italian terms					
Total # in MultiWordNet	38,653	4,985	28,517	3,911	1,240
Total # of Mappings in YAGOSUMO	30,937	4,332	21,752	3,643	1,210
Percentage	80.03	86.90	76.27	93.14	97.58

Table 7.2 Mappings among terms (English (a) and Italian (b)) in MultiWordnet and the the sematic frames stored in FrameNet

	Total # Synsets	# Verbs	# Nouns	# Adjective	# Adverbs
(a) English terms					
Total # in MultiWordNet	102,101	12,144	68,465	17,917	3,575
Total # of mappings in FrameNet	22,351	7,193	10,258	4,352	548
Percentage	21.89	59.23	14.98	24.28	15.32
(b) Italian terms					
Total # in MultiWordNet	38,653	4,985	28,517	3,911	1,240
Total # of mappings in FrameNet	12,357	3,643	7,252	1,212	250
Percentage	31.96	73.07	25.43	30.98	20.16

The annotation experiment we ran asked to four users from different countries and speaking different languages, to annotate three different videos with the help of the annotation system. In particular, we considered the following videos:

- The 2-hour movie "North by northwest" (NbN), a classic Hollywood movie by Alfred Hitchcock, about an advertiser who escapes from both a criminal gang, who tries to kill him (having mistaken him for a CIA agent), and from the police, who tries to arrest him because of an unjust accuse of homicide;
- The multi-prized short animated movie "Oktapodi", about an octopus who tries to save its partner from being cooked, after having been taken away from their love nest (a fish tank);
- A television commercial of the "Zippo" lighter, where a couple of gangsters try to burn a hostage but waste all the matches they have.

For all these resources, the users queried 289 times the lexical base for annotation. Considering all these requests, the users had to disambiguate in average

among 2.83 glosses. It is interesting to note that this value is higher than the overall ambiguity; in fact, we calculated that, in average, each natural language term stored within our framework is associated to 1.71 glosses. This behaviour means that annotators tend to use terms that are more generic than the average (i.e., they results in a higher number of possible correlated definitions); in fact, more specific terms lower the average of this ambiguity factor to less than 2.

Moreover, we also asked the user to reply to a subjective qualitative–oriented questionnaire about the difficulty of using appropriate linguistic terms and the consequent selection of the adequate definition. For this, we asked the annotators to reply to the following questions:

- Was it subjectively hard to make a selection from the list of definition provided by the system?
- How many times did you revise your choice by searching for a synonym?
- How many times did you change your interpretation because of the inadequate definitions proposed by the system?
- How many times did you resort to free text, giving up the search on an ontological concept?

The users quantified the responses of the first question using a 3-point scale ratings (“easy to use”, “intermediate”, and “hard to use”), while for the other questions they simply counted the number of cases that were in accordance with the proposed questions.

7.5.2 Results and Discussion

Regarding the first question of the questionnaire, the users replied that the framework was “easy to use” in 80.23 % of the cases, while for only 6.51 % of the cases they found the system “hard to use” (for the 13.26 % of the case they reported an “intermediate” difficulty), highlighting the simplicity of the proposed framework in annotating resources with ontological concept through a linguistic interface. Moreover, for 61.87 % of the cases the user did not have to revise their query to search a suitable definition (second question), while in only 9.76 % of the cases they had to repeat their requests (by inserting synonyms) more than once (and in 28.37 % of the cases they reformulated their query only once).

It is important to note that, as already reported, some data integration problems emerged; in fact, regarding the third question, the user had to change their formulations in 38.76 % of the cases, exhibiting the overall complexity of integrating different vast knowledge bases. In fact, in these cases, the annotators retrieved a set of results related to their queries but they were not satisfied with the proposed meanings; in other words, the system contained the terms provided by the users but they were not described in the way the users supposed. However, even with these problems, the users retrieved a satisfactory definition in 61.21 % of the cases, exhibiting an overall robustness of the presented approach. Notice also that only in

Table 7.3 Resource–based tag analysis: number of tags per category

Resource-based, 268 tags						
Title	Actor	Director	Production	Editing	Publishing	Genre
68	102	28	31	28	6	5

Table 7.4 Content–based tag analysis: number of tags per category

Content–based, 29 tags			
Character	Object	Environment	Action/Situation
10	8	7	4

16.92 % of the cases the users decided to resort to free text (fourth question) instead of insisting searching for the most suitable ontological concept. In other words, it means that in 83.08 % of the cases the users easily retrieved a related ontological concepts in one or very few attempts.

About the behavior displayed with the annotation of the semantic relations for situations/events through frames (i.e., any video unit contains at least one event or action). In particular we asked to the user to report how many times they found a suitable frame. The answers to this question resulted in 62.45 % of satisfactory mappings. It is interesting to note that the users typed in 97.67 % of the cases a verb when they needed to annotate events, so a frame was likely present.

Finally, we checked whether our ontology–based annotation could be recovered, at least in part, from the free tags provided by users in the public repositories. So, we made an informal survey of the user–contributed tags on the feature film case (North by Northwest) in YouTube. After searching YouTube with the simple keywords "North by northwest", we manually discarded all the results that did not belong to the original movie (59 % of the first 100 results consisted of advertising materials, CGI animations inspired by the movie, user–generated editings of the movie, etc.). We restricted our analysis to the Film and Animation category and considered only the first 100 results. By doing so, we collected 378 tags, yielding 183 different tags after eliminating the repeated tags. We then collected the tags of each result and manually analyzed them to let categories emerge, following the methodology of the Grounded Theory [47]. This methodology exploits both qualitative and quantitative aspects to group the data into categories and subcategories along the axis of each category, refining the categorization through the subsequent steps of analysis. Tags were divided into fourteen different categories, grouped into two main macro–categories: media-based tags, conveying information about media type, format, etc. and content–based tags. The latter can be further subdivided into actual content–based tags and general information about the resource, approximately corresponding to the Dublin Core data set[7] (information about the owner, the creator, the date, etc.).

[7] http://dublincore.org/

The results of this analysis are illustrated in Tables 7.3 and 7.4. Most tags (268) belong to the description of the resource itself. Actual content based tags are only 29; 13 tags convey media information. Among the content based tags, most tags refer to characters ("Roger", "mother") or their qualities ("blonde", "dress"). The "Other" category (49 unique tags) collects tags that are not related to the resource, such as advertising content, misspelled words, etc.

Since a relevant number of tags are copied from the metadata that accompany the various editions of the movie, approximately one third of tags are proper names belonging to the production professionals (such as the director) and actors. Also, tags were multilingual, featuring, beyond English, German (6 tags) and Italian (1 tag). Finally, 26 tags were stop words, like the article "the" or the preposition "by". Notice that this is due to the tagging interface of YouTube, that encourages users to slip multi–word tags (such as the title) into different tags. This informal analysis shows that, with respect to the story annotation schema we propose, the overlapping relies in the resource-based tags, that we encode according to the Dublin Core schema. The overlapping is not significant at the content level, that appears to be shallow in this example tagset. In particular, narrative aspects are mainly caught through the characters (10 occurrences) and the reference to objects (8 occurrences).

7.6 Conclusion

In this chapter we have presented an approach for the semantic annotation of media items, specifically targeted at video, that exploits very large scale, shared, commonsense ontology. The ontological terms are accessed through a linguistic interface that relies on multi-lingual dictionaries and action/event/situation template structures (semantic frames).

We have tested the validity and reliability of the proposed approach by allowing different users (not domain experts) annotate videos. The framework resulted promising from a user point of view because of its capacity to soften the complexity of accessing vast ontological knowledge bases. In fact, the presented application permits to leverage a large-scale commonsense knowledge base for annotating video by using semantic concepts. The access to such a component is provided by a multilingual linguistic interface, which revealed to be effective in the annotation task.

The future research plan includes an extension of alternative mapping systems among the different resources included within the proposed framework to help the user positively leverage a higher percentage of the natural language terms/expressions for annotation purposes. Moreover, we plan to extend the test of the proposed approach to a multi-lingual community of annotators to evaluate their feedbacks and collect wide-range annotations of different video sources.

References

1. Aylett, R., Vala, M., Sequeira, P., Paiva, A.: Fearnot!–An Emergent Narrative Approach to Virtual Dramas for Anti-Bullying Education. Lecture Notes in Computer Science, vol. 4871, p. 202. Springer, Berlin (2007)
2. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The berkeley framenet project. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, ACL '98, Stroudsburg, vol. 1, pp. 86–90 (1998). Association for Computational Linguistics
3. Ballan, L., Bertini, M., Bimbo, A.D., Serra, G.: Video annotation and retrieval using ontologies and rule learning. IEEE MultiMedia **17**, 80–88 (2010)
4. Bertini, M., Cucchiara, R., Del Bimbo, A., Torniai, C.: Video annotation with pictorially enriched ontologies. In: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), Amsterdam (2005)
5. Bertini, M., Del Bimbo, A., Serra, G.: Learning ontology rules for semantic video annotation. In: Proceedings of the 2nd ACM workshop on Multimedia semantics, MS'08, pp. 1–8. ACM, New York (2008)
6. Bizer, C., Heath, T., Berners-Lee, T.: Linked data-the story so far. Int. J. Semant. Web Inf. Syst. **4**(2), 1–22 (2009)
7. Bratman, M.: Intention, Plans, and Practical Reason. Harvard University Press, Cambridge (1987)
8. Cataldi, M., Damiano, R., Lombardo, V., Pizzo, A., Sergi, D.: Integrating commonsense knowledge into the semantic annotation of narrative media objects. In: Proceedings of the 12th International Conference on Artificial Intelligence Around man and Beyond, AI*IA'11, pp. 312–323. Springer, Berlin/Heidelberg (2011)
9. Cataldi, M., Damiano, R., Lombardo, V., Pizzo, A., Sergi, D.: Integrating commonsense knowledge into the semantic annotation of narrative media objects. In: AI* IA 2011: Artificial Intelligence Around Man and Beyond, pp. 312–323. Springer, Berlin/Heidelberg (2011)
10. Ceusters, W., Corso, J.J., Fu, Y., Petropoulos, M., Krovi, V.: Introducing ontological realism for semi-supervised detection and annotation of operationally significant activity in surveillance videos. In: Proceedings of the 5th International Conference on Semantic Technologies for Intelligence, Defense and Security (STIDS), Fairfax (2010)
11. Cohen, P.R., Levesque, H.J.: Intention is choice with commitment. Artif. Intell. **42**, 213–261 (1990)
12. Damiano, R., Lombardo, V.: An Architecture for Directing Value-Driven Artificial Characters. Agents for Games and Simulations II: Trends in Techniques, Concepts and Design, pp. 76–90. Springer, Berlin (2010)
13. Damiano, R., Lombardo, V.: An Architecture for Directing Value-Driven Artificial Characters. Agents for Games and Simulations II: Trends in Techniques, Concepts and Design, pp. 76–90. Springer, Berlin (2011)
14. Damiano, R., Pizzo, A.: Emotions in drama characters and virtual agents. In: AAAI Spring Symposium on Emotion, Personality, and Social Behavior. AAAI Press, Menlo Park (2008)
15. Ekin, A., Tekalp, A.M.: Automatic soccer video analysis and summarization. In: Storage and Retrieval for Media Databases, Santa Clara, pp. 339–350 (2003)
16. Esslin, M.: The Field of Drama. Methuen, London, 1988 (1987)
17. François, A.R., Nevatia, R., Hobbs, J., Bolles, R.C.: Verl: An ontology framework for representing and annotating video events. IEEE MultiMedia **5**, 76–86 (2005)
18. Gangemi, A., Presutti, V.: Ontology design patterns. Handbook on Ontologies, pp. 221–243. Springer, Berlin (2009)
19. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with dolce. In: Proceedings of the EKAW 2002, Siguenza (2002)
20. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum. Comput. Stud. **43**, 907–928 (1995)

21. Guarino, N., Giaretta, P.: Ontologies and knowledge bases: towards a terminological clarification. In: Mars, N.J.I. (ed.) Towards very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, pp. 25–32. IOS Press, Amsterdam (1995)
22. Hauptmann, A.: Towards a large scale concept ontology for broadcast video. In: Proceedings of the Image and Video Retrieval: Third International Conference, pp. 674–675. Springer, Berlin (2004)
23. Jewell, M., Lawrence, K., Tuffield, M., Prugel-Bennett, A., Millard, D., Nixon, M., Shadbolt, N.: OntoMedia: an ontology for the representation of heterogeneous media. In: Proceedings of the SIGIR workshop on Mutlimedia Information Retrieval. ACM SIGIR. ACM, New York (2005)
24. Jewell, M., Lawrence, K., Tuffield, M., Prugel-Bennett, A., Millard, D., Nixon, Schraefel, M.C., Shadbolt, N.R.: Ontomedia: An ontology for the representation of heterogeneous media. In: Multimedia Information Retrieval Workshop (MMIR 2005) SIGIR. ACM SIGIR. ACM, New York (2005)
25. Kompatsiaris, I., Marchand-Maillet, S., van Zwol, R., Marcel, S.: Introduction to the special issue on image and video retrieval: theory and applications. Multimedia Tool Appl. **6**, 1–6 (2011)
26. Kowalski, R., Sergot, M.: A logic-based calculus of events. New Gener. Comput. **4**(1), 67–95 (1986)
27. Leonardi, R., Migliorati, P.: Semantic indexing of multimedia documents. IEEE MultiMedia **9**, 44–51 (2002)
28. Liu, X., Troncy, R., Huet, B.: Finding media illustrating events. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval, pp. 1–8. ACM, New York (2011)
29. McCarthy, A.: Mental situation calculus. TARK: Theoretical Aspects of Reasoning about Knowledge. Kaufmann Publishers, Los Altos (1986)
30. McGuinness, D., Van Harmelen, F., et al.: Owl web ontology language overview. W3C Recomm. **10**, 2004–03 (2004)
31. Melo, G.D., Suchanek, F., Pease, A.: Integrating yago into the suggested upper merged ontology. In: Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence, vol. 1, pp. 190–193. IEEE Computer Society, Washington (2008)
32. Miller, G.A.: Wordnet: A lexical database for english. Commun. ACM **38**, 39–41 (1995)
33. Naphade, M., Smith, J.R., Tesic, J., Chang, S.-F., Hsu, W., Kennedy, L., Hauptmann, A., Curtis, J.: Large-scale concept ontology for multimedia. IEEE MultiMedia **13**, 86–91 (2006)
34. Nixon, L., Dasiopoulou, S., Evain, J.-P., Hyvonen, E., Kompatsiaris, I., Troncy, R.: Multimedia, broadcasting, and eculture. In: Domingue, J., Fensel, D., Hendler, J.A. (eds.) Handbook of Semantic Web Technologies, pp. 911–975. Springer, Berlin/Heidelberg (2011)
35. Norling, E., Sonenberg, L.: Creating interactive characters with BDI agents. In: Proceedings of the Australian Workshop on Interactive Entertainment IE2004, Sydney (2004)
36. Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., Rist, T. (eds.): Intelligent Virtual Agents, 5th International Working Conference, IVA 2005 Kos, Greece, September 12–14, 2005; Proceedings Volume 3661 of Lecture Notes in Computer Science. Springer, Berlin (2005)
37. Pease, A., Niles, I., Li, J.: The suggested upper merged ontology: a large ontology for the semantic web and its applications. In: Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, Edmonton, vol. 28 (2002)
38. Peinado, F., Cavazza, M., Pizzi, D.: Revisiting character-based affective storytelling under a narrative BDI framework. In: Proceedings of the ICIDIS08, Erfurt (2008)
39. Pianta, E., Bentivogli, L., Girardi, C.: Multiwordnet: developing an aligned multilingual database. In: Proceedings of the First International Conference on Global WordNet. Central Institute of Indian Languages, Mysore (2002)
40. Pokahr, A., Braubach, L., Lamersdorf, W.: Jadex: a BDI reasoning engine. Multiagent Syst. Artif. Soc. Simul. Organiz. **15**, 149 (2005)
41. Rao, A., Georgeff, M.: Deliberation and intentions. In: Proceedings of the 7th Conference on Uncertainity in ArtificialIntelligence, Los Angeles (1991)

42. Richard, B., Prié, Y., Calabretto, S.: Towards a unified model for audiovisual active reading. In: Tenth IEEE International Symposium on Multimedia, Berkeley, pp. 673–678 (2008)
43. Ryan, M.: Avatars of Story. University of Minnesota Press, Minneapolis (2006)
44. Saathoff, C., Schenk, S., Scherp, A.: Kat: The k-space annotation tool. In: SAMT 2008, Demo Session Proceedings. Springer, Berlin (2008)
45. Schuler, K.K.: Verbnet: a broad-coverage, comprehensive verb lexicon. Ph.D. thesis, University of Pennsylvania, Philadelphia (2005). AAI3179808
46. Snoek, C.G., Worring, M., van Gemert, J.C., Geusebroek, J.-M., Smeulders, A.W.: The challenge problem for automated detection of 101 semantic concepts in multimedia. In: Proceedings of ACM Multimedia, Santa Barbara, pp. 421–430 (2006)
47. Strauss, A., Corbin, J.: Basics of qualitative research: grounded theory procedures and techniques. Sage Publications, Newbury Park (1990)
48. Suchanek, F., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web, pp. 697–706. ACM, New York (2007)
49. Suchanek, F.M., Kasneci, G., Weikum, G.: YAGO: a core of semantic knowledge unifying WordNet and Wikipedia. In: WWW '07: Proceedings of the 16th International World Wide Web Conference, Banff, pp. 697–706 (2007)
50. Tonelli, S., Pighin, D.: New features for framenet – wordnet mapping. In: Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL'09), Boulder (2009)
51. Zhu, G., Yang, M., Yu, K., Xu, W., Gong, Y.: Detecting video events based on action recognition in complex scenes using spatio-temporal descriptor. In: Proceedings of the ACM Multimedia Conference, pp. 165–174. ACM, New York (2009)

Chapter 8
Knowledge in Action: Integrating Cognitive Architectures and Ontologies

Alessandro Oltramari and Christian Lebiere

Abstract In this work we present the *Cognitive Engine*, an integrated system whose architectural characteristics and operational capabilities are designed to approximate human visual intelligence. As humans usually do, the *Cognitive Engine* tries to make sense of a scene by meaningfully clustering visual data: basic individual movements are interpreted as constituting a particular action, and patterns of actions are gathered into more complex activities. In this respect, the *Cognitive Engine* results from augmenting the ACT-R cognitive architecture – a modular computational system used to model human cognitive processes – with relevant background knowledge embedded in HOMinE, a semantic resource for actions.

8.1 Introduction

> Representations of knowledge without an architecture are like programs without a computer – they do nothing. – [23], p. 18.

Humans can discriminate physical entities by their topological and morphological features, i.e. position, orientation, shape, configuration of proper parts, as well as at a more complex level, that is in terms of categories (person, nail, hammer), thematic roles (agent, patient, instrument), *gestalt schemas* (organized perceptual units that are not reducible to properties of their parts[1]), etc. Accordingly, 'visual intelligence' can be conceived as the human capability to identify events by means of the

[1] "A complex perception cannot be explained by the linear sum of the sensations that its parts arouse" [29], p. 118.

A. Oltramari (✉) · C. Lebiere
Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: aoltrama@andrew.cmu.edu; cl@cmu.edu

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_8,

relations between the entities in a scene, both at the perceptual and conceptual level. Human interpretation of events emerges as the result of intertwined perceptual, cognitive and inferential processes: reproducing this capability at the machine level requires a comprehensive infrastructure where low-level perceptual and high-level cognitive processes couple with knowledge representations. In this paper we focus on an integrated model – which we refer to as the *Cognitive Engine* – where the learning and knowledge retrieval mechanisms of the ACT-R cognitive architecture are combined with a semantic resource. We aim at integrating cognitive mechanisms and semantic contents to cope with the dynamics of knowledge flow in event recognition tasks, outlining a new application of research in ontologies and semantic resources in general. This research framework instantiates the general idea that cognitive systems benefit from both the mechanism-centered and knowledge-centered approaches to computationally achieve human level intelligence. The first approach, historically, has focused on general problem-solving programs [32] or architectures, i.e. [3, 31]. The second approach, partly arising from the limitations of the first, emphasized the knowledge of the system, especially common-sense knowledge, as the source of intelligence [24]. Those paradigms have encountered substantial successes in their own rights, but have up to now not achieved the ultimate goal of human-level intelligence. Moreover, both approaches have largely downplayed the other: systems that focus on mechanisms tend to treat knowledge as something to be engineered in ad hoc, task-specific ways, while those that focus on knowledge rely on narrowly tailored mechanisms to access and leverage their content, often raising unsustainable computational requirements in the process. Here we argue that those approaches are complementary, and that both of their central aspects, mechanisms and knowledge, need to be addressed systematically in Artificial Intelligence. Those two components strongly constrain each other, with learning mechanisms determining which knowledge can be acquired and in which form, and specific knowledge contents providing stringent requirements for mechanisms to be able to access them effectively [6]. In this chapter, we introduce each component, outline our proposal to combine them, and discuss an ongoing work to build a visual intelligent system. In particular, Sect. 8.2 illustrates the general framework of interaction between cognitive mechanisms (Sect. 8.2.1) and knowledge contents (Sect. 8.2.2), where human visual intelligence emerges as a unified psychological phenomenon stemming from both perceptual and conceptual structures (Sect. 8.2.3). In the attempt at modeling and simulating such intertwined features, in Sect. 8.3 we present the *Cognitive Engine*, the hybrid system resulting from the integration of ACT-R cognitive architecture and HOMinE, an ontology of actions (Sect. 8.3.1), whose joint objective is to *distill* action patterns from (possibly noisy) visual data (Sect. 8.3.3), reporting observations in a human-like linguistic format (Sect. 8.3.4). Finally, Sect. 8.4 provides an evaluation of *Cognitive Engine*, focusing on the most relevant features that have been implemented so far.

Currently available systems can be trained on specific tasks to identify a small set of physical entities, track their position and velocity, extract topological and morphological characteristics, i.e., orientation, shape, configuration of parts, etc. [18]; nevertheless, machines have not learned to bring that information together and understand the setting where objects are situated; in particular they cannot represent

the actions accomplished by those entities, such as bouncing, walking, picking up, etc. New research efforts have been recently directed towards developing automated programs

> that will be able to make sense of what they see. For example, AI programs would filter surveillance footage instead of a human, and automatically alert the police whenever any camera in the system spots something suspicious, such as someone leaving a package on a train platform or parking a car in an emergency zone and walking away [39].

The excerpt above efficaciously envisions the primary objective of the Mind's Eye program,[2] which is the context of our research project: developing visual intelligent systems to enhance video surveillance and support human operators. In the desired framework, visual intelligent systems should be able to reconstruct a "story" from basic information, blending relevant visual data with common-sense knowledge into a unifying conceptual pattern. Adopting an architectural perspective, implementing this capability would require three different strata of information elaboration: basic optical features (low-level), object detection (mid-level) and event classification (high-level). In this contribution we focus on high-level mechanisms and contents, namely the core visual intelligence as opposed to state-of-the-art machine vision.

8.2 Knowledge Mechanisms Meet Contents in Visual Intelligence

Cognitive architectures attempt to capture at the computational level the invariant mechanisms of human cognition, including those underlying the functions of control, learning, memory, adaptivity, perception and action. Accordingly, their goal is to model and simulate the dynamics of cognition, as opposed to *knowledge resources*, whose function is rather to properly encode and store the information that agents may need to access when interacting with the environment. In the following sections we focus on a particular cognitive architecture, ACT-R [5], introducing the general framework where ACT-R, integrated with a suitable knowledge resource, can adequately parse, disambiguate and describe visual information.

8.2.1 Mechanisms: Cognitive Architectures as Modules of Knowledge Production

ACT-R is a modular system: its components include perceptual, motor and memory modules, synchronized by a procedural module through limited capacity buffers (see Fig. 8.1). ACT-R has accounted for a broad range of cognitive activities at a

[2] http://www.darpa.mil/Our_Work/I2O/Programs/Minds_Eye.aspx

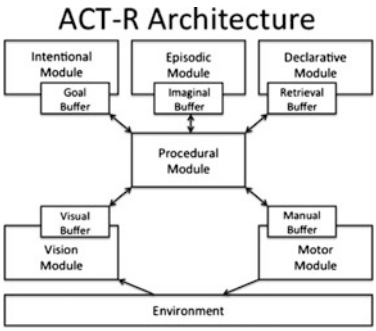

Fig. 8.1 Information from the environment is processed through the different modules of ACT-R

high level of fidelity, reproducing aspects of human data such as learning, errors, latencies, eye movements and patterns of brain activity. Declarative memory (DM) plays an important role in the ACT-R cognitive architecture. At the symbolic level, ACT-R models perform two major operations on DM: (1) accumulating knowledge units, so-called 'chunks', learned from internal operations or from interaction with the environment and (2) retrieving chunks that provide needed information (both chunk learning and retrieval are performed through a limited-capacity buffer called the 'retrieval buffer' that can only hold a single chunk at a time). Accumulation of symbolic knowledge triggers statistical learning processes that regulate subsymbolic (real-valued) activation quantities associated with those chunks that control their access during the retrieval process. The ACT-R theory distinguishes 'declarative knowledge' from 'procedural knowledge', the latter being conceived as a set of procedures (production rules) which coordinate information processing between its various modules (see [5], p. 26): according to this framework, agents accomplish their goals on the basis of (declarative) knowledge representations elaborated through procedural steps (in the form of 'if-then' clauses). This distinction between declarative and procedural knowledge is grounded in several experimental results in cognitive psychology regarding knowledge dissociation; major studies in cognitive neuroscience implicate a specific role of the hippocampus in "forming permanent declarative memories and the basal ganglia in production processes" (see [4], pp. 96–99, for a general mapping of ACT-R modules and buffers to brain areas and [42] for a detailed neural model of the basal ganglia's role in controlling information flow between cortical regions).

8.2.2 Contents: Ontologies as Declarative Knowledge Resources

Although discontinuously popular over the years, this separation between procedural and declarative knowledge has also been an important issue for AI. In 1980 John McCarthy first realized that, in order to enable full-fledged reasoning capabilities,

logic-based intelligent systems need to incorporate "re-usable declarative representations that correspond to objects and processes of the world"[28]. Along these lines, Patrick Hayes developed an axiomatic theory for naïve physics [22] and John Sowa [41] acknowledged the relevant role played by philosophy in defining a structured representation of world entities, i.e. an 'ontology'.[3] There have been numerous (and often alternative) attempts to define 'ontology' in AI: according to Guarino, "an ontology" is a language-dependent cognitive artifact, committed to a certain conceptualization of the world by means of a given language [21]. Besides the protocol layer, where the syntax of the communication language is specified, the ontological layer contains the semantics of that language: if concepts are described in terms of lexical semantics, ontologies take the simple form of dictionaries or thesauri; when ontological categories and relations are expressed in terms of axioms in a logical language, we talk about formal ontologies; if logical constraints are then encoded in a computational language, formal ontologies turn to computational ontologies. In the framework of cognitive architectures, ontologies play the role of "semantic specifications of declarative knowledge": in this contribution we propose to extend ACT-R with a scalable, reusable knowledge model that can be applied across a wide range of tasks, in particular for the purpose of action comprehension. Considering the state of the art, most research efforts have focused on designing methods for mapping large knowledge bases to the ACT-R declarative module (see [8, 9, 13, 14]). Here we commit on taking an integrated approach: instead of tying to a single ontology, we propose to build a hybrid computational ontology that combines different semantic dimensions of declarative representations.[4]

Our project consists in linking distinctive lexical databases, namely WordNet [16] and FrameNet [36] with a suitable computational ontology of actions, tying the resulting knowledge resource to ACT-R cognitive mechanisms for scene (visual) processing. Before presenting the core characteristics and functionalities of such an integrated system (see Sect. 8.3), we first need to outline the basic requirements it needs to fulfill in order to perform action understanding, analysing how this complex task is successfully accomplished by humans.

8.2.3 Human Visual Intelligence

Unfolding the relationships between visual processing and representations of events is an open problem for cognitive science. There is no clear explanation of how humans generalize over perceptual contents and create conceptualizations of

[3]This was the genesis of using the word 'ontology' in AI. *Ontology*, 'the study of being as such' – as Aristotle named it – in fact originated as a philosophical discipline.

[4]The adjective 'hybrid' is used to emphasize the heterogeneity of resources we are adopting for the purposes of the project. For a general survey on hybrid semantic approaches see [35]. For the sake of readability we will henceforth omit the mid-adjective computational.

events [12], which can be communicated in propositional form by natural language ("when the bus stopped at Forbes Avenue, John got off, crossed the street and entered in the pub") and used for inferential reasoning (John was riding the bus). In [43] and [10], authors pointed out that, as similar objects have a high degree of overlapping components (scissors and knives, chairs and tables, etc.), so similar events do share constitutive temporal parts (i.e. burying an object and digging soil, swimming and boating, writing and reading, etc.). Similarity between events is also dependent on distribution of related objects:

> if there's a refrigerator, there's probably a sink and a stove nearby. Objects that serve related ends typically appear together in contexts, specifically in scenes.

As [44] acknowledges in the previous passage (p. 445), "events are understood as action-object couplets" (p. 456) and "segmenting [events as couplets] reduces the amount of information into manageable chunks" (p. 457), where the segment boundaries coincide with achievements and accomplishments of goals (p. 460). The notion of segmentation is even more crucial if scene information processing is the focus: recognition doesn't correspond to an inventory of all the actions occurring in a scene. A selection process is performed by means of suitable 'cognitive schemas' (or *gestalts*, e.g. up/down, figure/ground, force, etc.), which carve visual presentations according to principles of mental organization and optimize the perceptual effort" [1]. Besides cognitive schemas, conceptual primitives have also been studied: in particular, [40] applied Hayes' naïve physics theory [22] to build an event logic. Within the adopted common sense principles, we can mention (i) *substantiality* (objects generally cannot pass through one another); (ii) *continuity* (objects that diachronically appear in two locations must have moved along the connecting path); (iii) *ground plane* (ground acts as universal support for objects).

As far as action-object pairs are central to characterize the 'ontology of events', verb-noun 'frames' are also relevant at the linguistic level[5]; in particular, identifying roles played by objects in a scene is necessary to disambiguate action verbs and highlight the underlying goals. In this respect, studies of event categorization revealed that events are always *packaged*, that is distinctly equipped with suitable semantic roles [26]: for example, the events which are exemplified by motion verbs like walk, run, fly, jump, crawl, etc. are generally accompanied with information about source, path, direction and destination/goal, as in the proposition "John ran out of the house (*source*), walking south (*direction*) along the river (*path*), to reach Emily's house (*destination/goal*)"; conversely, verbs of possession such as have, hold, carry,get, etc. require different kind of semantic information, as in the proposition "John (*owner*) carries Emily's bag (*possession*)". Note that it is not always the case that all possible semantic roles are filled by linguistic phrases: in particular, *path* and *direction* are not necessarily specified when motion

[5] We refer here to the very broad notion of 'frame' introduced by Minsky: "frames are data-structure for representing a stereotyped situation, like being in a certain kind of living room, or going to a child's birthday party" [30].

is considered, while *source* and *destination/goal* are (we do not focus here on agent and patient which are the core semantic roles).

As this overview suggests, there is an intimate connection between action understanding and cognition, both at the level of scene parsing (mechanism) and knowledge representation (ontology). In particular, in order to assess the cognitive soundness of a (human or artificial) visual intelligent system, four basic features need to be addressed:

- **Ontological similarity**: similar events are structured by the same component actions and states (see Sect. 8.3.2);
- **Conceptual packaging**: actions can be represented insofar that roles played by objects in a scene are identified (e.g., agent, patient, source, goal, instrument, duration, etc.).
- **Intentionality**: actions are characterized by goals that agents aim at satisfying;
- **Cognitive selectivity**: attentional mechanisms drive the visual system in detecting the causal aspects of a scene, focusing on the most distinctive actions and discarding accidental events.

The next sections present the *Cognitive Engine*, an integrated artificial system whose architectural characteristics, operational capabilities and knowledge resources are designed to approximate the cognitive machinery of visual intelligence.

8.3 *Making Sense* of Visual Data

Cognitive adequacy is a fundamental requisite that effective visual systems need to realize. Making sense of visual data means to be able to represent their semantic content. Reproducing this capability at the machine level requires a comprehensive infrastructure where low–level perceptual and high–level cognitive processes couple with knowledge representations: for example, basic body movements and physical interactions such as e.g., bending–over, extending–arm, holding, carrying (a manageable object for a given amount of time), etc. are interpreted as constituting the necessary stages for accomplishing a specific kind of action (e.g., hauling an object), and patterns of actions can be also gathered to assess more complex activities (hauling an object and giving it to another person as parts of 'exchange' action). Here we present the *Cognitive Engine* system,[6] where the learning and knowledge retrieval mechanisms of the ACT-R cognitive architecture are combined with HOMinE.

[6]Henceforth abbreviated with *CE*.

8.3.1 HOMinE: Model and Implementation

Ontologies play the role of 'semantic specifications of declarative knowledge' in the framework of cognitive architectures. As [8, 9, 13, 14] demonstrate, most research efforts have focused on designing methods for mapping large knowledge bases to the ACT-R declarative module. Here we commit on taking a different approach: instead of tying to a single ontology, we built a hybrid computational ontology that combines different semantic dimensions of declarative representations. Our proposal consists in linking distinctive lexical databases, i.e. WordNet [16] and FrameNet [36] with a suitable computational ontology of events, tying the resulting semantic resource to ACT-R cognitive mechanisms (see Sect. 8.3.2). A multi-level representation of events is needed to elicit and formalize their semantics: understanding the internal structure of events is necessary for enabling mechanism of action–recognition. Ontologies can specify meaning at different levels, depending on the focus of representation (language, hierarchical organization, logics, etc.): a full-fledged ontological model, in this sense, should include a comprehensive set of those semantic features.

Grounded on these design principles, HOMinE (Hybrid Ontology for the Mind's Eye project) exploits DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) top–level [27], which has been developed in order to address some core cognitive and linguistic features of common–sense knowledge, as a general model for aligning WordNet (WN) and FrameNet (FN), following the line of research of [19]: Fig. 8.2 shows some selected nodes of DOLCE backbone taxonomy, where the notion of ACTION is the main anchor to HOMinE. The root of the class hierarchy of DOLCE is ENTITY, which is defined as the class of anything that is identifiable by humans as an object of experience or thought. The first distinction is among CONCRETE ENTITY, i.e. the class of objects located in definite spatial regions, and ABSTRACT ENTITY, including objects that don't have proper spatial properties. In the line of [38], CONCRETE ENTITY is further distinguished into CONTINUANT and OCCURRENT, that is, roughly, entities without temporal parts (e.g. artefacts, animals, substances) and entities with temporal parts (e.g. events, actions, states) respectively. If DOLCE provides the axiomatic basis for the formal characterization of HOMinE,[7] SCONE is the selected framework of implementation.[8] SCONE is an open–source Knowledge-Base system intended for use as a component in many different software applications: it provides a LISP-based framework to represent and reason over symbolic common–sense knowledge. Unlike most diffuse KB systems, SCONE is not based on OWL (Ontology Web Language[9]) or Description Logics in general [7]: its inference

[7] For instance, DOLCE adapts Allen's temporal axioms [2], which are considered as state of the art in temporal representation and reasoning.

[8] http://www.cs.cmu.edu/~sef/scone/

[9] http://www.w3.org/TR/owl-features/

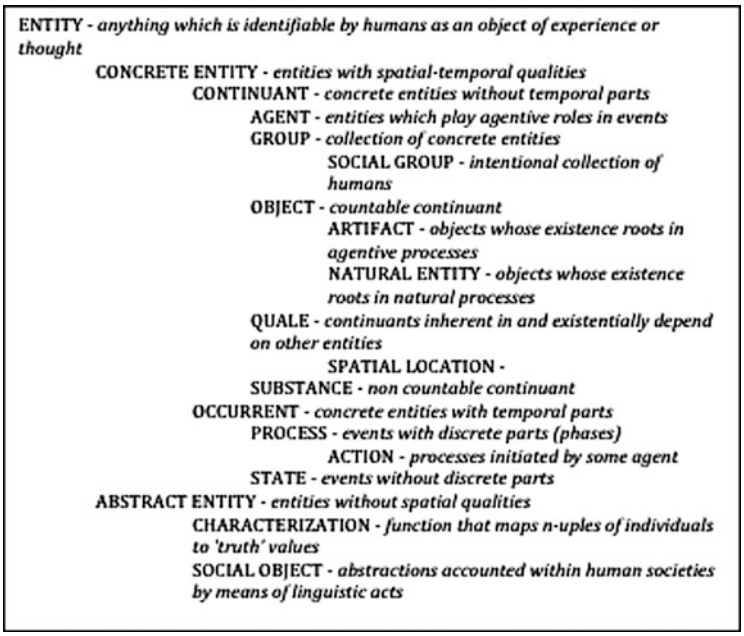

Fig. 8.2 An excerpt of DOLCE top level

engine adopts marker–passing algorithms [15] (originally designed for massive parallel computing) to perform fast queries at the price of losing logical completeness and decidability. In particular, SCONE represents knowledge as a *semantic network* whose nodes are locally weighted (*marked*) to optimize basic reasoning tasks, e.g. checking inherited properties, class membership, transitivity, using spreading of activation.[10] SCONE revealed to suitably support the task of retrieving degraded or incomplete information for of action understanding: the modularization and implementation of HOMinE with SCONE allowed for a straightforward logical modeling and inferencing of core ontological properties of events, such as: (i) participation of actors and objects in actions; (ii) temporal features based on the notions of 'instant' and 'interval'; (iii) common-sense spatial information.[11]

HOMinE's conceptual layer is based on a partition of WN related to verbs of action, such as walk, pick-up, haul, kick, chase, etc. WN is a semantic network whose nodes and arcs are, respectively, synsets ("sets of synonym terms") and semantic relations. Over the years, there has been an incremental growth of the

[10]Far from willing to deepen a topic that is out of scope to treat in our contribution, we refer the reader to [15] for details concerning marker–passing algorithms. Note that these inference mechanisms in SCONE are generally consistent with the activation-based retrievals mechanisms in ACT-R, raising an additional level of compatibility between the two frameworks.

[11]We will describe in Sect. 8.3.2 how SCONE functions as bridging component between ACT-R cognitive architecture and HOMinE knowledge resource.

lexicon (the latest version, WordNet 3.0, contains about 117 K synsets), and substantial enhancements aimed at facilitating computational tractability. In order to find the targeted group of relevant synsets, we basically started from two pertinent top nodes,[12] move#1 and move#2.[13] As one can easily notice, the former synset denotes a change of position accomplished by an agent or by an object (with a sufficient level of autonomy), while the latter is about causing someone or something to move (both literally and figuratively). After extracting the sub–hierarchy of synsets related to these generic verbs of action, we introduced a top-most category 'movement-generic', abstracting from the two senses of 'move' (refer to Fig. 8.3 for the resulting taxonomy of actions).

FrameNet (FN) is the additional conceptual layer of HOMinE. Besides wordnet-like databases, a computational lexicon can be designed from a different perspective, for example focusing on frames, to be conceived as orthogonal to domains. Inspired by frame semantics [17], FN aims at documenting "the range of semantic and syntactic combinatory possibilities (valences) of each word in each of its senses" through corpus-based annotation. Different frames are evoked by the same word depending on different contexts of use: the notion of 'evocation' helps capturing the multi-dimensional character of knowledge structures underlying verbal forms. For instance, if you point to the *bringing* frame, namely an abstraction of a state of affairs where sentient agents (e.g., persons) or generic carriers (e.g. ships) bring something somewhere along a given path, you will find several 'lexical units' (LUs) evoking different roles (or frame elements – FEs): i.e., the noun 'truck' instantiates the 'carrier' role. In principle, the same Lexical Unit (LU) may evoke distinct frames, thus dealing with different roles: 'truck', for example, can be also associated to the vehicle frame ('the vehicles that human beings use for the purpose of transportation'). FN contains about 12 K LUs for 1 K frames annotated in 150,000 sentences.

WN and FN are based on distinct models, but one can benefit from the other in terms of coverage and type of information conveyed. Accordingly, we have analyzed the evocation-links between the action verbs we have extracted from WN and the related FN frames: those links can be generated through 'FN Data search', an on–line navigation interface used to access and query FN.[14] Using a specific algorithm [11], WordNet synsets can be associated with FrameNet frames, ranking the results by assigning weights to the discovered connections [33]. The core mechanism can be resumed by the following procedure: first of all the user has to choose a term and look for the correspondent sense in WordNet; once the correct synset is selected, the tool searches for the corresponding lexical units (LUs) and frames of FrameNet.

[12] AKA Unique Beginners [16].

[13] 01835496 move#1, travel#1, go#1, locomote#1 (change location; move, travel, or proceed) "How fast does your new car go?"; "The soldiers moved towards the city in an attempt to take it before night fell". 01850315 move#2, displace#4 (cause to move or shift into a new position or place, both in a concrete and in an abstract sense) "Move those boxes into the corner, please"; "The director moved more responsibilities onto his new assistant".

[14] https://framenet.icsi.berkeley.edu/fndrupal/index.php?q=luIndex

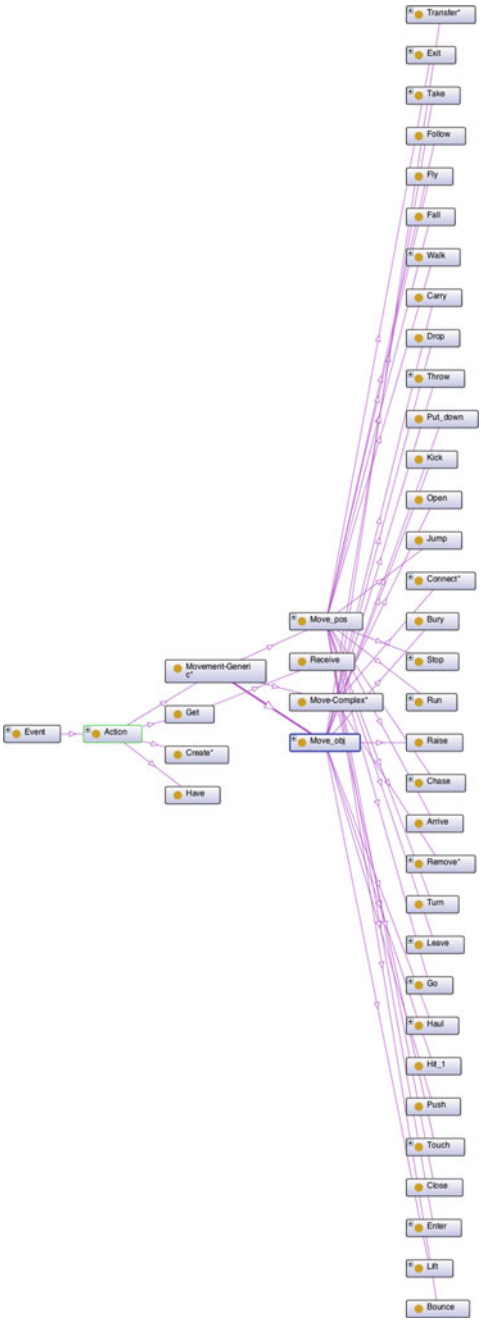

Fig. 8.3 HOMinE backbone taxonomy

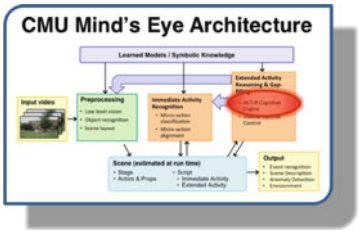

Fig. 8.4 CMU Mind's eye architecture

Afterwards, all candidate frames are weighted according to three important factors: the similarity between the target word (the LU having some correspondence to the term typed at the beginning) and the wordnet relative (which can be the term itself – if any – and/or its synonyms, hypernyms and antonyms); a variable boost factor that rewards words that correspond to LU as opposed to those that match only the frame name; the spreading factor, namely the number of frames evoked by that word:

$$\frac{similarity(wordnet\_relative, target\_word) * BoostFactor}{spreading\_factor(wordnet\_relative)}$$

In summary, our work led to a conceptual enrichment of declarative structures for basic action types: starting from WN synset information, and using FN data, we could identify typical roles and fillers of those verbs, logically constraining them to the HOMinE ontology encoded in SCONE framework. The effectiveness of this knowledge resource will emerge more clearly in the next section where, on the basis of the isomorphism between the elements of ACT-R chunks, namely slots and associated values, with FrameNet elements (roles) and fillers (WordNet synsets), we present an ACT-R model for action understanding, whose declarative memory has been specified with HOMinE's semantic layers.

8.3.2 The Cognitive Engine

The *CE* represents the core module of the Extended Activity Reasoning system (EAR) in the CMU-Minds Eye architecture (see Fig. 8.4). EAR receives outputs from the Immediate Activity Recognition system (IAR), which collects the results of different preprocessing algorithms and adopts learning–based methods to elaborate action probability distributions [25].

In the context of EAR, specific functions have been designed to extract and merge relevant information from the IAR output in order to feed the *CE* with suitable sequences of quasi-propositional descriptions (e.g., Person1-grasp-Bag2 + Person1 hold-Bag2 + Person1-drop-Bag2 + Bag2-on-Table3).[15] The *CE* is the result of

[15]These sequences reflect the most likely atomic events (so called 'micro-actions', 'micro-states' and'micro-poses') occurring in the environment, detected and thresholded by machine vision algorithms. The addition symbol exemplifies temporal succession while numbers stand for entity

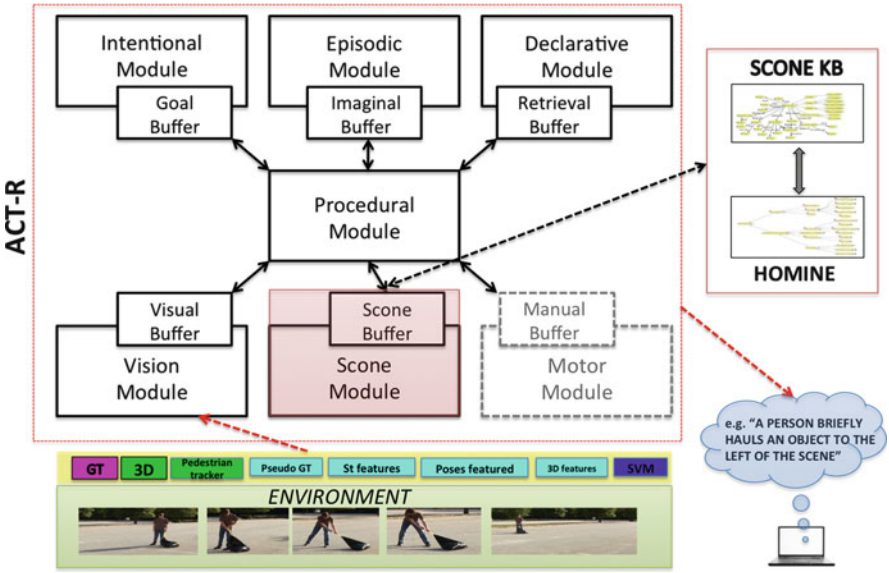

Fig. 8.5 A Diagram of the Cognitive Engine (colored boxes in the *bottom* represents the preprocessing algorithms and IAR system)

augmenting ACT-R with HOMinE. As Fig. 8.5 shows, we exploited the flexible nature of ACT-R to engineer a SCONE-MODULE as a bridging component between the cognitive architecture and the hybrid ontology. In this context, after an internal elaboration of the information input, *CE* produces two different outputs: (i) the identification of the correct action patterns ('recognition task') and (ii) a human–like textual report of the actions occurring in the scene ('description task'). *CE* is able to overcome situations with missing or corrupted input: ACT-R mechanisms of partial matching and spreading activation [5] can fill the gap(s) left by the missing atomic events and retrieve the best–matching action pattern. In the next subsections we describe in more details how *CE* performs both the recognition and description tasks.

8.3.3 Recognition Task

In the recognition task, visual intelligent systems have to process an evaluation dataset of videos[16] and output the probability distribution (per video) of a pre-defined list of 50 action verbs. Performance is measured in terms of consistency with

unique identifiers. For the sake of readability, we omit here the temporal information about start and end frames of the single atomic-events, as well as spatial coordinates of the positions of objects.

[16] http://www.visint.org/datasets.html. The description task applies to the same dataset.

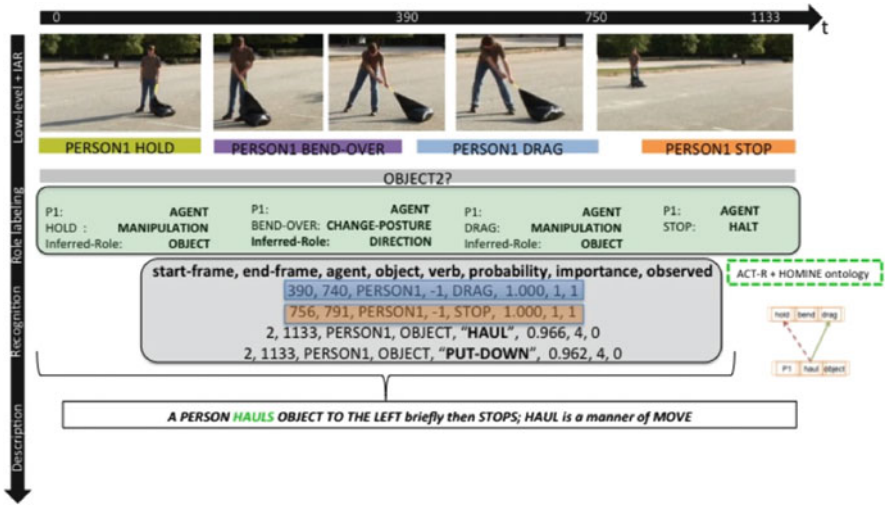

Fig. 8.6 EAR processing schema

human responses to stimuli: subjects have been asked to judge the presence/absence of every verb in each video. In order to meet these criteria, we devised the *CE* to work in a human-like fashion, trying to disambiguate the scene in terms of the most reliable conceptual structures: more specifically, it parses the atomic events extracted by IAR, so called micro-actions (e.g., extend-arm, walk, fill-with-tool, bend-over, etc.) and micro-states (e.g., be-in, be-near, sitting-on-object, holding, etc.), associating frames and roles to visual input from the videos. This specific information is retrieved from the FrameNet module of HOMinE: frames and frame roles are assembled in suitable knowledge units and encoded in the declarative memory of an appropriate ACT-R model. As with human annotators performing semantic role labeling [20], *CE* associates verbs denoting micro-actions and micro-states to corresponding frames. When related productions fire, the model retrieves the chunks corresponding to the roles played by the entities in the scene, for each atomic event. In order to prompt a choice within the available patterns of action, spreading activation is exploited through the ACT-R sub-symbolic computations [5]. Spreading of activation from the contents of verb roles triggers the evocation of chunks related to pattern components in the context of the perceived scene. The *CE* tries to make sense of a scene, clustering visual data according to semantic criteria, outputting the most likely verbs out of the predefined ones. The actual output is a time course of activation of the various patterns of action as the model is presented with a sequence of temporal frames (see Fig. 8.6). Partial matching based on similarity measures and spreading of activation based on compositionality are the main mechanisms used by *CE* to perform efficient action recognition. Base-level activations of verbs actions have been derived by frequency analysis of the American National Corpus. In addition, we constrain semantic similarity within verbs to the 'gloss-vector' measure computed over [34]. Finally, strengths of associations are

$$A_i = \ln \sum_j t_j^{-d} + \sum_k W_k S_{ki} + \sum_l MP_l Sim_{il} + N(0,\sigma)$$

Fig. 8.7 Equation for Bayesian activation pattern matching

set (or learned) by the architecture to reflect the number of patterns to which each micro-action is associated, the so-called 'fan effect' controlling information retrieval in many real-world domains [37].

Specifically, ACT-R can evaluate and identify the most likely action patterns by means of its core sub-symbolic computations, expressed by equation in Fig. 8.7:

- First term: the more recently and frequently a chunk has been retrieved, the higher its activation and the chances of being retrieved. In our context *i* can be conceived as a pattern of action (e.g., the pattern of HAUL), where *t* is the time elapsed since the jth reference to chunk *i* and *d* represents the memory decay rate.
- Second term: the contextual activation of a chunk *i* is set by the attentional weight given the element *k* and the strength of association between an element *k* and the *i*. In our context, k can be interpreted as the value BEND-OVER of the pattern HAUL in Fig. 8.6.
- Third term: under partial matching, ACT-R can retrieve the chunk *l* that matches the retrieval constraints to the greatest degree, computing the similarity between *l* and *i* and the mismatch score MP. In our context, for example, the chunk PULL could have been retrieved, instead of DRAG. This characteristics is particularly efficient when slot-values are dynamically changing – as in the case of a continuous visual input stream.
- Fourth term: randomness in the retrieval process by adding Gaussian noise.

8.3.4 Description Task

As for the previous task, the system's capabilities have to be evaluated according to human ground-truth, i.e. naturally rich and detailed textual descriptions. To meet these requirements, the format of descriptions produced by *CE* reflects a constrained natural language, whose main units are subject, verb, object, temporal index and spatial index (e.g., "A person picked-up an object then exited to the left of the scene"). The step-by-step reasoning over patterns of activation is exploited to fill the agent and patient thematic roles of the detected verb, corresponding respectively to subject(s) and object(s) (if any) of the verb(s) in the presented sequence. Start-frame and end-frame are also extracted from the recognition output and used at the level of description to identify the temporal intervals in which the classified actions occur: on the basis of a context-sensitive threshold,[17] some temporal intervals are mapped

[17]The qualitative duration of an interval may vary with the circumstances: accordingly, we are working on adding a mechanism for data-driven context sensitivity to *CE*.

to the adverbial phrase ‘briefly’, while others are mapped to the locution ‘for long’. Besides relative time measures, qualitative spatial information can be also added to textual descriptions by computing the difference along the x-axis and y-axis of the initial and final positions of tracked objects. In particular, this information is used to identify the basic direction of motion with respect to the camera (the observer view). Accordingly, the following heuristics have been defined: (i) if, for two different positions in the x-axis, the value of the difference is negative, then the final position of a given object is to the right of the initial position, else is to the left; (ii) if, for two different positions in y-axis, the value of the difference is negative, then the final position of a given object is on the top of the scene, else is on the bottom. Moreover, if the final y-position of a moving object is close to the top of the footage, eventually disappearing, *CE* is able to add a camera-centric information to the description output, namely ‘away’ from the scene, as in ‘the person walked away’. Modeled to reflect these general criteria, specific functions and production rules have been implemented in *CE* to output suitable textual descriptions. The next section illustrates some significant results. For reasons of brevity, we choose to focus on the performance of *CE* only for the description task. However, given the architectural dependence between the recognition and description, positive and negative trends are generally shared by the two tasks.

8.4 Evaluation

On the basis of human annotations of a subset of videos selected from the evaluation dataset, we ran about 20 trials for the description evaluation.[18] The resulting description outputs have been compared to the textual ground truth provided by a designated group of viewers (henceforth, abbreviated as ‘GTD’). Given syntactic variations, the challenging issue underlying this task is to provide measures of semantic coherence between descriptions: we adopt some simple methods to assess accuracy, completeness and importance. **Accuracy** is defined as ‘yes/no’ answer by a single human subject on whether the description output fits at least some part of the semantic content of GTD. A negative answer – as follows from the above assumptions – corresponds to a failure in the reasoning mechanisms at the EAR level.

Completeness has an articulated structure: it is a percentage measure calculated on the basis of the ratio between detected verbs used in the description outputs and the overall number of distinct verbs used in GTD, plus (i) 10 %, if the detected verb is present according to the annotators but not used in GTD[19]; (ii) 10 %, if the classes used to instantiate objects in the scene are the same across EAR and GTD (e.g., ‘truck’ ‘truck’); (iii) 5 %, if the class used by EAR system to instantiate

[18]The multiple runs are motivated by the need to reflect the stochasticity of the ACT-R architecture, specifically in information retrieval.

[19]To reward the match between machine output and human annotations used for training.

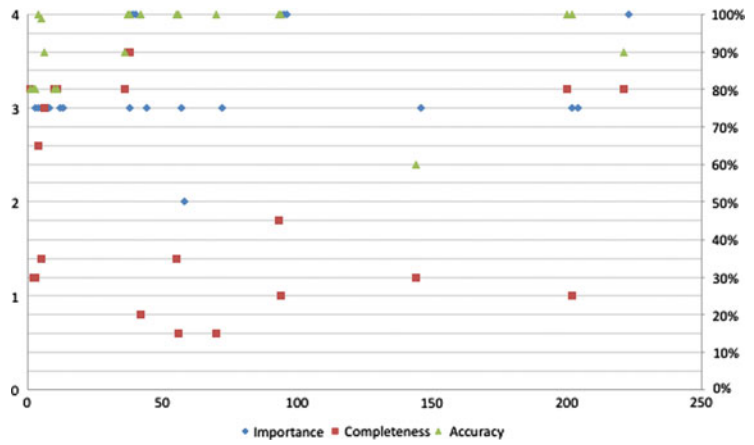

Fig. 8.8 Description performance

objects in the scene is the superordinate category of the class used GTD (e.g., 'vehicle' 'truck').[20] Note that these percentage add-ons have been properly weighted to magnify semantic effects on the experimental data. Accordingly, the agreement between annotators and the use of semantically correlated object categories have been rewarded and factorized in the computation of completeness. Finally, we measure **importance**, which reflects the level of reasoning required for recognizing a verb (represented by a 1–4 scale, equivalent to the number of atomic events required to activate a pattern of action). Considering the limited set of videos that we have focused on due to the labor-intensive nature of the evaluation, the level of accuracy is overwhelmingly positive (95 %), while completeness averages 53 %. Figure 8.8, which displays detailed variation over the stimuli set, reveals that the completeness average is in fact composed of a bimodal distribution between almost-complete (80 % or better) and much more partial (around 30 %). Despite the moderate percentage of completeness, the importance of the verbs used by *CE* in the description task is mostly 3, which demonstrates that the output refers to a reasonably high level of conceptual knowledge. This result can be explained by the fine-grained causal descriptions that ground truth can provide in some cases as opposed to the coarser level of complexity that *CE* can deal with (basic reports of what is occurring at given time-intervals). Compare, for example, the following description in GTD as opposed to machine outputs: (a) 'a woman on a bicycle stops and puts her feet down on the ground'; (b) 'PERSON2 ARRIVE PLACE from the left of the scene". Note that *CE* output refers to 'a person arriving' but not to the 'a person (entering into the scene) riding a bike', as observed by human viewers. Also, GTD indicates that the person put her feet down, which is a necessary movement to regain equilibrium when standing still on a bike, while *CE* could not prompt such

[20]Only (ii) and (iii) are mutually exclusive.

a piece of knowledge. Without going in depth in a topic that is intimately related to cognitive linguistics, we can mention that the main reason behind this divergence between the outputs is not a mere lack of semantic encoding by the earlier system components but the fact that 'putting down the feet' is causally salient and the observer suitably decided to render it in the free-text account of the scene because of the intentionality and cognitive selectivity principles (see Sect. 8.2.3), which are currently not supported by *CE*. The experimentation results for the description task demonstrate that intentionality is one of the most important issues to be addressed in the next phase of the project.

8.5 Conclusions and Future Work

In this chapter we presented the *Cognitive Engine*, a high-level artificial visual intelligent system. The soundness of the *Cognitive Engine* depends on ontological similarity and conceptual packaging, as defined in Sect. 8.2.3: the systems adequacy to these two principles has been illustrated in terms of disambiguating visual information on the basis of general patterns of actions, exploiting the dynamic integration between ACT-R knowledge mechanisms and HOMinE knowledge resource. Future work will be devoted to improve the system using reasoning and statistical inferences to derive and predict goals of agents (addressing the principle of intentionality) and mechanisms of abduction to focus on the most salient information from complex visual streams. We also plan to extend the *Cognitive Engine*'s functionalities to support a comprehensive range of action verbs: in this sense, the performance of the system will be tested on a higher scale of ambiguity than what has been shown in Sect. 8.3.3.

Acknowledgements This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0061. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

1. Albertazzi, L., Van Tonder, L., Vishwanath, D. (eds.): Perception Beyond Inference. The Information Content of Visual Processes. MIT, Cambridge (2010)
2. Allen, J.F.: An interval based representation of temporal knowledge. In: 7th International Joint Conference on Artificial Intelligence, vol.1, pp. 221–226. IJCAI/Morgan Kaufmann, Vancouver (1983)
3. Anderson, J.: The Architecture of Cognition. Harvard University Press, Cambridge (1983)
4. Anderson, J.: How Can the Human Mind Occur in the Physical Universe? Oxford University Press, Oxford/New York (2007)

5. Anderson, J., Lebiere, C.: The Atomic Components of Thought. Erlbaum, Mahwah (1998)
6. Anderson, J., Lebiere, C.: The newell test for a theory of cognition. Behav. Brain Sci. **26**, 587–637 (2003)
7. Baader, F., Calvanese, D., Mcguinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, Cambridge/New York (2003)
8. Ball, J., Rodgers, S., Gluck, K.: Integrating act-r and cyc in a large-scale model of language comprehension for use in intelligent systems. In: Papers from the AAAI workshop, pp. 19–25. AAAI, Menlo Park (2004)
9. Best, B., Gerhart, N., Lebiere, C.: Extracting the ontological structure of cyc in a large-scale model of language comprehension for use in intelligent agents. In: Proceedings of the 17th Conference on Behavioral Representation in Modeling and Simulation (2010). http://www.adcogsys.com/pubs/Brims2010-best-gerhart-lebiere-opencyc.pdf
10. Biederman, I.: Recognition by components: a theory of human image understanding. Psychol. Rev. **94**, 115–147 (1987)
11. Burchardt, A., Erk, K., Frank, A.: A WordNet detour to FrameNet. In: Schröder, B., Fisseni, B. Schmitz, H.-C., Wagner, P. (eds.) Sprachtechnologie, Mobile Kommunikation und Linguistische Resourcen. Computer Studies in Language and Speech, vol. 8, pp. 408–421. Peter Lang, Frankfurt am Main (2005)
12. Casati, R., Varzi, A. (eds.): Events. Dartmouth, Aldershots (1996)
13. Douglas, S., Ball, J., Rodgers, S.: Large declarative memories in ACT-R. In: Proceedings of the 9th International Conference of Cognitive Modeling, Manchester (2009)
14. Edmond, B.: Wn-lexical: an ACT-R module built from the wordnet lexical database. In: Proceedings of the 7th International Conference of Cognitive Modeling, Trieste, pp. 359–360 (2006)
15. Fahlman, S.: Using Scone's multiple-context mechanism to emulate human-like reasoning. In: First International Conference on Knowledge Science, Engineering and Management (KSEM'06). Lecture Notes in Artificial Intelligence. Springer, Guilin (2006)
16. Fellbaum, C. (ed.): WordNet, an Electronic Lexical Database. MIT, Boston (1998)
17. Fillmore, C.J.: The case for case. In: Bach, E. Harms, T. (eds.) Universals in Linguistic Theory. Rinehart and Wiston, New York (1968)
18. Forsyth, D.A., Ponce, J.: Computer Vision, a Modern Approach. Prentice Hall, New Jersey, USA (2004)
19. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Sweetening WordNet with DOLCE. AI Mag. **3**, 13–24 (Fall 2003)
20. Gildea, D., Jurafsky, D.: Automatic labelling of semantic roles. In: Proceedings of 38th Annual Conference of the Association for Computational Linguistics (ACL-00), pp. 512–520. ACL, San Francisco (2000)
21. Guarino, N.: Formal ontology in information systems. In: Guarino, N. (ed.) Formal Ontology in Information Systems. Proceedings of FOIS98, Trento, pp. 3–15. IOS, Amsterdam (1998)
22. Hayes, P.J.: The second naïve physics manifesto. In: Hobbes, J., Moore, R. (eds.) Formal Theories of the Common Sense World. Ablex Publishing Corporation, Norwood (1985)
23. Laird, J.E.: The SOAR Cognitive Architecture. MIT, Cambridge/London (2012)
24. Lenat, D.G., Prakash, M., Sheperd, M.: Cyc: using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. Artif. Intell. **6**(4), 65–85 (1985)
25. Maitikanen, P., Sukthankar, R., Hebert, M.: Feature seeding for action recognition. In: Proceedings of International Conference on Computer Vision. IEEE, Piscataway (2011)
26. Majid, A., Boster, J., Bowerman, M.: The cross-linguistic categorization of everyday events: a study of cutting and breaking. Cognition **109**, 235–250 (2008)
27. Masolo, C., Gangemi, A., Guarino, N., Oltramari, A., Schneider, L.: WonderWeb deliverable D17: the Wonderweb library of foundational ontologies. Tech. rep. (2002)
28. McCarthy, J.: Circumscription – a form of non-monotonic reasoning. Artif. Intell. **5**(13), 27–39 (1980)

29. Miller, G.A., Buckhout, R.: Psychology: The Science of Mental Life. Harper & Row, New York (1973)
30. Minsky, M.: A framework for representing knowledge. In: Winston, P. (ed.) Mind Design, pp. 111–142. MIT, Cambridge (1997)
31. Newell, A.: Unified Theories of Cognition. Harvard University Press, Cambridge (1990)
32. Newell, A., Shaw, J., Simon, H.: Report on a general problem-solving program. In: Proceedings of the International Conference on Information Processing, pp. 256–264. Unesco, Paris (1959)
33. Oltramari, A.: Lexipass methodology: a conceptual path from frames to senses and back. In: LREC 2006 (Fifth International Conference on Language Resources and Evaluation). ELDA, Genoa (2006)
34. Pedersen, T., Patwardhan, S.J., Michelizzi, M.: Wordnet: similarity: measuring the relatedness of concepts. In: Demonstration Papers at HLT-NAACL, pp. 38–41. ACL. East Stroudsburg (2004)
35. Prevót, L., Huang, C.R., Calzolari, N., Gangemi, A., Lenci, A., Oltramari, A. (eds.): Ontology and the Lexicon. Cambridge University Press, Cambridge/New York (2010)
36. Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C.: Framenet: Theory and Practice. ICSI, Berkeley (2005)
37. Schooler, L., Anderson, J.: The disruptive potential of immediate feedback. In: Proceedings of the Twelfth Annual Conference of The Cognitive Science Society, pp. 702–708. Erlbaum, Hillsdale (1990)
38. Simons, P. (ed.): Parts: A Study in Ontology. Clarendon, Oxford (1987)
39. Singer, P.W.: Wired for War. Penguin, New York (2009)
40. Siskind, J.M.: Grounding language in perception. Artif. Intell. Rev. **8**, 371–391 (1995)
41. Sowa, J.F.: Conceptual Structures. Information Processing in Mind and Machine. Addison Wesley, Reading (1984)
42. Stocco, A., Lebiere, C., Anderson, J.: Conditional routing of information to the cortex: a model of the basal ganglia's role in cognitive coordination. Psychol. Rev. **117**(2), 541–574 (2010)
43. Tversky, A.: Features of similarity. Psychol. Rev. **84**, 327–352 (1977)
44. Tversky, B., Zachs, J., Martin, B.: The structure of experience. In: Shipley, T., Zacks, T. (eds.) Understanding Events: From Perception to Action, pp. 436–464. Oxford University Press, Oxford/New York (2008)

Part III
Enhancing NLP with Ontologies

Chapter 9
Use of Ontology, Lexicon and Fact Repository for Reference Resolution in Ontological Semantics

Marjorie McShane and Sergei Nirenburg

Abstract This chapter presents an implemented algorithm for resolving reference within the theory of Ontological Semantics with an emphasis on the use of static knowledge resources: ontology—a world model of entity types; fact repository—a world model of entity tokens; and lexicon, which mediates between language and the ontology and fact repository. We show how reference resolution is tightly coupled with overall semantic analysis, from the first stages of determining which expressions have referential function to the final stage of creating a reference link from each referring expression in a text to its "anchor" in the model of memory of the intelligent agent processing the text. As such, there is no single reference resolution task; rather, reference-related subtasks are best distributed throughout an end-to-end text analysis system.

9.1 Introduction

Computer tractable knowledge resources such as lexicon, ontology and fact repository are only as good as the processing they can support in useful applications. As such, resources are best evaluated in the context of their use: when are they used, how are they used, how well do they fulfill their envisioned roles, and how could they be modified to better fulfill those roles? In this chapter, we address these questions with respect to OntoSem [25, 36] knowledge resources in support of the task of reference resolution as carried out by multi-functional, dialog-enabled intelligent agents.

The chapter begins with two background sections: Sect. 9.2 defines what reference resolution means for intelligent agents, using a definition that markedly contrasts with the narrow **co**reference task typically pursued in implemented

M. McShane (✉) · S. Nirenburg
University of Maryland, Baltimore County, MD, USA

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_9, © Springer-Verlag Berlin Heidelberg 2013

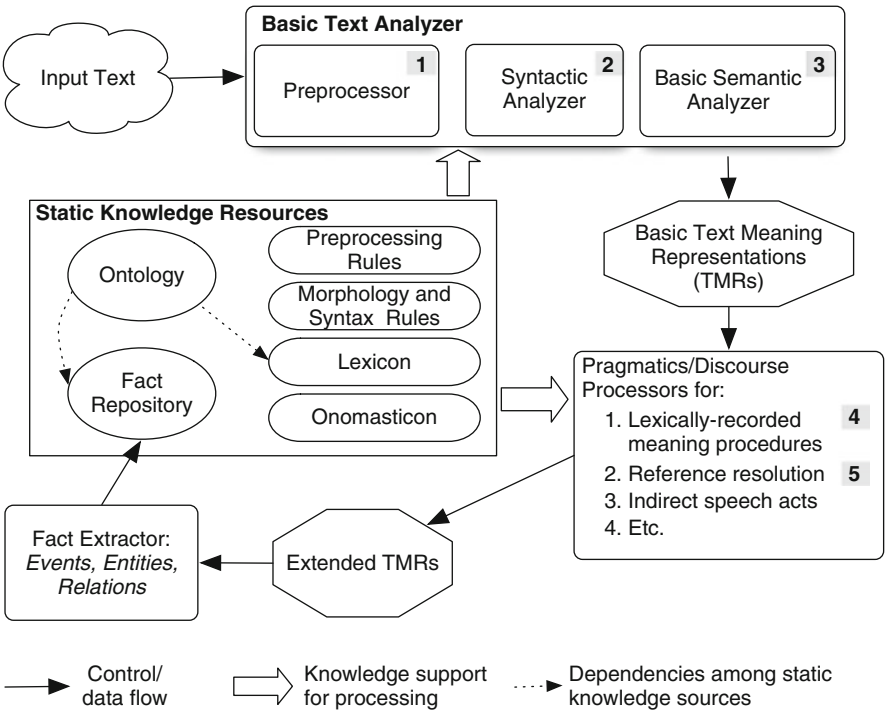

Fig. 9.1 The OntoSem text analyzer

systems; and Sect. 9.3 describes the environment (OntoAgent) and static knowledge resources that support the reported approach to reference resolution.

We then turn to the core of the chapter—our implemented algorithm for resolving reference. Treatment of reference is divided across five processing stages, indicated by numbers 1 through 5 in Fig. 9.1.[1]

- Stage 1. During preprocessing, the system detects proper names (one class of referring expressions) and begins the semantic analysis of their component parts (Sect. 9.4.1).
- Stage 2. From the syntactic parse, the system detects several types of structures that are potentially elliptical and adds reference-oriented metadata to the current state of analysis to support further downstream processing (Sect. 9.4.2).
- Stage 3. During "basic" semantic analysis the system:
  - Detects non-referring expressions (Sect. 9.4.3.1)
  - Detects (and in some cases resolves) certain types of ellipsis using clues from the dependency structures in the OntoSem lexicon (Sect. 9.4.3.2)

[1]This figure is further described in Sect. 9.3.

- Stage 4. The system runs lexically recorded procedural semantic routines to resolve idiosyncratic referring expressions such as *I, yourself* and *penultimate* (Sect. 9.4.4).
- Stage 5. Using a dedicated reference resolution module (Sect. 9.4.5)—which is one of several pragmatics/discourse analysis modules—the system:
  - Resolves referential definite descriptions (Sect. 9.4.5.1)
  - Resolves proper nouns (Sect. 9.4.5.2)
  - Resolves indefinite descriptions (Sect. 9.4.5.3)
  - Resolves bare NPs (Sect. 9.4.5.4)
  - Resolves third person pronouns (Sect. 9.4.5.5)
  - Resolves referential verbs (Sect. 9.4.5.6).

The chapter concludes (Sect. 9.5) with some further comments on the use of semantics in reference processing, and the role of knowledge resources in generating the semantic analyses that can improve reference resolution.

The broad coverage of phenomena treated here reflects one of the driving overall development methodologies of OntoSem, which prefers to treat all phenomena necessary for deriving high-quality text meaning representations from the outset, even if these treatments are understood to be incomplete. In other words, we prefer to implement incomplete algorithms rather than designate "wastebasket categories" that are not tackled at all by the system. Every one of the component algorithms presented below could—and, indeed, should—be expanded and improved, ideally with the participation of many people over many years. So, we are not claiming to have solved every aspect of automatic reference resolution, but we *are* suggesting that our approach and framework are sufficient to support indefinite continuation of the work, with no risk of reaching a ceiling of results past which a new approach or architecture will be needed.

9.2 Our View of Reference Resolution Versus Others

Automatic reference resolution in NLP has until now been approached in a highly selective manner. Consider the following contrasts between what most reference resolution systems to date have treated and what is ultimately needed to support sophisticated functioning by intelligent agents.

- Most reference resolution systems consider the establishment of textual coreference to be an end task, following the highly influential MUC reference resolution task definition [4] and all subsequent work that uses its corpus as a gold standard for machine learning. However, textual coreference is at best a clue to carrying out real reference resolution, which involves linking the meaning of referring expressions, within and across texts, to anchors in a fact repository or agent memory.

- Most reference resolution systems treat only *NPs* that are *overt in text* and have *NP sponsors* that are *single constituents*. However, verbs can also have referential function, both NPs and verbs can be elided, and both NPs and verbs can have sponsors that are entire spans of text or are disjunct constituents.[2]
- Most reference resolution systems treat entities that have been manually identified as being *referential* and having an *exact coreference* relationship with their sponsor. However, a central part of the reference resolution task is automatically identifying which entities are and are not referential, as is determining what kind of reference relationship an entity has with its sponsor—be it exact coreference, type coreference, a set-member relationship, etc.
- Most reference resolution systems assume that all earlier stages of processing have been carried out perfectly, which is an insupportable approach in any practical application (as discussed in [20, 34, 45]).
- Most reference resolution systems assume that sufficiently large manually annotated corpora are available to support machine learning; however, (a) manually annotated corpora are not available for the majority of reference phenomena in the majority of languages; (b) the practice of annotating corpora by merely indicating exact coreference links is insufficient; (c) the typical knowledge-lean machine learning approaches are insufficient for treating many kinds of reference phenomena (for a description of why, see [20]).
- Most annotation efforts, and systems built upon them, assume that there is always one specific and correct answer for reference resolution. However, referring expressions can be vague and language use in actual texts can be sloppy. It is noteworthy that the lack of a single, precise sponsor for referring expressions does not always prove problematic for people or for intelligent text processing agents, so we would suggest that the approach to evaluation fostered in competitions would be better supplanted by a more sophisticated understanding of the role of reference resolution in overall agent functioning.
- Most NLP on the whole assumes that deep semantic analysis is unattainable and therefore its results cannot be used by reference resolution modules. While it is true that no extant deep-semantic analysis system achieves perfect results, such systems do exist and provide useful results in many contexts. The challenge—apart from improving such systems so that they produce ever better results—is to implement automatic self-evaluation so that the system knows in which contexts it is confident enough about its semantic results to use them to inform reference resolution.

To summarize, if our goal is to create robust intelligent agents that can process reference with the same knowledge- and memory-oriented results as a human carrying out the same task, we must recast the reference resolution task as well as our approach to carrying it out in practical systems.

[2] Space does not permit a comprehensive overview of reference phenomena, but interested readers can find an accessible, example-rich treatment in [20].

9.3 The OntoAgent Environment and Its Resources

The treatment of reference presented here is implemented in the OntoAgent environment, which supports the modeling of human-like behavior in artificial intelligent agents that collaborate with people (for an overview of our group's work, see http://www.trulysmartagents.org/index.php). The agents in question have simulated bodies and simulated minds, with the latter providing cognitive capabilities that include interoception (the interpretation of one's bodily signals), learning, planning, decision-making, memory management and communication in natural language. Recent applications include Maryland Virtual Patient, in which a live clinician-in-training interacts with a virtual patient and a virtual mentor; and Clinician's Advisor, in which a live practicing physician receives advice from a virtual assistant Patent Pending (see, e.g., [22, 26–28, 39, 40]).

OntoAgent uses a primarily knowledge-based approach to agent modeling. This knowledge-based orientation extends to its text processing system, OntoSem, which is a practical implementation of the theory of Ontological Semantics [36]. In OntoAgent, all physiological, general cognitive and language processing capabilities of all intelligent agents rely on the same ontological substrate, the same organization of the fact repository (agent memory of assertions) and the same approach to knowledge representation [21].

The OntoAgent **ontology** is a formal model of the world that provides a metalanguage for describing meaning derived from any source, be it language, intelligent agent perception, intelligent agent reasoning or simulation. The metalanguage of description is unambiguous, permitting automatic reasoning about language and the world to be carried out without the interference of lexical and morphosyntactic ambiguities. A description of the ontology—as well as a rationale for its form and content—are available in [36]. Additional theoretical and practical issues are discussed in [32]. Here, we present highlights for orientation.

The ontology is organized as a multiple-inheritance hierarchical collection of frames headed by concepts that are named using language-independent labels. It currently contains approximately 9,000 concepts, most of which belong to the general domain. The number of concepts in the ontology is far fewer than the number of words or phrases in any language for several reasons: (1) Synonyms and hyponyms are mapped to the same ontological concept, with semantic nuances recorded in the corresponding lexical entries. (2) Many lexical items are described using a combination of concepts. (3) The same ontological property can be used with different values to represent different meanings on a given scale: e.g., *gorgeous* and *ugly* are described using the values 1 and .1, respectively, on the 0–1 scale AESTHETIC-ATTRIBUTE. (4) Concepts are intended to be cross-linguistically and cross-culturally relevant, so we tend not to introduce concepts for notions like *recall* in the sense of asking buyers to return a purchased good because it is highly unlikely that all languages/cultures use this concept. Instead, we describe the meaning of such words compositionally in the lexicons of those languages that do use it. For further discussion of the lexicon/ontology split, see [25].

Concepts divide up into events, objects and properties. Properties are primitives, which means that their meaning is understood to be grounded in the real world without the need for further ontological decomposition. The expressive power of the ontology is enhanced by multivalued fillers for properties, implemented using facets. Facets permit the ontology to include information such as "the most typical colors of a car are white, black, silver and gray; other normal, but less common, colors are red, blue, brown and yellow; rare colors are gold and purple." The inventory of facets includes: *default*, which represents the most restricted, highly typical subset of fillers; *sem*, which represents typical selectional restrictions; *relaxable-to*, which represents what is, in principle, possible but is not typical; and *value*, which is used primarily in the fact repository to indicate an actual value. Select properties from the ontological frame for the event DRUG-DEALING illustrate the use of facets.

DRUG-DEALING		
IS-A	value	CRIMINAL-ACTIVITY
AGENT	default	CRIMINAL, DRUG-CARTEL
	sem	HUMAN
	relaxable-to	SOCIAL-OBJECT
THEME	default	ILLEGAL-DRUG
INSTRUMENT	sem	MONEY
HAS-EVENT-AS-PART	sem	BUY, SELL
LOCATION	default	CITY
	sem	PLACE
	relaxable-to	PHYSICAL-OBJECT
...		

Objects and events are defined for an average of 16 properties each, many of whose fillers are inherited rather than locally specified. In short, the *meaning* of an object or event is the set of its property-facet-value triples. Unlike most ontologies, the OntoAgent ontology includes complex events (i.e., scripts), defined as events having obligatory or optional subevents, which in turn can have their own subevents, and so on (cf. [44]). Scripts support both simulation and reasoning about language and the world.

Since the OntoAgent ontology is language independent, its link to any natural language must be mediated by a **lexicon** that includes a proper-name component, called an **onomasticon**. Semantically, each lexical sense specifies which concept, concepts, property or properties of concepts defined in the ontology must be instantiated in the text-meaning representation to account for the meaning of a given lexical unit of input. For example, the English lexicon indicates that the one sense of *dog* maps to the concept DOG (a type of CANINE); another sense maps to HUMAN, further specified to indicate a negative evaluative modality (e.g., a woman can call her cheating ex-boyfriend a dog); and yet another sense maps to the event PURSUE. Senses for argument-taking words and modifiers are presented along with their typical syntactic configurations, such that a word in the configuration is

described syntactically and semantically. Take, for example, the adverbial sense of *overboard*, shown below. It says that, syntactically, this adverb (indicated by the variable called $var0) modifies a verb (indicated by $var1) and, semantically, that the verb it modifies must be a MOTION-EVENT. It further says that the SOURCE of the given MOTION-EVENT is a SURFACE-WATER-VEHICLE and its DESTINATION is a BODY-OF-WATER. We use an abbreviated lexical formalism for all cited examples.

```
(overboard-adv1
    (def "indicates that the source of motion is a surface water vehicle and
          the destination is a body of water")
    (ex "They threw the rotten food overboard.")
    (syn-struc
          ((root $var1) (cat v) (mods ((root $var0) (cat adv)))))
    (sem-struc
          ($var1 (sem MOTION-EVENT)
                (SOURCE SURFACE-WATER-VEHICLE)
                (DESTINATION BODY-OF-WATER)
```

This example highlights several aspects of the OntoAgent lexicon. First, it supports the combined syntactic and semantic analysis of texts. Second, the descriptions in its sem-strucs are, in terms of format and primitives used, the same as one would find in the ontology. And third, the sem-strucs—and often the associated syn-strucs—from the lexicon for one language can very often be ported into the lexicon of another language with little or no modifications, which greatly enhances the multi-lingual applicability of the OntoAgent suite of resources.[3] For discussion of the cross-lingual use of OntoAgent lexicons, see [25].

Whereas the ontology contains ontological concepts, like CITY and WAR, an agent's **fact repository** contains remembered instances of those concepts, like London (say, CITY-84) and World War II (say, WAR-4). For example, at a given time during the life of an agent, its fact repository might contain the following information about London; of course, vastly more could be added from processing encyclopedic texts about the city, reports about current events that have happened there over the centuries, etc. In fact, a "walking encyclopedia" intelligent agent could have a fact repository in which every fact publicly known about London would be linked to this FR anchor called CITY-84, with time stamps and attributions for all of the information, since property values can change over time.

[3]Procedural semantic routines, which are recorded, when needed, in a "meaning procedures" zone not shown in the example above, are also largely portable across languages. For more on meaning procedures, see Sect. 9.4.4 and [24].

```
CITY-84                                  ; the 84th instance of CITY in this FR
    LOCATION      value NATION-2         ; Great Britain
    CAPITAL-OF    value NATION-2
    LOCATION-OF   value WEDDING-16       ; Prince William/Kate Middleton wedding
                  value MEETING-76
                  value LECTURE-12
                  (and possibly hundreds or thousands of other events, each of
                  which will be described with all of the relevant properties
                  and values in its own fact repository frame)
    LOCATION-OF   value TOWER-1          ; Tower of London
                  ...
```

Each object and event that is used as a property filler in one frame also heads its own frame in which all of its known property values are recorded: e.g., the frame for WEDDING-16 will include all the facts this agent knows about the royal wedding of Prince William and Kate Middleton.

9.3.1 Comparing OntoAgent Static Knowledge Resources with Others

The knowledge base that is probably closest in spirit to the OntoSem ontology is that used in Knowledge Machine (KM) [5]. Both KM and OntoAgent use frame-based knowledge representation languages, but whereas KM focuses on logic-based reasoning, OntoAgent also uses the game-theoretic notion of utility. Some points of similarity among the ontologies in the two environments are: the use of a large inventory of properties to describe objects and events; the possibility of placing complex fillers in slots, nesting such descriptions to any depth, and tracking coreferences within such nested structures (permitting the construction of scripts and prototypes); the support of knowledge expressed in conditional statements; a three-tiered distinction between concepts (called 'classes' in KM), instances, and a hybrid entity that lies somewhere in between (in OntoAgent this hybrid is called an ontological instance; in KM it is called a proto-instance). This thumbnail comparison is not intended to imply complete overlap of two knowledge bases using different formalisms; rather, it suggests that if one is to pursue sophisticated machine reasoning, the content of the knowledge bases supporting that reasoning turn out to be strikingly similar across research paradigms.

Perhaps the best way to describe the amount of knowledge stored in the Onto-Agent ontology is to compare the latter with two well-known resources: CYC [41] and SUMO [35]. CYC is massive, containing "nearly five hundred thousand terms, including about fifteen thousand types of relations, and about five million facts (assertions) relating these terms" [7]. The knowledge in CYC is generally described as commonsense knowledge, which is formulated in the CycL formal language and recorded not in frames but as a "sea of assertion" (ibid). It is expanded using a

combination of automatic and manual methods. SUMO, by contrast, is only an upper ontology and, as such, is far smaller—containing 1,000 terms, 4,000 axioms stated in first order logic and 750 rules. As an upper ontology, SUMO covers only more abstract concepts; however, a number of other ontologies have been connected to it, including various domain ontologies and MILO (Mid-Level Ontology), which links SUMO with the domain ontologies. Let us make the comparisons between the OntoAgent ontology, CYC and SUMO explicit. *Number of concepts*: CYC—500,000, OntoSem—9,000, SUMO—1,000. *Number of statements/rules/axioms*: CYC–five million facts, OntoSem 100,000 RDF triples (conversion into RDF triples, as described in [11], is a useful counting mechanism even though OntoAgent does not use RDF), SUMO–4,000 axioms and 750 rules.

OntoAgent static knowledge resources are compiled primarily manually. There are three reasons for this: (1) our applications of interest require high-quality knowledge; (2) our experience correlates with that of [6], who report that there has been only modest success in using existing ontologies to build new ones; and (3) we are centrally interested in application-oriented research and development as opposed to the development of machine learning methods to carry out such tasks as learning ontologies and word nets from corpora, automatically merging ontologies and word nets, and generating word nets in one language by bootstrapping from another language (see, e.g., [1,10,42]). This is not to say that we do not use external resources in compiling our own—we do, but to inform rather than displace the manual acquisition process. For example, the lexicon acquisition process regularly involves checking WordNet [33] and dictionary.com for synonyms and hyponyms, and the development of the medical aspect of the ontology has benefitted from the University of Washington's Foundational Model of Anatomy [43]. We have found that pruning or cleaning a noisy resource is no less work than building a resource from scratch, and our acquisition methodology reflects this experience.

Each agent in an agent network can be endowed with its own ontology, lexicon and fact repository, all of which can be augmented during its simulated life through learning and experience. In addition, in order to support specific functionalities in agent systems, such as tutoring and advice-giving, agents can be configured to dynamically create and update models of other agents' knowledge and beliefs [23,37]. In order to moderate between its own knowledge bases and the understood knowledge bases of others, an agent relies on capabilities that include, non-exhaustively, the interpretation of ontological, syntactic and lexical paraphrase, as described in [29,30].

9.3.2 The OntoSem Text Analyzer

Figure 9.1, presented in Sect. 9.1, shows the architecture of the OntoSem text analyzer, which is a component of the OntoAgent environment. The OntoSem text analyzer takes as input natural language text and generates disambiguated, ontologically grounded structures—which we call Text Meaning Representations,

or TMRs—that are well suited to machine reasoning. Basic TMRs include the results of lexical disambiguation and the establishment of the semantic dependency structure, whereas extended TMRs include the results of reference resolution, the interpretation of indirect speech acts, and other discourse-level aspects of language processing.

As an example of OntoSem processing, consider the TMR for the input *Charlie watched the baseball game*.

VOLUNTARY-VISUAL-EVENT-1	
AGENT	HUMAN-1
THEME	BASEBALL-GAME-1
TIME	(<find-anchor-time) ; indicates past tense
textstring	"watched"
from-sense	watch-v1
HUMAN-1	
AGENT-OF	VOLUNTARY-VISUAL-EVENT-1
HAS-PERSONAL-NAME	"Charlie"
textstring	"Charlie"
from-sense	personal-name
BASEBALL-GAME-1	
THEME-OF	VOLUNTARY-VISUAL-EVENT-1
textstring	"baseball_game"
from-sense	baseball_game-n1

TMRs like this one can provide semantic features for many aspects of agent reasoning, including reference resolution.

This concludes our brief overview of OntoSem text processing. We now turn to the reference resolution algorithm that is distributed throughout five stages of processing.

9.4 The Reference Resolution Algorithm

This section describes how reference processing is divided across five stages of OntoSem text analysis, as summarized in Sect. 9.1 and illustrated using numbers 1 through 5 in Fig. 9.1.

9.4.1 Stage 1: Proper Name Analysis During Preprocessing

Preprocessing in OntoSem involves such processes as tokenization, HTML stripping, morphological analysis, the identification of dates, times and measures, and—of particular interest for reference resolution—the identification and analysis of proper names, which is a reference-resolution task that has often been pursued in

isolation under the rubric "the named entity recognition (NER) task". The OntoSem preprocessor [32] represents a conglomeration of resources and engines, including the preprocessor provided with the Stanford parser [14]. The version of the Stanford preprocessor we integrated into our system did not provide a semantic analysis of the components.[4] For example, for the input *Army Capt. Patrick Horan* returns (NP (NNP Army) (NNP Capt.) (NNP Patrick) (NNP Horan)). In order to support the depth of semantic analysis sought in OntoSem, we have developed a post-Stanford engine that uses our ontologically-mapped onomasticon and an inventory of patterns to further semantically analyze named entities. The output of this engine for our example is: (HUMAN-1 (HAS-TITLE *Army Capt.*) (HAS-PERSONAL-NAME *Patrick*) (HAS-SURNAME *Horan*)). If any semantic ambiguities are detected at this stage (e.g., *Washington* as a person, state, city, etc.), they are retained until further semantic analysis is brought to bear.

9.4.2 Stage 2: Detection of Potentially Missing Elements in the Syntactic Parse

For syntactic analysis, OntoSem currently uses the Stanford parser [8, 14] out of the box. However, before utilizing its output to support semantic analysis, we modify that output in certain ways. Some of those modifications specifically support reference resolution, such as the detection of potential instances of verbal gapping (1), unexpressed subjects in the latter conjuncts of VP-coordinate structures (2) and unexpressed objects in the former conjuncts of clausal coordinate structures (3).[5]

(1) Recently, wildlife researchers discovered a lungless frog and marine scientists ___, new soft corals.

(2) Kerry tripped and ___ fell.

(3) Kerry washed ___ and James dried the dishes.

We will use gapping as an illustration of OntoSem's post-syntactic transformations. Gapping is an elliptical process in which the verb in the latter conjunct of a conjoined structure is elided under type-coreference (not instance-coreference) with the verb in the first conjunct. The major constraints on gapping (see [17, 18] for others) are: (1) the antecedent must be overt in the preceding clause; (2) the gap must be surrounded by overt categories for which there are semantically "parallel" counterparts in the antecedent clause; (3) gapping can be used in coordinate and

[4] Stanford's "CoreNLP", which includes more extensive proper name analysis, was not available until recently.

[5] We refer to these missing categories as "unexpressed" rather than the more theoretically-charged "elided" in order concentrate on the fact that these configurations pose difficulties for machine processing, no matter how they are treated within one or another theoretical paradigm.

comparative structures but not in subordinate ones; (4) the overt constituents in the gapped clause must be major constituents; and (5) gapping is recursive, meaning it can be used in multiple clauses in a row.

The Stanford parser typically treats gapping structures as conjoined nominals—essentially, appositives. This is, in fact, not inappropriate because the same syntactic configuration populated by different lexical items can be, in one case, an appositive (4), in another case, a gapping structure (5), and in yet another case, ambiguous without further context (6) (*Did John invite Jean, who is Sue's sister, or did Jean invite Sue's sister?*). In short, semantic and discourse analysis are required to weigh in on the final interpretation.

(4) Jake made pizza and Mary's favorite dessert, cupcakes.

(5) I ate a burger and my dog, a bone.

(6) John invited Mary and Jean, Sue's sister.

We have developed a gapping-detection function that (a) detects syntactic configurations that might indicate gapping, (b) recovers the missing verbal element by copying the verbal string (which has not yet been semantically analyzed) from the previous conjunct, (c) adds metadata to the copied string that explicitly blocks instance-coreference, thus facilitating the later reference resolution task and (d) reinterprets the incorrect NP coordinate structure as a clausal coordinate structure with a gap. This candidate parse is passed on to the semantic analyzer along with the original appositive analysis so that semantics can be used the final arbiter between interpretations.

Similar transformations are used to reconstruct unexpressed subjects and objects in the configurations illustrated by (2) and (3) above. One noteworthy difference between the treatment of gapping and these other two "missing-element" phenomena is that for the latter, the metadata of the copied string indicates that this represents instance (not type) coreference—i.e., the overt and unexpressed categories are coreferential.

To summarize, OntoSem's post-syntactic transformations *detect* a potentially missing category, *fill* it with a copy of the appropriate string from the text, *add* reference-oriented metadata to the copy of the string that indicates whether it should be interpreted as type-coreference or instance-coreference, and suggest potentially needed *corrections* to the original parse.

9.4.3 Stage 3: Reference Processing During Basic Semantic Analysis

The main goals of *basic* semantic analysis in OntoSem are lexical disambiguation and the establishment of semantic dependencies. However, this stage also includes certain reference-oriented processes, such as the detection and analysis of

non-referring expressions, and the detection—and, in some cases, resolution—of certain types of ellipsis.[6] We discuss these in turn.

9.4.3.1 Lexically Supported Detection and Analysis of Non-referring Expressions

Most nouns and verbs have referential function and therefore should be subject to reference resolution procedures. However, there are at least four categories of non-referential expressions that must be detected automatically.

Pleonastic *it*. The identification of pleonastic (i.e., non-referential) *it*, as found in sentences like (7), has traditionally been undertaken either as a standalone task (e.g., [3, 16]) or in the context of resolving personal pronouns (e.g., [15]).

(7) It is clear that the game will be cancelled due to weather.

Methods for detecting pleonastic *it* include machine learning and pattern matching. Working within our group, Johnson [13] improved extant pattern-based methods in three ways: by expanding upon the word lists used in previously proposed patterns; by permitting certain types of intervening material in patterns; and by adding new patterns. He also developed a method of automatically evaluating the system's confidence in the pleonastic interpretation of each instance of *it*. Johnson's pleonastic *it* detection system relies on both the lexicon and the ontology for its heuristics.

Lexically null-semmed elements. One aspect of lexical description is the indication of which elements of a dependency structure should not be compositionally analyzed. For example, in the idiom *kick the bucket* (which is recorded under the head word *kick*), the NP *the bucket* is "null-semmed"—i.e., receives a null semantic interpretation—since its meaning is folded into the overall meaning of the idiom, DIE. Any element that has a null-sem interpretation in its lexically recorded dependency structure is excluded from all subsequent aspects of reference processing.

Predicate nominals. Predicate nominals regularly have attributive rather than referential function. For example, in *Peter is a doctor*, the NP *a doctor* should not create a new instance of an object of the type PHYSICIAN in the text meaning representation. Instead, the text meaning representation of the input should be: (HUMAN-1923 (HAS-PERSONAL-NAME *Peter*) (HAS-SOCIAL-ROLE PHYSICIAN)). We prepare the system to arrive at this interpretation with a special lexical sense of *be* (and certain other copular words) that expects the subject to be of the type

[6] For reasons of space, we omit another aspect of reference processing that is carried out at this stage: the detection of configurations in which the lexical disambiguation of a verbal head should be postponed until it can be informed by the reference resolution of one of its arguments. Interested readers can find relevant discussion in [31].

HUMAN and the predicate nominal to be of the type SOCIAL-ROLE. Only structures that meet these constraints will be interpreted using this lexical sense. Of course, this covers just one semantic use of predicate nominals; for others, we create similar lexical senses of *be* with associated constraints. This basic semantic analysis of predicate nominals excludes them from further reference processing, since the filler of SOCIAL-ROLE is a type, not a token.

Appositives. Appositives can be thought of as reduced predicate nominal constructions. For example, the sentence *Peter, a physician who lives next door, cuts his grass too short* could be rephrased as *Peter is a physician who lives next door and he cuts his grass too short*. Appositives are treated in the lexicon and in TMR very similarly to predicate nominal constructions, being excluded from subsequent reference processing.

9.4.3.2 Lexically Supported Detection (and Resolution) of Ellipsis

Several types of ellipsis can be detected from the combination of the syntactic parse and the lexical items in question. These include verb phrase (VP) ellipsis, verbal complement ellipsis, sluicing, and the semantic ellipsis of head verbs in lexically idiosyncratic configurations.

VP ellipsis and verbal complement ellipsis. VP ellipsis is the ellipsis of the whole verb phrase, licensed by an overt auxiliary or aspectual verb, with or without infinitival to: *John went because he wanted to [e]; John doesn't want to go but Mary does [e]; John started [e] early*. Verbal complement ellipsis is similar, but the licensing verb has full semantics: *I know <heard, tried> [e]*. Lexical specifications can help to detect VP ellipsis and verbal complement ellipsis, but the resolution of the elided categories must be carried out after basic semantic analysis, using the reasoning methods of the dedicated reference resolution engine. Here we concentrate on how relevant lexical senses are recorded to support ellipsis *detection*.

All lexical senses that anticipate VP ellipsis and verbal complement ellipsis include a description of the meaning of the overt elements as well as a call to a procedural semantic routine—recorded in the optional meaning-procedure zone—that will resolve the elided ones (for more on meaning procedures in OntoSem, see [24]). For example, one sense of *want* covers inputs like *John wants to*, in which the meaning of *want* and its relationship with John are clear (*want* indicates volitive modality that is attributed to *John*), but the event in question—which is formally the scope of the volitive modality—must be reconstructed using a reference resolution function. Like most meaning procedures, the one recorded in this lexical sense will be carried out after basic semantic analysis. Referring back to Fig. 9.1, this occurs after the basic TMR has been generated, during the discourse processing stage. To reiterate, a *call* to the meaning procedure is recorded in the basic TMR, whereas the results of its resolution are reflected in the extended TMR, which is then used to populate the agent's fact repository.

The resolution of the meaning of elided VPs and complements can be quite complex due to the wide variety of reconstructions possible, as described in [12]. To take just two examples:

(8) Dr. Smith thinks John should **have an endoscopy** but John doesn't want to [e].

(9) Dr. Smith **wants to run every morning** and Dr. Jones does [e] too.

Both of these examples show type-coreference (not instance-coreference) between the elided category and its sponsor, but in (8) that event must be stripped of its modality ([e] = *have an endoscopy*, not *should have an endoscopy*) whereas in (9) the modality is part of the reconstruction ([e] = *wants to run every morning*, not *run every morning*). The reasoning required to arrive at the correct interpretation in each case is non-trivial.

Sluicing. Sluicing is the ellipsis of embedded questions, which is licensed by the question word, as in (10).

(10) **Fido's bone is** in the yard, but we don't know exactly where [e].

Like VP ellipsis and verbal complement ellipsis, sluicing structures are detected using lexically recorded patterns. So, the OntoSem lexicon contains a sense of *where* that expects the syntactic configuration [subject + verb + where]; this sense includes a call to a procedural semantic routine that will resolve the meaning of *where* using the semantic interpretation—i.e., TMR—of the preceding context.

Event ellipsis indicated by aspectual + OBJECT. When aspect indicators like *start* and *finish* are used with an overt complement that is ontologically an OBJECT, the implied event is elided. For example, in (11) the implied event is BUILD and in (12) it is BROADCAST or PUBLISH.

(11) He conscripted 700,000 slaves to finish the Great Wall.

(12) Manufacturers Hanover this week started a new series of ads that push "Power Savings."

The OntoSem lexicon has dedicated senses of aspectuals that expect an OBJECT complement. These senses include a meaning procedure that attempts to dynamically recover the most specific possible meaning of the elided EVENT based on the meaning of the overt arguments. For example, when processing (11), the engine will search the ontology for an EVENT for which the default AGENT is SLAVE and the default THEME is WALL. If more than one match is found, all options are retained for possible later disambiguation based on further context. If no matches are found searching on the fillers of the *default* facet, fillers of the *sem* facet are analyzed, with the likely outcome that more than one candidate match will be returned. In this case, as earlier, all options will be retained for possible later disambiguation, which can exploit information from the context and/or the fact repository, as for giving preference to the meaning WRITE in a context like *John Steinbeck started a new book*, if the FR contained the information that John Steinbeck was a writer.

Idiosyncratic event ellipsis. In some cases, an event is elided in a particular lexically-constrained construction that permits both detection and reconstruction of the ellipsis—i.e., no meaning procedures are needed. For example, when one invites someone to some place (*She invited me to Paris*), the elided event is a MOTION-EVENT; similarly, when one forgets something—and that *something* is a PHYSICAL-OBJECT (*I forgot my phone*)—the elided event is TAKE. The semantic descriptions of the associated senses of *invite* and *take* include the event meanings so that the *basic* TMRs for such inputs contain the full meaning representation with no need for additional procedural semantic processing.

To summarize, lexicon entries can help the analyzer to detect certain types of ellipsis. In some cases, the configuration itself also suggests the semantic resolution of the elided category, but in most cases, resolution requires further reference processing procedures carried out during pragmatic/discourse analysis.

9.4.4 Stage 4: Running Lexically Recorded Meaning Procedures

Many referring expressions are conveniently resolved using individually crafted functions that can be recorded in the meaning procedures zone of the given lexical sense. Words like this include: the first and second person pronouns, whose resolution often involves seeking metadata in a written work, dialog application, email thread, etc.; the reflexive pronouns, which must corefer with their subjects; phrases like *abovementioned, penultimate*; and many others. We develop functions for each of these individually and record calls to them in the OntoSem lexicon. Like all meaning procedures, the functions are run after the basic TMR has been created, and their results are stored in the extended TMR.

9.4.5 Stage 5: Dedicated Reference Resolution Module

The dedicated reference module, which is one of OntoSem's pragmatics/discourse processors, is called when basic TMRs have already been constructed and most lexical disambiguation and dependency-oriented decisions have been made. TMRs are a source of semantic heuristics that can be leveraged for reference resolution. Since we are focusing here on the role of static knowledge resources in reference resolution, we must emphasize that since the generation of TMRs depends centrally on knowledge recorded in the lexicon and ontology, *any time that an element of TMR is used in a reference resolution function, the lexicon and ontology are directly involved*.

A precondition for resolving referring expressions (REs) is detecting which elements are REs to begin with. In OntoSem, this can be carried out in two ways:

starting from the TMR and chaining back to text elements, or starting from text elements and chaining forward to the TMR. More specifically, if the system starts with TMRs, it can interpret all OBJECT and EVENT instances in TMRs as referring expressions. (All text elements that are not REs will not have given rise to an OBJECT or EVENT instance in TMR.) Since each TMR frame contains metadata that links the concept instance to the text string(s) that gave rise to it, a list of text-level referring expressions can readily be generated. To take an example from the previously presented TMR, since VOLUNTARY-VISUAL-EVENT-1 is an instantiated event, the textstring that gave rise to it, *watched*, must be a referring expression.

VOLUNTARY-VISUAL-EVENT-1

AGENT	HUMAN-1
THEME	BASEBALL-GAME-1
time	(¡ find-anchor-time) ; indicates past tense
textstring	"watched"
from-sense	watch-v1

The second way of compiling the inventory of REs is starting from text strings. The system can use the syntactic parse to detect NPs and main verbs, remove from that list non-referential ones, as detected using methods described earlier, and add to the list any elided categories that were detected in earlier processing. We have been experimenting with both methodologies.

Interestingly enough, the list of entities that can serve as potential sponsors for REs does not precisely match the list of REs. The list of candidate sponsors includes:

- All REs.
- All TMR frames headed by modality, aspect and properties, as these have a special kind of referential function. For example, in (13) the TMR frame headed by aspect—reflecting the meaning of *stopped*—must be available as a sponsor for the RE *its cessation*.

  (13) The fighting stopped early last week. Its **cessation** was a welcomed relief.

- Semantic sets that do not comprise a syntactic constituent—which, for the sake of efficiency, are created only on an as-needed basis. These can be required to resolve plural REs, as in (14):

  (14) Your accordion is much bigger than my horn but **they** are almost equally loud.

- Text spans—or, semantically speaking, propositions—which can serve as sponsors for referring expressions like pronominal *it/that/this*, abstract objects (e.g., *this speech, the preceding paragraph*), and so on. Text spans, like sets, are composed on an as-needed basis.

The list of candidate sponsors for any particular RE is the subset of the elements described above that fall within the so-called *window of coreference*,

which is typically understood to be the preceding three or four sentences of text (an oversimplification, but one that works for most contexts).

We now move on to the algorithm for resolving five classes of REs that remain unresolved at this point in the text analysis process: definite descriptions, indefinite descriptions, bare NPs, third person pronouns, and referential verbs.

9.4.5.1 Resolve Referential Definite Descriptions: NP-Def

Definite descriptions, which are NPs with the article *the* in English, may or may not have a textual sponsor. As has been noted in the literature, the percentage of definite NPs that are part of a coreference chain has been counted at 50 [46], 37 [2], and 61 % [9] for different corpora. As such, a large part of the treatment of NP-Def involves resolving reference directly to the fact repository.

Our approach to treating NP-Def is comprised of nine ordered steps. Note that seeking a textual coreferent is not the first step.

Step 1. If the given NP-Def is listed in the onomasticon, and therefore already has a FR anchor (i.e., it is known to the agent), resolve reference by linking directly to that anchor: e.g., *the United Nations* will be remembered as an instance of INTERNATIONAL-ORGANIZATION in the FR of most agents.

Step 2. Else if the given NP-Def is not listed in the onomasticon but contains a proper noun part, then: (a) Create a new FR anchor for the interpretation of the whole NP-Def and (b) treat the proper noun part in the same way as all proper nouns are treated: if it already has an FR anchor, then link to that anchor; otherwise, create a new FR anchor for it and link to that new anchor. For example, if an agent knows about *the University of Maryland Medical Center*, but has no specific knowledge or memories of *the University of Maryland Medical Center emergency room*, then if the latter string were encountered, the following memory modifications would take place:

Original FR

```
MEDICAL-CENTER-FR7
    HAS-NAME        "University of Maryland Medical Center"
    LOCATION        CITY-413                ; Baltimore
    RELATION        UNIVERSITY-122          ; University of Maryland
```

FR addition

```
EMERGENCY-ROOM-FR4                          ; a new FR anchor
    RELATION        MEDICAL-CENTER-FR7      ; linking to the known
                                              proper-noun part
```

Outstanding issues include the potential ambiguity of the proper-noun part and the necessity of analyzing unknown proper nouns, both semantically and morphologically (e.g., "Sorbian" should link to the FR entity "Sorbia", if it exists, or create a new FR entity for "Sorbia", not "Sorbian").

Step 3. Else seek a confident (high-scoring) textual sponsor for NP-Def. We currently use a knowledge-based approach for this process that involves leveraging linguistic configurations that we have found to have predictive power with respect to coreference relations (for details see [19,32]). These configurations are described using feature values derived from any aspect of OntoSem processing. To take just one example: the so-called "broad" referring expressions like pronominal *it, that* and *this* can be difficult to resolve automatically because they can have so many different types of sponsors: an object, an event, a proposition, many propositions, or even something vague that cannot be pointed to directly in the text. However, in certain configurations, one can make a rather confident guess as to the sponsor for a broad referring expression using combinations of automatically detectable feature values. For example, in (15), *it* can be predicted to corefer with the proposition *moods don't last* using a set of feature values that can be informally described as follows: (a) the RE is a broad referring expression; (b) it the subject of its sentence; (c) its sentence is the first sentence of a new speaker turn; (d) the previous speaker's turn is comprised of a single proposition; (e) there is no neuter singular NP in the previous speaker's turn that could serve as sponsor for the RE.

(15) [Lord Illingworth] She is more than a mystery—she is a mood. [Mrs. Allonby] Moods don't last. [Lord Illingworth] It is their chief charm (Oscar Wilde).

This particular feature-value combination has been assigned a medium-high score since it is fairly predictive but one can find counterexamples—e.g., the preceding remark could be an aside. However, in this context this feature-value combination will be the strongest hypothesis for the correct reference resolution since no other higher-scoring feature-value combinations matches this input.

Once candidate textual sponsors have been evaluated, processing can continue in several ways. Under one control structure, if the highest-scoring candidate sponsor scores above a threshold, it is selected and processing stops; this is a good, least-effort solution if one has confidence in the scoring system and if processing time/effort is important. Alternatively, all of the algorithms below can be run and their output (if any) can be scored; then the scores of all candidate solutions can be compared at the end, with the highest scorer winning. Under yet a third control structure, all results scoring above a given threshold can be retained and the system can determine whether or not this particular reference decision is important for the operation of the given intelligent agent—it might not be. If it is, then the way the residual ambiguity is resolved depends upon in the application: if an end user is in the loop, the system can query him; if a developer is in the loop, he can be asked to provide context-specific disambiguation rules; and so on.

Step 4. Determine if the entity is in the "universally known" list, which contains entities like the *sun, the atmosphere, the moon* etc.[7] For now, we assume that all agents have the same "universally known" list, and that there are corresponding FR anchors to which a direct link can be made.
One obvious outstanding issue with "universally known" elements is that strings used to indicate them can have also have non-default interpretations, as when *the sun* refers to a model of the sun in a child's depiction of the solar system. Carrying out Step 3 before this step attempts to take care of at least some such cases. Another outstanding issue involves incorporating agent learning into the evaluation of definite descriptions. For example, say a person encounters sentence (16), never having heard of a lower esophageal sphincter.

(16) When a person swallows, the lower esophageal sphincter must relax.

Any average person will understand that the lower esophageal sphincter must be some body part involved in swallowing, and he will certainly not search the preceding context for a textual sponsor to justify the use of the article *the* in this NP. The same behavior must be modeled in intelligent agents—a process we have begun to develop in the overall architecture of our cognitive agents [38].

Step 5. If the RE is plural, create and evaluate composed candidates. Our current algorithm, simplified for presentation, is as follows. If two or more candidates in the list of candidate sponsors map to the same ontological concept as the RE, create a set from them and consider that set a high-scoring sponsor (17). (Concept mappings are shown in square brackets.) Else, if two or more candidates map to the same concept that is a descendant of the RE concept, create a set from them and use that set as the sponsor (18). Else, if two or more candidates share an ancestor that is at least three levels from the root of the ontology, and if that common root is semantically compatible with the RE, create a set from the elements and use that set as the sponsor (19). (Note that although the lexical description of *ruckus* is headed by the concept PHYSICAL-EVENT, it also includes additional property-based descriptors, such as HAS-EVENT-AS-PART MAKE-NOISE, which do not affect the reference-oriented evaluation function.)

(17) Their goalie [GOALIE] had his eye on our goalie [GOALIE] the whole game. After the game while the other players were shaking hands, **the goalies [GOALIE]** fought.

(18) Their dog [DOG] is nice and our cat [DOMESTIC-CAT] is too. After dinner, **the animals [ANIMAL]** played.

(19) Mary ran [RUN] after Gina and then the cat hissed [HISS] at Mary. The **ruckus [PHYSICAL-EVENT]** scared the dog.

[7] We currently have only a small "universally known" list, not of the magnitude of the lexicon or ontology. Further developing this resource is on the agenda.

Step 6. Determine if there is an object-meronymic relationship between the RE and any of the candidate sponsors, starting with the closest. The query is, "Does the ontological mapping of the RE (in (20), WINDOW) fill the HAS-OBJECT-AS-PART slot of the ontological mapping of any of the candidate sponsors (in (20), ROOM)?"

(20) I walked in the room and **the window** was open.

If object meronymy is detected, a new anchor for the RE is created in the FR and that anchor is linked to the anchor for sponsoring candidate using the relation HAS-OBJECT-AS-PART and its inverse PART-OF-OBJECT. For example, if the window in (20) is remembered in the FR as WINDOW-FR-18 and the room is remembered as ROOM-FR-712, then the following assertions will be added to the FR:

WINDOW-FR-18 PART-OF-OBJECT ROOM-FR-712
ROOM-FR-712 HAS-OBJECT-AS-PART WINDOW-FR-18

Step 7. Determine if there is a "member of candidate set" relation between the RE and any of the candidates, working backwards. Example (21) shows such relation between the boldface and underlined REs.

(21) A couple walked into the hospital and **the man** was carrying a cane.

Set-member relations can be detected readily at the level of TMR as long as the set was described appropriately in the lexicon. In the OntoSem lexicon, one meaning of *couple* is lexically described as: (set (MEMBER-TYPE HUMAN) (CARDINALITY 2)). As such, the TMR virtually contains two available humans that can be used individually as sponsors for reference resolution. We can formally recast the set notation describing *couple* into the equivalent (set (HAS-ELEMENTS HUMAN-1, HUMAN-2)) and create a coreference link between *the man* and one of these humans.

Step 8. Seek a bridging analysis by interpreting event scripts. The use of a definite description can be licensed by reference to an event that suggests the existence of its participants. For example, the mention of a flight suggests the existence of all of the human and non-human participants in the flight: the pilot, the flight crew, the plane, the passengers, etc. As such, one can use a definite description to refer to *the pilot* without having explicitly introduced him before, as in (22):

(22) When a flight is bumpy, the pilot typically explains why.

The FR augmentation that takes place when a bridging analysis is detected is comprised of two parts: creating a new anchor for the definite description, and linking it to the existing anchor for the sponsoring event using the appropriate property. In this case, the new instance of PILOT will fill the AGENT slot of the instance of FLY-PLANE that sponsored the reference resolution. Event scripts, which are being added to the OntoSem ontology as time permits, allow for this kind of reference-oriented reasoning to take place.

Step 9. If none of the above functions leads to a confident, high-scoring resolution of the NP-Def, then the system can either accept the best of the bad textual coreference analyses found in Step 3 or create a new FR anchor with no additional reference links. We have not yet worked on optimizing this decision process.

9.4.5.2 Resolve Proper Nouns

Proper nouns never strictly *require* a textual sponsor since they can always be directly linked to their FR anchor. However, if they have a textual sponsor it can be useful to detect it. For example, if *John* was previously referred to in the text as *John McDuff, III*, then establishing that textual linking would help to find the correct FR anchor for this instance of *John*, particularly since an agent's FR could realistically contain hundreds or thousands of people with the first name *John*.

We currently seek text sponsors for proper nouns by comparing candidate sponsors with the RE using properties from the HAS-NAME subtree of the ontology. If a candidate sponsor (a) matches the RE on HAS-PERSONAL-NAME and/or HAS-SURNAME, (b) does not have any conflicting values for name-oriented properties and (c) does not conflict in gender, if gender is known, then a coreference relation is established between the RE and the given candidate. If these conditions do not hold, then we seek an anchor for the proper name in the FR using the same matching algorithm. If there is a match, then we add any new information about the entity to the FR. If there is no match, a new FR anchor is created.

There are many outstanding issues related to name matching, some of which can be found in the literature and system descriptions devoted to the Named Entity Recognition task (cf. 9.4.1). Among them are the use of nicknames; the ambiguous use of bare surnames, as in the case of Bill Clinton and Hillary Clinton both appearing in the political press as "Clinton"; and the significant likelihood of creating false positive FR links if only name- and gender-based properties are considered. The latter necessitates a much more sophisticated matching algorithm that accounts for a wide variety of factors, including the fact that (a) people can have different social roles concurrently (someone can be both a physician and a researcher), (b) people can have different social roles over time (someone can be a teacher until he is 40, then become a politician), (c) many of a person's feature values can change over time—age, marital status, place of residence, etc., (d) there can be multiple individuals with the same name that have similar known feature values, and so on.

9.4.5.3 Resolve Indefinite NPs: NP-Indef

Indefinite NPs are NPs with the article *a* or *an* in English. Those that function as referring expressions (not descriptors, which should have been detected earlier) should, in most cases, create a new FR anchor, without the need to seek a

coreferential text sponsor or an existing FR anchor. Indeed, the difference between *A boy walked in the room* and *The boy walked in the room* is that, in the first case, the use of *a* asserts that this is a new entity in the discourse. There are, however, at least two exceptions to this rule. One involves generics, as in *A hungry lion will eat whatever prey it can find*. Another involves multi-text applications in which it is expected that a given event will be reported multiple times in multiple ways. Consider, for example, an application that compiles the reports about criminal activities, such as (23) and (24), from a number of different media sources.

(23) On Monday, April 15, 2011, an armed gunman walked into the First National Bank and demanded a million dollars in cash.

(24) Yesterday, a masked man carrying a handgun held up the First National Bank, demanding a large sum of cash.

In these texts, *an armed gunman* and *a masked man carrying a handgun* refer to the same person and should be coreferred in the FR. Of course, the heuristics for determining whether entities described as NP-Indef in different texts are coreferential are similarly complex as those for establishing FR coreference for proper names.

9.4.5.4 Resolve Bare NPs: NP-Bare

The referential status of bare NPs—which are NPs with no leading article, determiner, quantifier, or possessive pronoun—is quite complex and we have not found any linguistic descriptions that are sufficiently detailed to support the development of a truly broad-coverage algorithm for resolving their reference. We, therefore, begin with a crude algorithm that will be improved over time.

If NP-Bare is singular AND the ontological mapping is to an ABSTRACT-IDEA (e.g., WISDOM), RELIGION (e.g. JUDAISM) or ANIMAL-DISEASE (e.g., AUTISM)
  And if there is an FR anchor
  Then corefer with that FR anchor (cf. (25), (26))
  Else if NP-Bare is plural and the ontological mapping is to ANIMAL or PLANT
    And if there is an FR anchor
    Then corefer with that anchor (cf. 27)
    Else create a new FR anchor (cf. 28)

(25) **Wisdom** will help you to live well.

(26) Doctors do not know what causes **autism**.

(27) **Bears** like to eat berries.

(28) Do you have to do **physiotherapy**?

Among the many outstanding issues involved in the interpretation of generics is the use of modifiers to qualify what is, by default, a single abstract entity.

For example, if one says that justice in the US is not the same as justice in the former Soviet Union, the notion of "justice" is split into different categories and these instances of justice must be recorded separately in the FR. Similarly, if one talks about bears in the western United States or bears in a particular zoo, these are different sets of bears.

9.4.5.5 Resolve Third Person Pronouns

Third person pronouns always require either a textual sponsor or a real-world sponsor that is perceived using another perceptive apparatus: vision, hearing, touch or smell. Since we constrain the discussion here to textual input, all third person pronouns require textual sponsors. The selection of textual sponsors was briefly discussed in Step 3 of the of NP-Def resolution algorithm (Sect. 9.4.5.1).

9.4.5.6 Resolve Referential Verbs

Resolving the reference of events expressed as verbs in text is more difficult than resolving NPs that refer to events because NPs—at least in English—provide some information about whether or not the event is new to the discourse. For example, *a dispute* is most likely a new discourse event, whereas *the dispute* is most likely not. But what about *The kids argued*: Is it the description of a new event or is more information about a known argument being provided? It depends on the context and/or the agent's particular FR. In (29), the only instance of this string introduces a new event, whereas in (30) the 3 instances of *they argued* in the second sentence provide additional information about the argue event introduced in the first sentence.

(29) The kids argued all afternoon.

(30) The kids argued all afternoon. They argued about their toys, they argued about what cartoons to watch, and they even argued about who would brush the dog.

In addition to having or not having a textual sponsor, a verb can have or not have a FR sponsor. For example, for many people the 'appoint' event reported in (31) is known (even if some details, like the date, have been forgotten) and therefore the correct reference action would be to corefer this input to the existing FR anchor.

(31) In 1981, Sandra Day O'Connor was appointed the first female member of the US Supreme Court.

When comparing an event with a potential textual or FR sponsor, a conflict in case-role fillers or indications of time or place generally means a lack of coreference. For example, the two instances of READ in (32) and the two instances of RUN (realized as *jog*) in (33) must be different because they have different themes and temporal modifiers, respectively.

(32) John read *War and Peace* then, right afterward, he read *Anna Karenina*.

(33) Yesterday John jogged fast but today he jogged slowly.

However, different in case-role fillers and modifiers can be compatible with event coreference as long a they do not conflict: e.g., in (34), the first clause indicates the time of the event and the second one, its manner; and in (35), the first clause does not specify the theme of the event but the second one does.

(34) John jogged yesterday, but he jogged so slowly!

(35) John read all day yesterday. He was reading *War and Peace*.

There actually are contexts in which case-role fillers or modifiers can conflict even though the event instances are coreferential, as is the case when new information further specifies or corrects previously reported or known information. For example, in (36) the theme of the second instance is an ontological descendant of the theme of the first instance, and in (37) the first rendering provides incorrect information about the theme and location of the reading, which is later corrected.

(36) Remember Stephanie? She sent me a birthday card. Actually, it was an e-card.

(37) John was reading *War and Peace* at Starbucks all day yesterday. No, wait, not *War and Peace*—it was *Anna Karenina* that he was reading. And it wasn't at Starbucks, it was at Borders. He finished *War and Peace* at Starbucks last week.

To state the obvious, significant reasoning is needed to determine whether a verb should corefer with another event in the textual context, whether it should corefer with an existing FR anchor, or whether it should create a new FR anchor. Initial corpus investigation suggests that unless there is strong evidence of textual coreference for a verb, a textual coreference relation should not be established. This pragmatic generalization is, broadly speaking, the opposite of what we do with many classes of NPs, where we try hard to find a sponsor.

9.5 Final Thoughts: Semantics in Reference Resolution

Central to OntoSem processing of reference is the use of semantic features. Since semantic features are derived from text meaning representations, and since text meaning representations are generated using the lexicon and ontology as necessary resources, this aspect of reference resolution directly relies on these knowledge bases.

Without question, a correct semantic analysis of text entities can be a very strong heuristic for establishing coreference relations: e.g., two entities represented in TMR as DOG and GUIDE-DOG are likely to be coreferential, whereas two entities

represented as DOG and SWIMMING-POOL are not, using ontological distance as a measure.

But what happens if a referring expression is semantically underspecified, such as a pronoun? In this case, the system can exploit the recorded semantic constraints imposed on it by its selecting event to suggest its meaning. That hypothesized meaning can then be used as a proxy for the meaning of the RE itself. For example, if a text reads *He barked all night long*, the use of the pronoun *he* only lexically constrains the identity of *he* to ANIMATE. However, using the expressive means of *facets* (default, sem and relaxable-to), the ontology records the information that the most typical AGENT of BARK is DOG; that other possible AGENTs of BARK are CANINE, SEAL, SEA-LION; and that even HUMANs can BARK as an exception.

BARK

AGENT	default	DOG
AGENT	sem	CANINE, SEAL, SEA-LION
AGENT	relaxable-to	HUMAN

When evaluating the potential sponsors for *he* in our sentence, the system first attempts to find a DOG in the preceding context then, if it is not successful, it attempts to find a CANINE, SEAL or SEA-LION, then, if it is still not successful, it seeks a HUMAN.

Semantics can also be useful for establishing textual coreferents in contexts that show parallelism effects. Consider, the following example:

(38) Kevin hit the fence with a stick—he whacked it really hard.

This is a type of "repetition structure" (cf. [18]), which shows strong syntactic and semantic parallelism between sequential clauses: the verbs instantiate the same event (*hit* ¿ HIT, *whack* ¿ HIT) and the arguments are coreferential in a parallel manner across clauses (the syntactic subjects, which are the semantic AGENTs, are coreferential; and the syntactic direct objects, which are the semantic THEMEs, are coreferential). If the events in question were not coreferential, there would be a much diminished confidence that the pairs of arguments were coreferential, even if they were parallel at surface structure. For example, in (39) the direct objects (semantically, the THEMEs) are not coreferential.

(39) Kevin hit the fence with a stick and he regretted it later.

The real challenge in learning to exploit semantic features in the near- and mid-term is that their automatic computation is error prone at the current state of the art. This means that systems must be able to evaluate their own confidence in the value of each feature in each context (e.g., each lexical disambiguation decision), and then algorithms that use those features as input must determine whether or not a sufficient threshold of confidence has been achieved. As a gross simplification, if the text analysis system is not confident about its semantic analysis of most of the candidate sponsors for a given referring expression, it is probably better to rely on more surfacy heuristics, like the distance of the candidate from the RE and the

relative syntactic positions of the RE and the candidate. The tradeoff is essentially between high information content and high risk (for semantic features) and low information content and low risk (for more surface-level features).

References

1. Aramaki, E., Imai, T., Kashiwagi, M., Kajino, M., Miyo, K., Ohe, K.: Toward medical ontology using natural language processing. In: Proceedings of the OntoLex Workshop at the International Joint Conference on Natural Language Processing, pp. 53–58 (2005)
2. Bean, D., Riloff, E.: Corpus-based identification of non-anaphoric noun phrases. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, pp. 373–380. Association for Computational Linguistics (1999)
3. Boyd, A., Gegg-Harrison, W., Byron, D.: Identifying non-referential *it*: a machine learning approach incorporating linguistically motivated patterns. In: Proceedings of the ACL Workshop on Feature Selection for Machine Learning in NLP, Ann Arbor, pp. 40–47 (2005)
4. Chinchor, N.: MUC-7 Named Entity Recognition Task Definition. version 3.5 (1997). Accessed 17 Sept 1997
5. Clark, P., Porter, B., KM: The Knowledge Machine 2.0: Users Manual
6. Cohen, P., Chaudhri, V., Pease, A., Schrag, R.: Does prior knowledge facilitate the development of knowledge-based systems? In: Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), pp. 221–226. AAAI Press, Menlo Park (1999)
7. CYC. http://cyc.com/cyc/technology/technology/whatiscyc_dir/whatsincyc
8. de Marneffe, M.-C., MacCartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: Proceedings of LREC 2006, Genoa (2006)
9. Gundel, J.K., Hedberg, N., Zacharski, R.: Definite descriptions and cognitive status in English: why accommodation is unnecessary. J. Engl. Lang. Linguist. **5**, 273–295 (2001)
10. Huang, C.-R., Calzolari, N., Gangemi, A., Lenci, A., Oltramari, A., Prevot, L. (eds.) *Ontology and the Lexicon: A Natural Language Processing Perspective*. Studies on Natural Language Processing. Cambridge University Press, Cambridge/New York (2010)
11. Java, A., Nirenburg, S., McShane, M., Finin, T., English, J., Joshi, A.: Using a natural language understanding system to generate semantic web content. Int. J. Semant. Web Inf. Syst. **3**, 50–74 (2007)
12. Johnson, K.: What VP ellipsis can do, and what it can't, but not why. In: Baltin, M., Collins, C. (eds.), *The Handbook of Contemporary Syntactic Theory*, pp. 439–479. Blackwell Publishers, Oxford (2001)
13. Johnson, B.: A rule-based system for identifying pleonastic 'it'. Master's thesis, University of Maryland Baltimore County (2010)
14. Klein, D., Manning, C.D.: Fast exact inference with a factored model for natural language parsing. In: Advances in Neural Information Processing Systems 15 (NIPS-2002), pp. 3–10. MIT Press, Cambridge/London (2003)
15. Lappin, S., Leass, H.J.: An algorithm for pronominal anaphora resolution. Comput. Linguist. **20**, 535–561 (1994)
16. Li, Y., Musilek, P., Reformat, M., Wyard-Scott, L.: Identification of pleonastic *it* using the web. J. Artif. Int. Res. **34**(1), 339–389 (2009)
17. Lobeck, A.C.: Ellipsis: Functional Heads, Licensing, and Identification. Oxford University Press, New York (1995)
18. McShane, M.: A Theory of Ellipsis. Oxford University Press, Oxford/New York (2005)
19. McShane, M.: Advances in difficult aspects of reference resolution. Working Notes. ILIT working paper 01–09 (2009)

20. McShane, M.: Reference resolution challenges for intelligent agents: the need for knowledge. IEEE Intell. Syst. **24**(4), 47–58 (2009)
21. McShane, M., Nirenburg, S.: A knowledge representation language for natural language processing, simulation and reasoning. Int. J. Semant. Comput. **6**(1), (2012)
22. McShane, M., Nirenburg, S.: Dialog modeling within intelligent agent modeling. In: Jönsson, A., Alexandersson, J., Traum, D., Zukerman, I. (eds.) Proceedings of the IJCAI-09 Workshop on Knowledge and Reasoning in Practical Dialog Systems, pp. 52–59, Pasadena (2009)
23. McShane, M., Beale, S., Nirenburg, S., Jarrell, B., Fantry, G.: Inconsistency as a diagnostic tool in a society of intelligent agents. Artif. Intell. Med. **55**(3), 137–48 (2012)
24. McShane, M., Beale, S., Nirenburg, S.: Some meaning procedures of ontological semantics. In: Lino, M.T., Xavier, M.F., Ferreira, F., Costa, R., Silva, R. (eds.) Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC-2004). ELRA, Paris (2004)
25. McShane, M., Nirenburg, S., Beale, S.: An NLP lexicon as a largely language-independent resource. Mach. Transl. **19**(2), 139–173 (2005)
26. McShane, M., Fantry, G., Beale, S., Nirenburg, S., Jarrell, B.: Disease interaction in cognitive simulations for medical training. In: Oxley, L., Kulasiri, D. (eds.) MODSIM 2007 International Congress on Modelling and Simulation. Virginia Beach (USA) (2007)
27. McShane, M., Nirenburg, S., Beale, S., Jarrell, B., Fantry, G.: Knowledge-based modeling and simulation of diseases with highly differentiated clinical manifestations. In: Bellazzi, R., Abu-Hanna, A., Hunter, J. (eds.) Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME 07), pp. 34–43. Springer, Berlin/Heidelberg (2007)
28. McShane, M., Jarrell, B., Fantry, G., Nirenburg, S., Beale, S., Johnson, B.: Revealing the conceptual substrate of biomedical cognitive models to the wider community. In: Westwood, J.D., Haluck, R.S., Hoffman, H.M., Mogel, G.T., Phillips, R., Robb, R.A., et al. (eds.) Medicine Meets Virtual Reality 16, pp. 281–286. IOS Press (2008)
29. McShane, M., Nirenburg, S., Beale, S.: Resolving paraphrases to support modeling language perception in an intelligent agent. In: Proceedings of the Symposium on Semantics in Systems for Text Processing (STEP 2008), Venice, pp. 179–192 (2008)
30. McShane, M., Nirenburg, S., Beale, S.: Two kinds of paraphrase in modeling embodied cognitive agents. In: Proceedings of the Workshop on Biologically Inspired Cognitive Architectures, AAAI 2008 Fall Symposium, Washington (2008)
31. McShane, M., Beale, S., Nirenburg, S.: Reference resolution supporting lexical disambiguation. In: Proceedings of the 2010 IEEE Fourth International Conference on Semantic Computing, ICSC '10, pp. 56–59, Pittsburgh (PA), USA, IEEE Computer Society (2010)
32. McShane, M., Nirenburg, S., Beale, S.: Meaning-centric Language Processing. (In preparation)
33. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Wordnet: an on-line lexical database. Int. J. Lexicogr. Spec. Issue WordNet **3**, 235–244 (1990)
34. Mitkov, R.: Outstanding issues in anaphora resolution. In: Gelbukh, Al. (ed.) Computational Linguistics and Intelligent Text Processing, pp. 110–125 (2001)
35. Niles, I., Pease, A.: Towards a standard upper ontology. In: Welty, C., Smith, B. (eds.) Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001) (2001)
36. Nirenburg, S., Raskin, V.: Ontological Semantics. The MIT Press, Cambridge (2004)
37. Nirenburg, S., McShane, M.: Agents modeling agents: incorporating ethics-related reasoning. In: Guarini, M., Bello, P. (eds.) Proceedings of the Symposium on Moral Cognition and Theory of Mind at the 2012 AISB and IACAP World Congress, Birmingham (2012)
38. Nirenburg, S., Oates, T., English, J.: Learning by reading by learning to read. In: Proceedings of the International Conference on Semantic Computing (ICSC 2007), pp. 694–701. Springer, Berlin/New York (2007)
39. Nirenburg, S., McShane, M., Beale, S.: A simulated physiological/cognitive "double agent". In: Beal, J., Bello, P., Cassimatis, N., Coen, M., Winston, P. (eds.) Papers from the AAAI Fall Symposium, Naturally Inspired Cognitive Architectures. AAAI Press, Menlo Park (2008)

40. Nirenburg, S., McShane, M., Beale, S.: Aspects of metacognitive self-awareness in Maryland virtual patient. In: Proceedings of AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems. AAAI Press, Menlo Park (2010)
41. Panton, K., Matuszek, C., Lenat, D.B., 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, pp. 1–31. Springer, Berlin (2006)
42. Pustejovsky, J., Rumshisky, A., Castano, J.: Re-rendering semantic ontologies. In: Proceedings of the LREC Workshop on Ontologies and Lexical Knowledge Bases, Las Palmas (2002)
43. Rosse, C., Villaraza Mejino, J.L.: A reference ontology for biomedical informatics: the foundational model of anatomy. J. Biomed. Inf. **36**, 478–500 (2003)
44. Schank, R.C., Abelson, R.P.: Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures. Lawrence Erlbaum, Hillsdale (1977)
45. Stoyanov, V., Gilbert, N., Cardie, C., Riloff, E.: Conundrums in noun phrase coreference resolution: making sense of the state-of-the-art. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, *ACL '09*, vol. 2 , pp. 656–664. Association for Computational Linguistics, Stroudsburg (2009)
46. Vieira, R., Poesio, M.: Corpus-based development and evaluation of a system for processing definite descriptions. In: Proceedings of the 18th Conference on Computational Linguistics, *COLING '00*, vol. 2, pp. 899–903. Association for Computational Linguistics, Stroudsburg (2000)

Chapter 10
Ontology-Based Semantic Interpretation via Grammar Constraints

Smaranda Muresan

Abstract We present an ontology-based semantic interpreter that can be linked to a grammar through grammar rule constraints, providing access to meaning during language processing. In this approach, the parser will take as input natural language utterances and will produce ontology-based semantic representations. We rely on a recently developed constraint-based grammar formalism, which balances expressiveness with practical learnability results. We show that even with a lightweight ontology, the semantic interpreter at the grammar rule level can help remove erroneous parses obtained when we do not have access to meaning.

10.1 Introduction

Semantic parsing maps natural language utterances to formal representations of their underlying meaning. Recently, several machine learning approaches have been proposed for mapping sentences to their meaning representations [10, 11, 21, 29, 36–38]. These approaches differ in the mount of annotation required—unsupervised methods that start from syntactic parses [29], supervised methods that require annotation of full sentences [10, 36–38], supervised methods that require annotation of a small set of representative utterances that can be phrases, clauses or sentences [21]. Moreover, these approaches differ in the meaning representation languages they use—from λ-expressions [36–38] and command-like languages [10] to ontology-based representations [21]—and the integration, or lack thereof, of the meaning representations with grammar formalisms—Combinatory Categorial Grammars (CCGs) [35] are used by Zettlemoyer and Collins [37, 38], and Lexicalized Well-Founded Grammars [20, 25] are used by Muresan [21].

S. Muresan (✉)
School of Communication and Information, Rutgers University, New Brunswick, NJ 08901, USA
e-mail: smuresan@rutgers.edu

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_10,

Simultaneously, in recent years, there has been significant interest in ontology-based natural language processing, starting from ontology-base semantic representations [26], to using ontologies in various applications, such as question answering [1, 2], and building annotated corpora, such as the OntoNotes project [13].

In this chapter, we present an ontology-based semantic interpreter that can be linked to a grammar through grammar rule constraints, providing access to meaning during parsing, generation and learning. The parser will take as input natural language text and will produce ontology-based semantic representations. We integrate this in a learning framework through the use of our Lexicalized Well-Founded Grammar formalism, which is a constraint-based formalism, which balances expressiveness with provable learnability results [20,22,23,25]. We present several principles that allow for grammar reversibility and parsing termination (parsing and interpretation intertwine). The semantic interpreter can use either a lightweight ontology—based just on information regarding the semantic roles of verbs, prepositions, the attributes of adjectives, adverbs and also nouns that appear in noun-noun compounds, or a heavyweight ontology based on a hierarchy of concepts and roles. We show that even with a lightweight ontology, the semantic interpreter at the grammar rule level can help remove 40 % of erroneous parses. Moreover, we discuss how our framework can be used to support the idea that "a lexicon can sometimes be the basis for the development of a practical ontology" [12], with experiments in the health domain.

First, we review the Lexicalized Well-Founded Grammar formalism [20, 25], emphasizing the representation of language expressions, the structure of the lexicon, and how semantic composition and interpretation can be encoded as constraints at the grammar rule level. In Sect. 10.3, we present the ontology-based semantic interpretation (local vs global interpretation, principles, and the semantic interpreter). In Sect. 10.4, we show how the semantic interpreter could be used to build terminological knowledge bases from text, and show preliminary results on how this interpretation at the grammar rule level can help remove some of the erroneous utterance parses obtained when we do not have access to meaning. In Sect. 10.5 we discuss the issue of ambiguity, pointing to future work on enhancing the ontology with probabilities/weights. We conclude in Sect. 10.6.

10.2 Lexicalized Well-Founded Grammar

Lexicalized Well-Founded Grammar (LWFG) is a recently developed formalism that balances expressiveness with practical—and provable—learnability results [20, 22, 23, 25]. Formally, LWFG is a type of Definite Clause Grammar [27] that is decidable in polynomial time and can be learned from examples also in polynomial time. In LWFG, each string is associated with a syntactic-semantic representation, called *semantic molecule*, and grammar rules have two types of constraints, one for semantic composition (Φ_c)—defines how the meaning of a natural language expression is composed from the meaning of its parts—and one for semantic

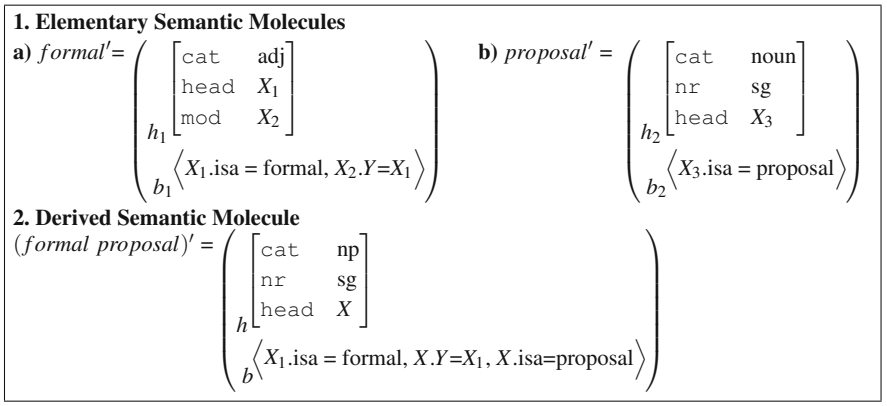

Fig. 10.1 (1) Elementary semantic molecules for the adjective *formal* (**a**) and the noun *proposal* (**b**); (2) A derived semantic molecule for the noun phrase *formal proposal*

interpretation (Φ_i)—validates the semantic constructions based on a given semantic model (i.e., an ontology).

10.2.1 Semantic Molecule: A Syntactic-Semantic Representation

A *semantic molecule* associated with a natural language string w, is a syntactic-semantic representation, $w' = \binom{h}{b}$, where h (*head*) encodes syntactic/compositional information acting as valence for molecule composition, and b (*body*) is the actual semantic representation of the string w.

This representation is simple enough to allow learning and tractable inferences, but expressive enough for natural language [20]. The representations associated with the lexical items are called *elementary semantic molecules*, while the representations built by the combination of others are called *derived semantic molecules* (see Fig. 10.1).

Formally, the *head* (h) of a semantic molecule is a one-level feature structure (i.e., feature values are atomic). In Fig. 10.1 the heads are shown as attribute-value matrices (AVMs). Each semantic molecule has at least two attributes, one encoding the syntactic category of the associated string, `cat`, and the other encoding the semantic head of the string, `head`. In addition, feature attributes for agreement and other grammatical features can be present (e.g., `nr` for number agreement). All these sets of attributes are finite and are known a priori for each syntactic category. Being a one-level feature structure, no recursive or embedded structures are allowed, which makes this representation appealing for a learning framework. Recursion in the grammar is obtained through recursive grammar rules and semantic

composition constraints, which are described in Sect. 10.2.2. In Fig. 10.1, we show the elementary semantic molecules for the adjective *formal* and the noun *proposal*. For the adjective, the semantic molecule head contains in addition to the `cat` and `head` attributes, an attribute `mod`, which specifies the index of the modified noun. This information is necessary for combining an adjective and a noun to obtain a noun phrase (e.g., *formal proposal*). For the noun, we have additional attributes needed for agreement, such as `nr` for number (which can be singular `sg` or plural `pl`).

The body, b, of a semantic molecule is a flat representation, called OntoSeR (Ontology-based Semantic Representation) [20, 21]. OntoSeR is built as a conjunction of atomic predicates (AP), $\langle concept \rangle.\langle attr \rangle = \langle concept \rangle$. The formal definition is given below:

$$
\begin{aligned}
\langle OntoSeR \rangle &\stackrel{\text{def}}{=} < AP > \mid < OntoSeR >< lop >< OntoSeR > \\
< AP > &\stackrel{\text{def}}{=} < conceptID > . < attr >=< concept > \\
< AP > &\stackrel{\text{def}}{=} < conceptID >=< conceptID >< coord >< conceptID > \\
< concept > &\stackrel{\text{def}}{=} < conceptID > \mid < conceptName > \\
< conceptID > &\stackrel{\text{def}}{=} < logicalVariable > \\
< conceptName > &\stackrel{\text{def}}{=} < lexicalWord > \\
< attr > &\stackrel{\text{def}}{=} < attrID > \mid < attrName > \\
< attrID > &\stackrel{\text{def}}{=} < logicalVariable > \\
< attrName > &\stackrel{\text{def}}{=} < lexicalWord > \\
< coord > &\stackrel{\text{def}}{=} < lexicalCoord > \\
< lop > &\stackrel{\text{def}}{=} \wedge
\end{aligned}
$$

where $\langle lop \rangle$ is the logical operator, which we consider to be the logical conjunction $\wedge$. The $\langle coord \rangle$ operator is one of the linguistic coordinators, such as *and*, *or*, *but*. $\langle conceptID \rangle$ is a variable denoting a concept identifier in the semantic model; $\langle conceptName \rangle$ is the name of a concept in the semantic model; $\langle attrID \rangle$ is a variable denoting a slot; and $\langle attrName \rangle$ is the name of a slot in the semantic model. The slot is either a property or a relation. The richness of the semantic model can range from just a lightweight ontology—encoding the admissibility relations that we can find at the level of lexical entries, such as thematic roles of verbs and prepositions—to a heavyweight ontology with hierarchy of concepts and roles, and relations among concepts. OntoSeR can be seen as an ontology-query language, which is sufficiently expressive to represent many aspects of natural language and yet sufficiently restrictive to facilitate learning.

For example, the OntoSeR for the adjective *formal* (Fig. 10.1) is $\langle X_1.isa = formal, X_2.Y = X_1 \rangle$, which says that the meaning of an adjective is a concept ($X_1.isa = formal$), which is a value of a property of another concept ($X_2.Y = X_1$) in the semantic model. The variable X_2 will be instantiated through composition, when the adjective *formal* will be combined with a noun, e.g., *proposal* to build a noun phrase *formal proposal*. The variable Y will be instantiated by the semantic interpretation based on the semantic model (e.g., Y = `manner`). The semantic composition and semantic interpretation are modeled in our framework as grammar rule constraints, as detailed in the next section.

10.2.2 Semantic Composition and Interpretation as Grammar Constraints

The lexicon in LWFG consists of words paired with *elementary semantic molecules* (w, $\binom{h}{b}$). The lexicon in LWFG is not learned. Unlike other lexicalized rich grammar formalisms, such as Combinatory Categorial Grammars [35], the lexicon in LWFG does not specify the syntactic context in which the word is anchored. That context will be learned from examples, by learning grammar rules and compositional constraints.

In addition to the lexicon, a LWFG has a set of constraint grammar rules, which can be recursive and where the nonterminals are augmented with tuples formed of strings and their semantic molecules (w_i, $\binom{h_i}{b_i}$). For example, a simple grammar rule for a noun phrase, such as *formal proposal* could be:

$$NP(w, \binom{h}{b}) \rightarrow Adj(w_1, \binom{h_1}{b_1}), Noun(w_2, \binom{h_2}{b_2}) : \Phi_c(h, h_1, h_2), \Phi_i(b).$$

Grammar rules have two types of constraints—one for *semantic composition*, Φ_c, and one for *semantic interpretation*, Φ_i. The composition constraints Φ_c are applied to the heads of the semantic molecules, the bodies being concatenated through logical conjunction together with a variable substitution given by the Φ_c constraints. Figure 10.1(2) shows that the body of the semantic molecule for the noun phrase *formal proposal* is a concatenation of the bodies of the adjective *formal* and the noun *proposal*, together with the variable substitution $\{X_2/X, X_3/X\}$ given by Φ_c, which is a system of equations—a simplified version of "path equations" [33]. For the grammar rule above, $\Phi_c(h, h_1, h_2) = \{h.cat = np, h.head = h_1.mod, h.head = h_2.head, h.nr = h_2.nr, h_1.cat = adj, h_2.cat = noun\}$ (the part $\{h.head = h_1.mod, h.head = h_2.head\}$ indicates that the variable denoting the semantic head of the noun phrase *formal proposal* (X), should be the same as the variable denoting the semantic head of the noun *proposal* (X_2), and also the same with the variable denoting the `mod` attribute of the adjective *formal* (X_1), giving precisely the substitution mentioned above $\{X_2/X, X_3/X\}$).These constraints are learned together with the grammar rules.

The semantic interpretation constraints, Φ_i, represent the validation based on a semantic model, and are not learned. Currently, Φ_i is a predicate which can succeed or fail—when it succeeds, it instantiates the variables of the semantic representation with concepts/slots in the semantic model [20, 21]. For example, given the phrase *formal proposal*, Φ_i succeeds and returns $\langle X_1.isa = formal, X.manner = X_1, X.isa = proposal \rangle$, while given the phrase *fair-hair proposal* it fails. The semantic interpretation constraint, Φ_i is important for the disambiguation required for some phenomena (e.g., prepositional phrase attachment, coordinations), and for the semantic interpretation of phenomena not usually analyzed by current broad-coverage grammars or statistical syntactic parsers (e.g., prepositions, noun-noun compounds).

Before describing the ontology-based interpretation in the next section we give a brief overview of the learning model for LWFGs.

10.2.3 LWFG Learning Model

Unlike stochastic grammar learning for supervised statistical parsing (e.g., [4, 5]), LWFG is suited to learning in data-poor settings. And unlike previous formalisms used for deeper representation, such as Tree Adjoining Grammars [15], Head-driven Phrase Structure Grammars [28] or Lexical Functional Grammars [3,16], the LWFG formalism is accompanied by a formal guarantee of efficient learnability [20, 22, 23, 25]. Learnability results have been proven for some classes of Combinatorial Categorial Grammars [35], but to our knowledge no tractable learning algorithm has been proposed.

LWFG's learning is a relational learning framework, which characterizes the "importance" of substructures in the model not simply by frequency, as in most previous work, but rather linguistically, by defining a notion of "representative examples" that drives the acquisition process. In formal grammar learning theory it has been shown that learning from good examples, or representative examples, is more powerful than learning from all the examples [9]. Informally, representative examples are "building blocks" from which larger structures can be inferred via reference to a larger unannotated, or weakly annotated corpus (called the generalization corpus). For example, *effect*, *the effect*, and *adverse effect*, annotated similarly to *proposal* and *formal proposal* shown above, might all be representative examples for the English nominal system; *adverse* annotated similarly to *formal*, might be a representative example for English adjectives; and the unannotated generalization corpus might contain *the major adverse effect*. With this information, it is possible to learn grammar rules permitting English noun heads to be modified by a determiner and multiple adjectives (learning recursive grammar rules).

We treat grammar learning as an inductive logic programming (ILP) problem, and we have defined a complete grammar lattice as a search space for grammar induction, proving a learnability theorem for LWFGs [20, 25]. This is an important theoretical result since it shows the learnability of a complex class of

syntactic-semantic grammars from positive examples. With this theorem as the formal underpinning, we have defined three algorithms for LWFG learning and have studied their efficiency, search space properties, and annotation effort required for the training data [23].

- **Learning from ordered representative examples.** In this case, the learner is presented with an ordered set of representative examples, that is learning from simpler examples first, and then gradually from more complex examples. The search space for grammar learning is a boolean algebra, and the efficiency is polynomial [23]. The annotation effort is reduced, since only the representative examples need to be annotated, while the generalization corpus can be unannotated. The order of magnitude for the representative examples is hundreds of examples, while the generalization corpus can be several thousands.
- **Learning from unordered representative examples.** A practical problem of the previous algorithm is that in some cases it is hard to determine a priori the right order of the representative examples. Thus, we introduced a second algorithm which learns a grammar from unordered representative examples using an iterative method with theory revision. We proved that the grammar converges to the same target grammar as the previous algorithm [23]. This algorithm is polynomial and the search space is a complete grammar lattice [20, 25].
- **Learning from entire generalization corpus.** When the learner does not know the representative examples, we introduced a polynomial algorithm able to learn from the entire generalization corpus using again an iterative method with theory revision [23]. In this case, the entire generalization corpus needs to be annotated.

Due to the property of the search space all the above algorithms converge to the same target grammar. These algorithms belong to the class of Inductive Logic Programming methods (ILP), based on entailment [8]. Like all existing ILP methods, our algorithms are able to use background knowledge, which in our case includes the lexicon (pairs of words and their elementary semantic molecules), the previously learned grammar rules and constraints, and a robust parser as an innate inference engine.[1] Unlike existing ILP methods, the search space for our induction is a complete lattice, ensuring polynomial efficiency of the learning algorithms.

Annotation of training data. In order to learn a LWFG, annotations for phrases, clauses, and sentences are required, in the form of semantic molecules discussed in Sect. 10.2.1. It is clear that even for a small corpus of representative data which our learning model needs, writing by hand these annotations might be a difficult task. We have developed an annotation tool that, through interaction with the LWFG parser and lexicon, replaces manual assignment of full semantic representations with the manual specification of unlabeled dependencies between words (or chunks). This could be accomplished since in our framework the lexicon

[1] We call the parser robust since when no full parse is possible it returns the minimum number of chunks.

is given and the semantic representation of a phrase is just a concatenation of the semantic representations of its words together with variable bindings that indicate dependencies (obtained via Φ_c; see Sect. 10.2.2). Description of the annotation tool is left for a future publication.

10.3 Ontology-Based Semantic Interpretation

The $\Phi_i(b)$ constraint of LWFGs can be seen as a *local semantic interpretation* at the utterance/grammar rule level, providing access to meaning during parsing/generation.[2] It is built using a meta-interpreter with *freeze* [31]. We give the details of this interpreter in Sect. 10.3.2.

Before we could talk about the semantic interpreter, and the principles that govern the semantic interpretation, we first discuss the levels of representation needed to get from natural language utterances to knowledge: utterance, text, and ontology levels.

10.3.1 Levels of Representation

Once we learn a LWFG, we can use a syntactic/semantic parser and semantic/pragmatic interpreter to transform utterances to semantic representations. We have three levels of semantic representation: the utterance level, the text level and the ontology level (see Fig. 10.2).

The syntactic/semantic parser in conjunction with the learned grammar gives us directly the semantic representation (OntoSeR) of each utterance via the Φ_i constraints. This is the *utterance level* representation. During parsing, we have two types of representations: **OntoSeR$^-$**—the semantic representation obtained before the semantic interpretation constraint Φ_i is applied; and **OntoSeR$^+$**—the semantic representation after the semantic interpretation constraint Φ_i is applied. Thus, Φ_i can be seen as a **local level semantic interpretation**. In Fig. 10.2 we show an example of OntoSeR$^-$ and OntoSeR$^+$ for the utterance *a virus that does not persist in the blood serum*. At OntoSeR$^-$ both the conceptIDs and attrIDs remain variables. For example, the semantic roles of the verb *persist*, the meaning of the preposition *in*, and the relations among the nouns *blood* and *serum* are still variables: `P1`, `P2` and `P3`, respectively. At OntoSeR$^+$ the attrIDs become constant, while the conceptIDs remain variables to allow further composition to take place. In the example given in Fig. 10.2, the semantic interpretation constraint Φ_i instantiates the attrIDs variables with roles from the semantic model (i.e., ontology)—`th`, `loc`, and the dummy `of`, respectively. This example shows the representation of several

[2] Lexicalized Well-Founded Grammars are reversible grammars.

utterance level representation OntoSeR$^-$ (before local semantic interpretation Φ_i.)

$\langle$(A.det=a)$_a$, (A.isa=virus)$_{virus}$, (A.isa=that)$_{that}$, (B.tense=pr)$_{does}$, (B.neg=y)$_{not}$, (B.isa=persist, B.P1=A)$_{persist}$, (P2.isa=in, B.P2=C)$_{in}$, (C.det=the)$_{the}$, (D.isa=blood, C.P3=D)$_{blood}$, (C.isa=serum)$_{serum}$ $\rangle$

utterance level representation OntoSeR$^+$ (after local semantic interpretation Φ_i.)

$\langle$(A.det=a)$_a$, (A.isa=virus)$_{virus}$, (A.isa=that)$_{that}$, (B.tense=pr)$_{does}$, (B.neg=y)$_{not}$, (B.isa=persist, B.th=A)$_{persist}$, (loc.isa=in, B.loc=C)$_{in}$, (C.det=the)$_{the}$, (D.isa=blood, C.of=D)$_{blood}$, (C.isa=serum)$_{serum}$ $\rangle$

text level knowledge representation TKR (after assertion.)

$\langle$(1.det=a)$_a$, (1.isa=virus)$_{virus}$, (1.isa=that)$_{that}$, (2.tense=pr)$_{does}$, (2.neg=y)$_{not}$, (2.isa=persist, 2.th=1)$_{persist}$, (loc.isa=in, 2.loc=3)$_{in}$, (3.det=the)$_{the}$, (4.isa=blood, 3.of=4)$_{blood}$, (3.isa=serum)$_{serum}$ $\rangle$

ontology-level knowledge representation OKR (after global semantic/pragmatic interpretation)

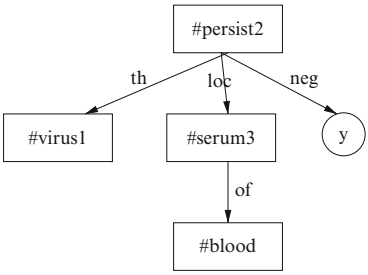

Fig. 10.2 Levels of representation for the utterance *a virus that does not persist in the blood serum*

linguistic phenomena, such as relative clauses (*virus that* ...), negation (*does not persist*), and noun compounds (*blood serum*). For readability, we indicate what part of OntoSeR corresponds to each lexical item. It can be noticed that OntoSeR contains representations of both ontological meaning (concepts and relations among concepts) as well as extra-ontological meaning, such as tense (*B*.`tense`=pr). Both at the OntoSeR$^-$ and OntoSeR$^+$ levels, we can exploit the reversibility of the grammar since both these representations are used during parsing/generation.

After parsing each utterance, their semantic representations form the *text level knowledge representation* **TKR**. The variables become constants, and no composition can happen at this level. However, we still have (indirect) reversibility, since TKR represents all the asserted OntoSeRs$^+$. Therefore, all the information needed for reversibility is still present. In Fig. 10.2, we see that TKR is the same as OntoSeR$^+$, except that the variables are constants (e.g., `A` becomes `1`, `B` becomes `2`).

In order to transform these representations to knowledge (*ontology-level knowledge representation* **OKR**), we use a semantic/pragmatic interpreter that implements task-specific interpretation and filtering.While the semantic interpretation at the grammar level Φ_i can be seen as local semantic interpretation, the interpretation

from TKR to OKR can be seen as a **global semantic interpretation**. OKR is a directed acyclic graph (DAG) $G = (V, E)$. Vertices V are concepts (corresponding to nouns, verbs, adjectives, adverbs, pronouns, cf. Quine's criterion [34, p. 496]), or values of extra-ontological properties, such as `y` corresponding to the `neg` property. Edges, E, are semantic roles given by verbs, prepositions, adjectives and adverbs, or are extra-ontological properties, such as `neg` (negation). At the OKR level we assume the *principle of concept identity* which means that there is a bijection between a vertex in OKR and a referent (see Sect. 10.3.3). In Fig. 10.2, we give an example of OKR for the same utterance *a virus that does not persist in the blood serum* obtained using our semantic/pragmatic interpreter. Determiners, even if represented at the level of OntoSeR, they are not interpreted at the OKR level (they are filtered by the global level interpreter). At OKR we have both concepts (e.g., #blood), and instances of concepts (e.g., #virus1, #persist2) (see Sect. 10.3.3). A concept and an instance of this concept are two different vertices in OKR, having the same name. We notice that vertices are either concepts/instances of concepts or values of extra-ontological properties.

In this chapter, the semantic interpretation (both local and global) is based only on a lightweight ontology which contains only the admissibility relations that we can find at the level of lexical entries (i.e., we do not use synonymy, anaphora and predefined hierarchies of concepts and roles). For the verb thematic roles we considered the thematic roles derived from Dorr's LCS Database (e.g., `ag=agent`, `th=theme`, `prop=proposition`) [7]. For adjectives and adverbs we took the roles (properties) from WordNet [19]. For prepositions we considered the LCS Database. We also have added specific/dummy semantic roles when they were not present in these resources (e.g., `of` between #blood and #serum).

Natural Language as Problem Formulation Principle. The TKR contains only the logic-based problem formulation that can be further solved using logic as problem solving [18]. That is, the local semantic interpreter Φ_i does not deal with deep reasoning, meaning that we are concerned only with the meaning explicitly given in text. Thus, TKR can contain the representation of a paradox formulation in natural language, even if the reasoning about its solution cannot be emphasized. This principle applies only to the local semantic interpreter Φ_i and not to the global interpreter, where reasoning could take place. This principle assures the tractability of Φ_i, which in turns assures the termination of parsing.

10.3.2 The Local Ontology-Based Semantic Interpreter

The local semantic interpretation is performed at the rule level through $\Phi_i(b)$, which is built using a meta-interpreter with *freeze* [31]. Given the definition of OntoSeR given in Sect. 10.2.1 and the notation $\Phi_i(b) = b'$, the interpretation of OntoSeR is given below:

$$(AP)' \leftarrow (postpone\ (AP))'$$
$$(OntoSeR_1\ \langle \text{lop} \rangle\ OntoSeR_2)' \leftarrow OntoSeR_1'\ \langle \text{lop} \rangle\ OntoSeR_2'$$
$$postpone\ (AP) \leftarrow freeze\ (X \in var\ (AP)\ , AP)$$

The above definition entails that an atomic predicate, AP, is postponed through the *freeze* predicate until at least one of its variables becomes instantiated. Thus our semantic interpreter is a meta-interpreter with *freeze* [31]. This allows a nondeterministic efficient search in the ontology. The search strategy of the meta-interpreter is independent of the actual representation of the ontology, allowing an interface with any ontology at the level of atomic predicate meaning. The ontology-based interpretation is not done during the composition operation, but afterwards. Thus, for example, the head of the noun phrase *formal proposal* does not need to store the slot Y, a fact that allows us to use flat feature structures for representing the head of the semantic molecule. At this point, when Φ_i is applied, the variable Y becomes instantiated with the value taken from the ontology (e.g., `manner`).

The meta-interpreter can be enhanced with generative ontology[3] $X' \leftarrow X.isa = X'$, $(X.Y = Z)' \leftarrow X'.Y' = Z'$ (admissible concept rule), $(Y = Z) \leftarrow X.Y = X.Z$ (well-formedness principle for distinct simultaneous roles), $X.Y = Z \leftrightarrow Z.Y^{-1} = X$ (inversion principle), and also with a set of admissible affinities and role relations specified as atomic axioms. The latter refers to the ontologically admissible combinations of concepts and relations (e.g., $event.ag = person$, $ag.isa = by$).

The OntoSeR is an ontology independent semantic representation, in the same way an ontology is a language independent logical structure. The meta-interpreter allows all the logical operators (i.e., conjunction, disjunction, negation) and provides the soundness of meaning. For negation, the meta-interpreter either adopts the negation as failure strategy of logic programming, or treats negation as an atomic predicate that will be handled at the ontology level. The *freeze* interpreting technique provides the soundness of logic programs with negation as failure. Two predicates are implemented for asserting to and querying the ontology, respectively. In the querying process, different OntoSeRs can have the same answer, thus transforming the problem of logical equivalence viewed as "meaning identity" [32] into equivalence viewed as concept identity. This ensures the computational tractability requirement for a semantic framework.

Having the local semantic interpreter Φ_i is important for the disambiguation required for some phenomena (e.g., prepositional phrase attachment, coordinations),

[3] Starting from a skeleton ontology, generative ontologies are formed by rules for combining concepts using semantic roles (binary relations) as binders: "The role relations express possible relations among the nodes in the lattice constituting the ontology. Thereby they make possible the generation of an infinite number of ontological nodes in the lattice, thus establishing a generative ontology." [14]

and for the semantic interpretation of phenomena not usually analyzed by current broad-coverage grammars or statistical syntactic parsers (e.g., prepositions, noun-noun compounds). We discuss the issue of ambiguity in Sect. 10.5, while in Sect. 10.4 we show some preliminary results of how Φ_i could help.

10.3.3 Global Semantic Interpreter

In this chapter, the global (task-specific) semantic interpretation (from TKR to OKR) is geared towards terminological interpretation. We filter determiners, and some verb forms, such as aspect, since temporal relations appear less in terminological knowledge than in factual knowledge. However, we treat modals and negation, as they are relevant for terminological knowledge. The semantic interpreter considers both concepts (e.g., #blood), and instances of concepts (e.g., #virus1, #persist2). Concepts are denoted in OKR by #name_concept. An instance of a concept is denoted by the name of a concept followed by the instance number (e.g., #virus1). A concept and an instance of this concept are two different vertices in OKR, having the same name. Concepts form a hierarchy based on the `subsume` relation (`sub`), which is the inverse of the `isa` relation. At the OKR level we have the **principle of concept identity**, which means that there is a bijection between a vertex in OKR and a referent. For example, if we do not have pronoun resolution, the pronoun and the noun it refers to will be represented as two separate vertices in the graph. Currently, our global semantic/pragmatic interpreter implements only a **weak concept identity principle** that facilitates *structure sharing* and *inheritance* (we do not have anaphora resolution, for example). To give these two properties we first introduce some notations.

A DAG is called *rooted* at a vertex $u \in V$, if there exists a path from u to each vertex of the DAG. We have the following definition:

Definition 10.1. Two subDAGs rooted at two vertices u, u' are equal if the set of the adjacent vertices to u and u' respectively, are equal and if the edges incident from u and u' have the same semantic roles as labels.

Property 10.1 (Structure Sharing). In an OKR, all vertices $u, u' \in V$ with the same name, and whose subDAGs are equal are identical (i.e., the same vertex in OKR).

Using a hash table, there is a linear algorithm $O(|V| + |E|)$ which transforms an OKR to an equivalent OKR which satisfies Property 10.1.

Property 10.2 (Inheritance). A concept in a hierarchy of concepts can be linked by the `sub` relation only to its parent(s), and not to any other ancestors. A subDAG defining a property of a concept from the hierarchy of concepts can be found only once in the OKR at the level of the most general concept that has this property.

In the next section we discuss how starting with a lightweight ontology and using a learned grammar and our semantic interpreter that implements the weak concept identity principle, we can get a step closer to building ontologies/terminologies from text.

10.4 Knowledge Acquisition and Querying Experiments

We have performed a pilot experiment, whose purpose is two-fold: (1) to show that the semantic representation, interpretation and parsing can be used to acquire knowledge from text and to query this knowledge using natural language questions, obtaining precise answers at the concept level; and (2) to show that the local semantic interpretation at the grammar rule level, Φ_i, could help in disambiguation, even if it is based on a lightweight ontology.

The task was reduced to terminological knowledge, where the input text consists of definitions in the medical domain. These definitions were automatically extracted from text by DEFINDER system [17, 24] from consumer-health articles. Before describing our acquisition and querying experiment, we briefly present our method for learning a syntactic-semantic grammar for definitions. The grammar was learned using the LWFG learning model described in Sect. 10.2.3. We chose the representative examples guided by the type of phenomena we wanted to model and which occurred in a development set of medical definitions (approximatively 80 definitions). The phenomena included: complex noun phrases (e.g., noun compounds, nominalization), prepositional phrases, relative clauses and reduced relative clauses, finite and non-finite verbal constructions (including, tense, aspect, negation, and subject-verb agreement), copula *to be*, raising and control constructions. Since our goal is to query the acquired terminological knowledge using natural language questions, we also learned grammar rules for wh-questions (including long-distance dependencies). In order to learn the grammar, we annotated 151 representative examples and 448 examples were used as a generalization corpus. We should mention that the representative examples were not full definitions (e.g., representative examples for learning grammar rules for noun phrases include *formal proposal, the proposal, paper and poster* annotated with their semantic molecules as exemplified in Fig. 10.1). Annotating these examples requires knowledge about categories and their attributes. We used 31 categories (nonterminals such as NP, ADJP) and 37 attributes (e.g., category, number, person). Regarding the lexical items, we used a total number of 13 lexical categories (i.e., preterminals, or parts of speech) and 46 elementary semantic molecule templates. For example, the nouns have three types of elementary semantic molecules, which corresponds to basic nouns, modifier nouns (e.g., in case of noun compounds) and nominalizations (where the semantic representation is similar to the representation of a verb). For grammar learning, only a reduced lexicon is needed (e.g., only a few lexical items are given for every open word class, such as nouns (20), verbs (13, 6 of which are for raising and control verbs), adjectives (14), adverbs (9), proper nouns (4)). For the lightweight ontology, used only in the acquisition/querying experiment and not during grammar learning, we only used information regarding the semantic roles of verbs, prepositions, attributes of adjectives, adverbs and also nouns that appear in noun-noun compounds (i.e., no synonymy, or hierarchy of concepts and roles). For the semantic roles of verbs and prepositions we extracted the thematic roles from the "LCS Database" [7]. For adjectives and adverbs we used information from WordNet [19].

However, since we used medical definitions, these resources do not contain all the required information and thus we were forced to manually introduce this missing information (especially for adjectives, nouns, and specific roles of prepositions).

The corpus of definitions used in the acquisition and querying experiment consists of the correctly extracted definitions by DEFINDER, which were used in DEFINDER's evaluation [24], and which were different from our development set used in building the representative examples for grammar learning. In the next two sections we present and discuss the acquisition and querying experiments.

10.4.1 Acquisition of Terminological Knowledge from Consumer Health Definitions

In this experiment we tested the use of the learned grammar, syntactic/semantic parser and the semantic/pragmatic interpreter based on the lightweight ontology to acquire terminological knowledge from consumer health definitions. While our grammar covered all the constructions present in the corpus of definitions, we obtain besides the correct semantic representations also incorrect semantic representations, which shows that our lightweight ontology is not enough to remove all erroneous parses. In order to gain further insight, we looked at the number of alternative semantic representations obtained with and without our local semantic interpreter Φ_i. Without Φ_i, the average number of representation obtained by the parser is 2.53 per definition. After Φ_i is applied, the average number of different representations obtained for a definition is 2.00. This result shows that even with a lightweight ontology our semantic interpreter helps remove some erroneous parses. However, it is not enough to obtain only the correct semantic analysis in all cases. Thus, we developed the system to allow a user to manually select the correct OKR, which was then added to the knowledge base. The selection of the OKR-level of representation for human validation is due to the fact that this representation is much more "readable" for a user than the OntoSeR levels (as can be seen from Fig. 10.2).

In order to further discuss the processes of knowledge acquisition, we present an example of constructing a hierarchy of concepts from definitions of *hepatitis*, *Hepatitis A* and *Hepatitis B*. The definitional text and OKRs are presented in Fig. 10.3, OKR being shown only for the last two definitions for readability reasons. The acquisition of knowledge can be done directly, since we consider both concepts (#hepatitis, #blood) and instances of concepts (#virus25,#virus33) in our OKR representation [26].

The defined term is always a concept, and it is part of the `sub` hierarchy. The concepts in the `sub` hierarchy are presented in bold in Fig. 10.3. All the definitional properties of concepts are directly linked to the concept vertex (facilitated by our interpretation of copula be-predicative). For example, even if in the text we have *Hepatitis B is an acute viral hepatitis*, the properties "acute" and "viral" are linked to the concept #hepatitisB and not to the concept #hepatitis. This is obtained since only

1. Hepatitis is a disease caused by infectious or toxic agents and characterized by jaundice, fever and liver enlargement.
2. Hepatitis A is an acute but benign viral hepatitis caused by a virus that does not persist in the blood serum.
3. Hepatitis B is an acute viral hepatitis caused by a virus that tends to persist in the blood serum.

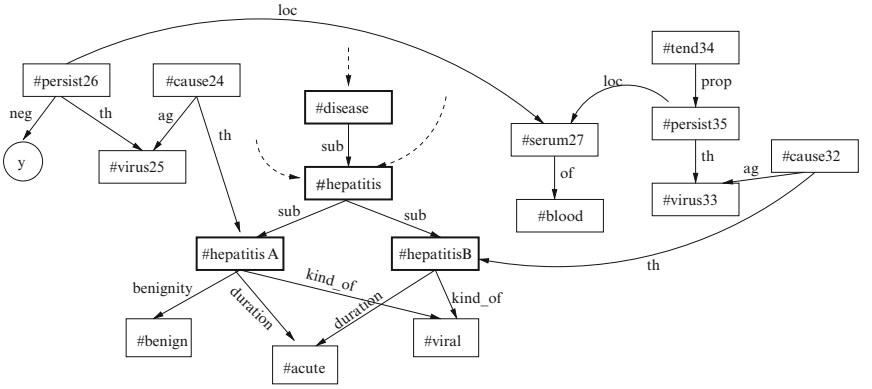

Fig. 10.3 Terminological knowledge acquired from consumer health definitions

#hepatitis was previously part of the `sub` hierarchy. If the concept #viral_hepatitis is present, then this most specific concept is selected as the direct parent of #hepatitisB.

In addition to the concepts that are defined, we can also have *concepts that are referred* (i.e., they are part of the definition of a medical term), *if they do not have any modification* (e.g., #blood in definition of Hepatitis A, and Hepatitis B). If a referred concept has modifications, it is represented as an instance of a concept in OKR. As a consequence, various verbalizations of concept properties can be differentiated in OKR, allowing us to obtain direct answers that are specific to each verbalization. For example, the term *virus* appears in the definition of both *Hepatitis A* and *Hepatitis B*. In OKR, they are two different instances of a concept, #virus25 and #virus33, since they have different modifications: *persists in the blood serum*, and *does not persists in the blood serum*, respectively. These modifications are an essential part of the *differentia* of the two concepts #hepatitisA and #hepatitisB, causing the distinction between the two. When we ask the question *What is caused by a virus that persists in the blood serum?* (Q1 in Fig. 10.4), we obtain only the correct answer #hepatitisB (A1 in Fig. 10.4).

Another important aspect that shows the adequacy of our representation for direct acquisition and query is the OKR-equivalences that we obtain for different syntactic forms. They are related mainly to verbal constructions. Among OKR-equivalences we have: (1) active and passive constructions; (2) *-ed* and *-ing* verb forms in reduced relative clauses are equivalent to passive/active verbal constructions (e.g., the question can be formulated in present tense, active voice *What causes hepatitis A?*, while the answer is obtained from a definitional statement involving the reduced relative clause *hepatitis A is an acute but benign viral hepatitis caused by a virus* . . . ;

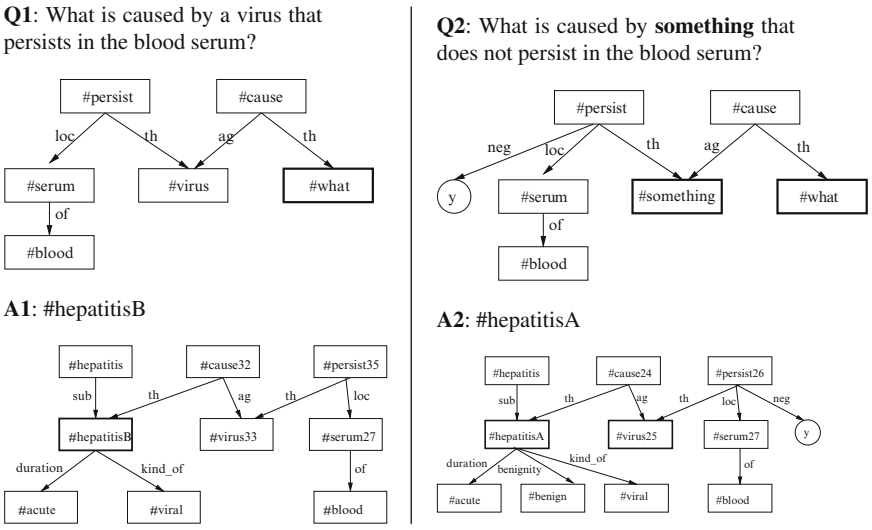

Fig. 10.4 Examples of precise and vague questions, their OKR representations and the concept-level answers

(3) constructions involving raising verbs, where we can take advantage of the fact that the controller is not the semantic argument of the raising verb (e.g., in the definition of Hepatitis B we have *... caused by a virus that tends to persist in the blood serum*, while the question can be asked without the raising verb *What is caused by a virus that persists in the blood serum?*).

A consequence of our weak concept identity principle is that we have structure sharing among OKRs (for example, the OKRs of Hepatitis A and Hepatitis B share the representation corresponding to *blood serum* (#serum27, #blood)), as well as hierarchies of concepts and inheritance.

10.4.2 Natural Language Querying

Besides acquisition of terminological knowledge, our grammar and semantic interpreter facilitates natural language querying of the acquired terminological knowledge by treatment of wh-questions. For this experiment, we created a benchmark of 29 questions. The type of questions we used are "Who did what to whom?", that is only questions regarding the verbs' arguments. Since in our knowledge base we obtained a hierarchy of concepts (an example of hierarchy is given in Fig. 10.3), the questions can be related to this hierarchy: e.g., the question *Which are viral diseases?* has as answer #hepatitisA and #hepatitisB, even if their direct parent is #hepatitis and not #disease. Since OKR is a direct acyclic graph, the natural language querying is reduced to a graph matching problem. A question is a subgraph

of the utterance graph where the wh-word substitutes the answer concept. An answer is a vertex in the OKR of an utterance, together with all the edges incident from/to it. We have experimented both with precise and vague questions. An example of a vague question is *What is caused by something that does not persist in the blood serum?*, where *something* is considered as a variable concept that will match a vertex in the OKR. We obtain precise answers at the concept level (see example in Fig. 10.4). A practical advantage of being able to handle vague questions is that we can obtain all the concepts that are in a particular relation with other concepts, or that have particular properties. For questions we have an average of 6.06 semantic representations per question without Φ_i validation. After semantic validation, we have an average of 2.35 semantic representations per question. In this experiment though, even if the lightweight ontology is not always enough to eliminate incorrect semantic representations of questions, we only obtain the correct answer(s), since we match the OKRs of these questions against the manually validated knowledge base.

10.5 Ambiguity Handling

The method for mapping text to knowledge introduced in this chapter relies on a general grammar learning framework and a task-specific semantic interpreter. Learning is done based on annotated examples that do not contain ontology-specific roles or concepts, and thus our learning framework is general. We can use any ontology, depending on the application.

Since in our experiment we only used a lightweight ontology containing only the admissibility relations that we can find at the level of lexical entries, our qualitative evaluation seems to support the intuition that "a lexicon can sometimes be the basis for the development of a practical ontology" [12]. However, while the knowledge we obtained (OKR) does have properties such as structure sharing, inheritance, hierarchies of concepts, relations among concepts, we can not claim at this point that this knowledge is an actual ontology, which will imply a deeper level of formalization, and also application of a strong concept identity principle dealing with synonymy and anaphora. Since we focus on terminological knowledge, modals and negation are important, while temporal reasoning is not. However, if we would not filter tense and aspect, the semantic interpreter could be further developed toward temporal reasoning needed for factual knowledge bases.

An important aspect that needs further discussion is ambiguity. Natural language utterances in isolation could be highly ambiguous. We can have many representations (OntoSeRs/TKRs/OKRs) corresponding to the same utterance. In this case, the robust parser provides all alternatives. Let us consider the classical example:

(1) a. I saw the man with the telescope.

From Fig. 10.5 we can see that this utterance has two OntoSeRs and two ontology level representations (OKRs). This is possible since there are two grammar rules

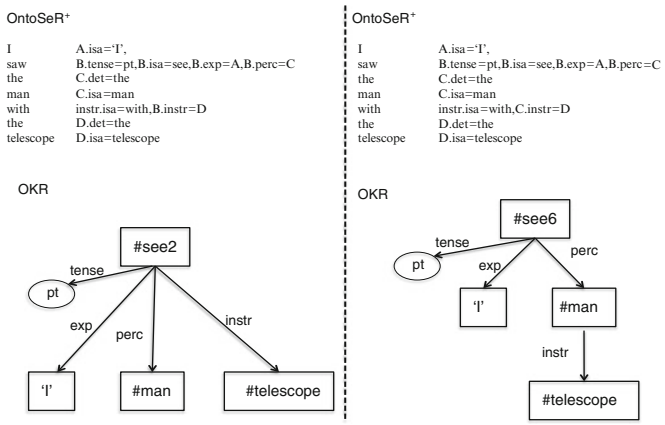

Fig. 10.5 Two OKRs for *I saw the man with the telescope*

from which this utterance can be derived, and the compositional constraints and the ontology constraints satisfy both alternatives. The ambiguity can be eliminated in this case only if we have discourse context, which will be handled by the global semantic interpreter. In this case, we would have two OntoSeRs and TKRs but only one OKR representation, since the global interpreter, which considers discourse context, will be able to remove the erroneous interpretation.

However there are cases where ambiguities can be eliminated by the use of grammar constraints, providing linguistic or semantic context:

(2) a. The two endocrine glands [located above the kidney] [that secrete hormones and epinephrine]

b. I saw the man with the blue shirt.

In the first example the second relative clause can be attached to the noun *kidney* or the noun *glands*. Since using LWFGs we can model agreement between the head noun and the verb in the relative clause, we have that the relative clause is attached to the noun *glands* (plural). This is achieved through the compositional constraints Φ_c. In the second example, the ambiguity can be eliminated through semantic interpretation given a heavyweight ontology with hierarchies of concepts and roles, as well as selectional restrictions. This way, the Φ_i constraint, based on this strong semantic context, allows only one interpretation: the prepositional phrase *with the blue shirt* is associated with the noun *man* and not with the verb *saw*.

Besides moving from a lightweight to a heavyweight ontology as semantic model, another step of our future work is to investigate the use of probabilistic ontologies. In our current work, the semantic interpretation Φ_i acts as a hard constraint. However, constructions in language are more or less likely to appear in a certain context, and thus our semantic interpretation constraints Φ_i should be soft constraints, rather than hard constraints. An important aspect of our future work lies in extending the ontology-level knowledge representation (OKR) to a weighted

representation, and extending the parser to work with both hard and soft constraints. Several alternative could be explored. For example, Markov Logic (ML) [6] is a probabilistic extension of first-order logic, and its strength is rooted in the ability to combine soft and hard first-order formulae. The choice of Markov Logic is further supported by the very recent work of Poon and Domingos (2010) on using Markov Logic for induction of ontologies from text (especially IS-A relations).

10.6 Conclusions

In this chapter we have presented an ontology-based semantic interpreter that is linked to a grammar through grammar rules constraints, providing access to meaning during language processing. In a pilot experiment, we showed that the interpreter could be used to acquire terminological knowledge and to query the knowledge using natural language questions, obtaining precise answers at the concept level. We also showed that even with a lightweight ontology as semantic model, the semantic interpreter is useful to remove some of erroneous utterance parses obtained when we do not have access to meaning. In future work, we plan to use a heavyweight ontology as semantic model, as well as to enhance the ontology with weights/probabilities.

Acknowledgements The author acknowledges the support of the National Science Foundation (IIS-1065195). The author thanks the anonymous reviewers for their feedback. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author, and do not necessarily reflect the views of the funding organization.

References

1. Basili, R., Hansen, D.H., Paggio, P., Pazienza, M.T., Zanzotto, F.: Ontological resources and question answering. In: Workshop on Pragmatics of Question Answering, Held Jointly with NAACL 2004, Boston (2004)
2. Beale, S., Lavoie, B., McShane, M., Nirenburg, S., Korelsky, T.: Question answering using ontological semantics. In: ACL 2004: Second Workshop on Text Meaning and Interpretation, Barcelona (2004)
3. Bresnan, J.: Lexical-Functional Syntax. Blackwell, Oxford (2001)
4. Charniak, E.: A maximum-entropy-inspired parser. In: Proceedings of the first conference on North American chapter of the Association for Computational Linguistics (NAACL-2000), Seattle (2000)
5. Collins, M.: Head-driven statistical models for natural language parsing. Ph.D. thesis, University of Pennsylvania (1999)
6. Domingos, P., Richardson, M.: Markov logic: a unifying framework for statistical relational learning. In: Getoor, L., Taskar, B. (eds.) Introduction to Statistical Relational Learning, pp. 339–371. MIT, Cambridge (2007)
7. Dorr, B.J.: Large-scale dictionary construction for foreign language tutoring and interlingual machine translation. Mach. Trans. **12**(4), 271–322 (1997)

8. Dzeroski, S.: Inductive logic programming in a nutshell. In: Getoor, L., Taskar, B. (eds.) Introduction to Statistical Relational Learning. MIT, Cambridge (2007)
9. Freivalds, R., Kinber, E.B., Wiehagen, R.: On the power of inductive inference from good examples. Theor. Comput. Sci. **110**(1), 131–144 (1993)
10. Ge, R., Mooney, R.J.: A statistical semantic parser that integrates syntax and semantics. In: Proceedings of CoNLL-2005, Ann Arbor (2005)
11. He, Y., Young, S.: Spoken language understanding using the hidden vector state model. Speech Commun. **48**(3–4), 262–275 (2006). Special issue on spoken language understanding in conversational systems
12. Hirst, G.: Ontology and the lexicon. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies in Information Systems. Springer, Berlin (2003)
13. Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., Weischedel, R.: Ontonotes: the 90 % solution. In: Proceedings of HLT-NAACL 2006, New York (2006)
14. Jensen, P.A., Nilsson, J.F.: Ontology-based semantics of prepositions. In: Proceedings of ACL-SIGSEM Workshop: The Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications, Toulouse (2003)
15. Joshi, A., Schabes, Y.: Tree-adjoining grammars. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 3, chap. 2, pp. 69–124. Springer, Berlin/New York (1997)
16. Kaplan, R., Bresnan, J.: Lexical-functional grammar: a formal system for grammatical representation. In: Bresnan, J. (ed.) The Mental Representation of Grammatical Relations, pp. 173–281. MIT, Cambridge (1982)
17. Klavans, J., Muresan, S.: Evaluation of DEFINDER: a system to mine definitions from consumer-oriented medical text. In: Proceedings of The First ACM+IEEE Joint Conference on Digital Libraries, Roanoke (2001)
18. Kowalski, R.A.: Logic for Problem Solving. North-Holland, Amsterdam (1979)
19. Miller, G.: WordNet: an on-line lexical database. J. Lexicogr. **3**(4), 235–312 (1990)
20. Muresan, S.: Learning constraint-based grammars from representative examples: theory and applications. Tech. rep., Ph.D. Thesis, Columbia University (2006)
21. Muresan, S.: Learning to map text to graph-based meaning representations via grammar induction. In: Coling 2008: Proceedings of the 3rd Textgraphs Workshop on Graph-Based Algorithms for Natural Language Processing, Manchester, pp. 9–16 (2008)
22. Muresan, S.: A learnable constraint-based grammar formalism. In: Proceedings of COLING, Beijing (2010)
23. Muresan, S.: Learning for deep language understanding. In: Proceedings of IJCAI-11, Barcelona (2011)
24. Muresan, S., Klavans, J.L.: A method for automatically building and evaluating dictionary resources. In: Proceedings of the Language Resources and Evaluation Conference (LREC-2002), Las Palmas (2002)
25. Muresan, S., Rambow, O.: Grammar approximation by representative sublanguage: a new model for language learning. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL), Prague (2007)
26. Nirenburg, S., Raskin, V.: Ontological Semantics. MIT, Cambridge (2004)
27. Pereira, F.C., Warren, D.H.: Definite Clause Grammars for language analysis. Artif. Intell. **13**, 231–278 (1980)
28. Pollard, C., Sag, I.: Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago (1994)
29. Poon, H., Domingos, P.: Unsupervised semantic parsing. In: Proceedings of EMNLP'09, Singapore (2009)
30. Poon, H., Domingos, P.: Unsupervised ontology induction from text. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10), pp. 296–305. Association for Computational Linguistics, Stroudsburg, PA, USA (2010)
31. Saraswat, V.: Concurrent constraint programming languages. Ph.D. thesis, Department of Computer Science, Carnegie Mellon University (1989)

32. Shieber, S.: The problem of logical-form equivalence. Comput. Linguist. **19**(1), 179–190 (1994)
33. Shieber, S., Uszkoreit, H., Pereira, F., Robinson, J., Tyson, M.: The formalism and implementation of PATR-II. In: Grosz, B.J., Stickel, M. (eds.) Research on Interactive Acquisition and Use of Knowledge, pp. 39–79. SRI International, Menlo Park (1983)
34. Sowa, J.F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks Cole Publishing, Pacific Grove (1999)
35. Steedman, M.: Surface Structure and Interpretation. MIT, Cambridge (1996)
36. Wong, Y.W., Mooney, R.: Learning synchronous grammars for semantic parsing with lambda calculus. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL-2007), Prague (2007)
37. Zettlemoyer, L.S., Collins, M.: Learning to map sentences to logical form: structured classification with probabilistic categorial grammars. In: Proceedings of UAI-05, Edinburgh (2005)
38. Zettlemoyer, L.S., Collins, M.: Learning context-dependent mappings from sentences to logical form. In: Proceedings of the Association for Computational Linguistics (ACL'09), Singapore (2009)

Chapter 11
How Ontology Based Information Retrieval Systems May Benefit from Lexical Text Analysis

Sylvie Ranwez, Benjamin Duthil, Mohameth François Sy, Jacky Montmain, Patrick Augereau, and Vincent Ranwez

Abstract The exponential growth of available electronic data is almost useless without efficient tools to retrieve the right information at the right time. This is especially crucial in the context of decision making (e.g. for politicians), innovative development (e.g. for scientists and industrials) or economic development (e.g. for market or concurrence studies). It is now widely acknowledged that information retrieval systems (IRS in short) need to take semantics into account. In this context, semantic Web technologies have been rapidly widespread and accepted. This article surveys semantic based methodologies designed to efficiently retrieve and exploit information. Some of them, based on terminologies, are fitted to open context, dealing with heterogeneous and unstructured data, while others, based on taxonomies or ontologies, are semantically richer but require formal knowledge representation of the studied domain. Hence, a continuum of solutions exists from terminology to ontology based IRSs. These approaches are often seen as concurrent and exclusive, but this chapter asserts that their advantages may be efficiently combined in a hybrid solution built upon domain ontology. The original approach presented here benefits from both lexical and ontological document description, and combines them in a software architecture dedicated to information retrieval in specific domains. Relevant documents are first identified via their conceptual indexing

S. Ranwez (✉) · B. Duthil · M. François Sy · J. Montmain
LGI2P Research Center from Ecole des Mines d'Alès, Parc scientifique G. Besse, F-30035, Nîmes Cedex 1, France
e-mail: Sylvie.Ranwez@mines-ales.fr; Benjamin.Duthil@mines-ales.fr; Mohameth.Sy@mines-ales.fr; Jacky.Montmain@mines-ales.fr

V. Ranwez
SupAgro Montpellier (UMR AGAP), 2 place Pierre Viala, F-34060, Montpellier Cedex 1, France
e-mail: ranwez@supagro.inra.fr

P. Augereau
IRCM, Institut de Recherche en Cancérologie de Montpellier Inserm U896 and Université Montpellier 1, CRLC Val d'Aurelle Paul Lamarque, F-34298, Montpellier, France
e-mail: patrick.augereau@inserm.fr

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_11,

based on domain ontology, and then each document is segmented to highlight text fragments that deal with users' information needs.The system thus specifies why these documents have been chosen and facilitates end-user information gathering.

11.1 Introduction

The exponential growth of available electronic data is almost useless without efficient tools to retrieve the right information at the right time. This is especially crucial with respect to decision making (e.g. for politicians), innovative development (e.g. for scientists and industrial stakeholders) and economic development (e.g. for market or competitive analysis). It is now widely acknowledged that information retrieval systems (IRSs in short) need to take semantics into account to enhance the use of available information. However, there is still a gap between the amounts of relevant information that can be accessed through optimized IRSs on the one hand, and users' ability to grasp and process a handful of relevant data at once on the other. Even though Semantic Web technologies and ontologies are now widespread and entrenched, they are hampered by the fact that they cover few aspects that a document deals with – this is known as the semantic gap issue. They should thus be jointly used with terminological or lexical approaches to enrich document description.

This chapter starts with a survey on semantic based methodologies designed to efficiently retrieve and exploit information. Hybrid approaches including lexical analysis are then discussed. Terminology based lexical approaches are tailored to open contexts to deal with heterogeneous and unstructured data, while other taxonomy or ontology based approaches, are semantically richer but require formal knowledge representation of the studied domain and conceptual indexing. While these latter are often implemented at the document level, automatic terminology indexing allows fine-grained descriptions at the sentence level. Hence, there is a continuum of solutions from terminology to ontology based IRSs. These approaches are often seen as concurrent and exclusive, but this chapter asserts that their advantages may be efficiently combined in a hybrid solution built upon domain ontology. The original approach presented here benefits from both lexical and ontological document description, and combines them in a software architecture dedicated to information retrieval and rendering in specific domains. Relevant documents are first identified via their conceptual indexing based on domain ontology, and then segmented to highlight text fragments that deal with users' information needs. The system thus specifies why these documents have been chosen and facilitates end-user information gathering.

Section 11.2 presents related works, introduces information retrieval main layers and distinguishes conceptual and keyword-based strategies. However some limits of those two IRS categories are raised, that justify hybrid approaches. Therefore such approaches that involve ontology enrichment with lexical information as well as text segmentation are depicted. The underlying generic architecture, called *CoLexIR* (Conceptual and Lexical Information Retrieval), uses an ontology and lexical resources interfacing strategy, as summarized in Fig. 11.1. The next sections

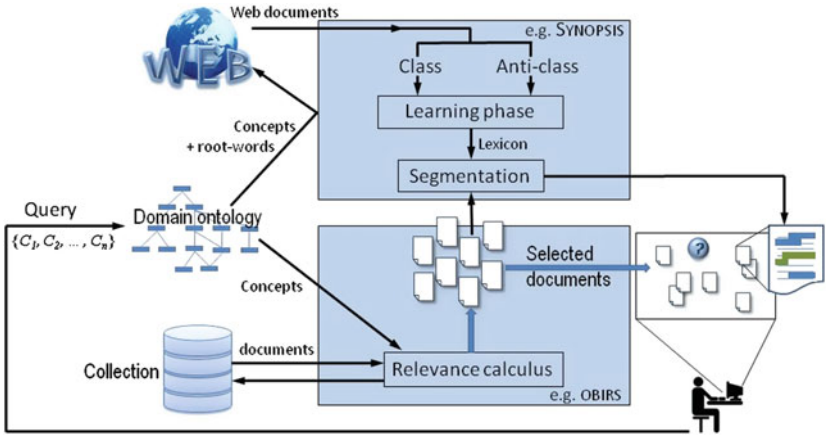

Fig. 11.1 Overview of our CoLexIR approach

are dedicated to this hybrid approach. Section 11.3 details the different phases of text segmentation that are implemented within the *Synopsis*approach [17]. Section 11.4 presents the *CoLexIR* system that supplements *OBIRS*, an ontological based information retrieval system [53], with ontology lexical component. The pros and cons are discussed, particularly the complementarity of both approaches is underlined and we show how their limits may be overcome by this combination. An evaluation of the *CoLexIR* environment through a case study is proposed in Sect. 11.5. It involves expert evaluations to assess the relevance and man-machine interactions in our system, using a set of BMC cancer publications as corpus. This latter and its indexing by medical subject headings (MeSH[1]) concepts are freely accessible via PubMed (biomedical literature from the National Center for Biotechnology Information). These tests focus especially on document previews that facilitate and speed up bibliographic research by pinpointing relevant sentences from relevant documents. Some perspectives are finally given, particularly concerning the possibility of automatic indexing approaches in closed contexts (data warehouse containing similar documents indexed using ontological concepts) and their extension to open contexts (the Web containing heterogeneous and poorly indexed documents).

11.2 Related Work

The main task of an information retrieval system (IRS) is to select information which is likely to meet user needs, expressed as queries. Three processes are usually implemented in IRSs to fulfill this task [35]: (i) an indexing process which aims to

[1] http://www.ncbi.nlm.nih.gov/mesh

provide a representation that is as compact and expressive as possible of resources (textual, multimedia documents) and queries; (ii) a matching process for selecting relevant resources w.r.t. to a query; (iii) a query reformulation process that typically occurs between the two previous points.

Document and query indexing models (singleton or complex structure) and query-document matching strategies (strongly dependent on the indexing model) are generally sufficient to characterize and identify information retrieval models as Boolean, vectorial or probabilistic ones. Among IRS processes, indexing plays a key role because it provides content description of resources, allowing search tools to match them with user queries. Depending on the indexing methods, IRSs are historically classified in two categories [22]: keyword-based IRSs, also called *syntactic* search systems, and conceptual IRSs, known as *semantic* search systems.

11.2.1 Conceptual Versus Keyword-Based IRSs

Keyword-based IRSs often represent documents and queries as a bag-of-weighted-words or multiwords (phrase). This representation, obtained through document lexical analysis, summarizes document contents by a set of key terms [18]. A keyword-based IRS relevance process may rely on an exact match, an approximate match, or a string distance between words within documents and query indexing. Hence, when a query is submitted, these systems will retrieve documents indexed by exact query keywords or some of their lexical variations (e.g. *tumorous* instead of *tumor*). Unfortunately, they are hampered by the so-called synonymy problem and miss documents having query keyword synonyms in their indexing (e.g. *carcinoma* instead of *tumor*) [5, 21, 22]. Keyword-based IRSs also fail to consider various kinds of semantic relationship between words (hyponyms, hypernyms) as well as polysemous problems (e.g. *cancer* as astrological sign or as illness) due to language ambiguity [4, 21]. All of these issues account for the lack of precision of keyword-based information retrieval systems, which is a well known problem [51].

To overcome these limitations, conceptual resources have been used to represent document contents based on their meaning rather than on their words. These conceptual resources may be arranged from less formal ones (thesaurus with strong lexical compounds: WordNet or UMLS) to more formal ones (e.g. Gene Ontology), and from general to domain specific. In any case, manual or automatic extraction techniques [56] are needed to use such term meanings or concepts for indexing purposes. But this is beyond the scope of this chapter.

Conceptual IRSs are based on the assumption that document contents are better described by conceptual abstractions of real word entities than by lexical relationships that may be found within it or dictionaries [4, 15]. The emergence of domain ontologies, boosted by the development of the Semantic Web (in its infrastructure and content), has led to an increase in conceptual IRSs. In these systems, ontology based concepts are used as pivot language for indexing documents and expressing queries. Such conceptual description of the world may

also be used as a semantic guideline while visualizing documents or data. Besides, ontologies provide a conceptual space in which metrics (semantic similarities or distances) can be deployed to implement the relevance calculus process in IRSs. According to [50], a domain ontology O can be formally defined as follows:

Definition 11.1. $O := \{C, R, H_C, H_R, Rel, A\}$, where C and R are respectively a set of concepts and a set of non-taxonomic relations. H_C is a heterarchy of concepts with multiple inheritance. H_R is a heterarchy of relations. $Rel : R \longrightarrow C \times C$ defines non-taxonomic relations between concepts, while A is a set of logical axioms.

Complete conceptual indexing is hard to achieve in realistic collections. Indeed, domain ontologies may be hampered by weak coverage of some subject matters addressed in those documents, because ontologies do not focus on those aspects and hence do not model them [5]. Also, high quality indexing requires human expertise and is thus a tedious task. This is known as the *semantic gap* issue. In the same way, automatic or semi-automatic indexing techniques cannot always extract all significant document information. For example, information extraction tools perform well on some tasks such as Name Entity Recognition, fact or relation recognition but poorly on complex tasks such as event extraction [14]. In order to increase the ontology coverage and improve both document and user query indexing within conceptual based IRSs, lexical components can be added to the ontology, as detailed in following section.

11.2.2 Hybrid Ontology Based Information Retrieval System

Hybrid IRSs have been designed to take both keyword based and conceptual based indexing units into account. We propose the following definition for a hybrid ontology based information retrieval system:

Definition 11.2. An ontology based information retrieval system is called *hybrid* when it manages document indexes of different semantic granularities (ontology based and keyword based) at different text levels (whole document and passages) during indexing and matching processes and/or during the result rendering stage.

- *Document indexing at different semantic granularities:* ontology based and keyword based document indexing may coexist within realistic collections. Indeed, it may happen that the indexing process failed in attaching some document keywords to concepts from the used ontology. In this case, information retrieval relevance models need to consider both kinds of indexing to prevent the possible loss of information. This leads to hybrid relevance models, which are discussed below.
- *Document indexing of different text levels:* indexing units in both keyword based and ontology based IRSs may be related to the whole document (*document level*)

or to some of its parts (*passage level*). A passage is not necessarily a paragraph within a document but any continuous subset (portion) of texts. Characterization of the document parts allows passage retrieval, which is suitable for multi-topic documents. In this case, passages are considered independently, indexed with concepts or keywords, and treated as documents. But characterization of document parts may also be used to give the user some justification of the document selection. When the concepts occurring in different document passages are known, it is possible to segment texts and highlight passages that deal with user query concepts. This provides most users with insight into the results as shown by Hersh [24] in the biomedical domain and by Lin et al. [33] in the Architecture/Engineering/Construction domain. This user interaction improvement will be discussed below.

11.2.2.1 Hybrid Relevance Model

Although hybrid relevance models encompass both conceptual and keyword based models through different semantic granularity indexes, they are considered separately during document relevance (RSV) evaluation. Indeed, a document keyword is used as an indexing unit only when information extraction tools have failed in connecting it to a concept within the ontology. Most hybrid IRSs consider these two kinds of indexing unit as covering disjoined aspects of the document content and thus propose relevance models using two separate kinds of document/query suitability assessment: conceptual/semantic based and keyword based. A merged strategy of these two outputs is then applied. Three kinds of query are thus possible in such hybrid relevance models:

- Fully semantic or conceptual queries (using only ontology concepts and relations);
- Only keyword queries (no semantic description of documents is available);
- Mixed queries (both keyword and conceptual queries are available).

Many hybrid relevance models have been proposed in the literature. *K-search* [5] combines ontology based and keyword based search to support document retrieval (*ad hoc* retrieval) and knowledge retrieval through Resource Description Framework (RDF) triples search using Sparql. Having two kinds of document descriptions (RDF triples with a link to their resources and keyword indexes), the authors define a hybrid relevance model as the combination of a keyword based model (e.g. using *Lucene*[2]) and a semantic model (like *Sesame*[3]) used independently. Keyword searches return a set $\triangle$ of documents. Semantic searches return a list of RDF triples associated with the documents they come from, and thus implicitly define the set

[2] http://lucene.apache.org/core/

[3] http://www.openrdf.org/

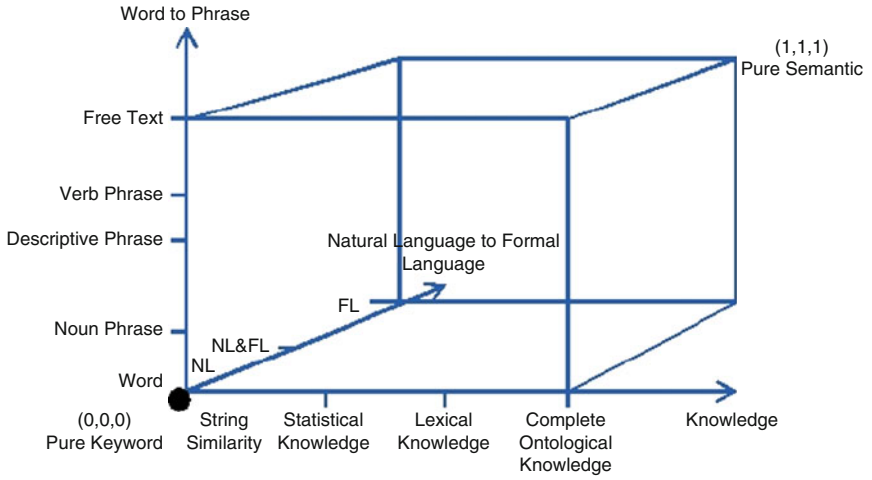

Fig. 11.2 Semantic continuum: the classification of hybrid IRSs as proposed by Giunchiglia et al. [21]

$\triangle^{rdf}$ of these documents. The overall *K-search* results consist of the intersection of $\triangle$ and $\triangle^{rdf}$.

Giunchiglia et al. [21] extends keyword search and proposes to classify hybrid IRS according to three dimensions in a Cartesian space that accounts for the so called *semantic continuum* (see Fig. 11.2). Each dimension supplements the keyword search using semantic search when possible. Note that the scales of the three axes in Fig. 11.2 are not ordinal but the semantic complexity increases as the distance from the origin increases. The *Natural Language to Formal Language* axis goes from natural language to formal language in order to solve keyword search polysemous and synonymy problems (word to concept). Considering this axis, a 0 coordinate means that the IRS considers words as indexing units whereas 1 means that conceptual units are used. When systems move through this axis, some words may not be mapped to a concept due to a lack of background knowledge (weak coverage: semantic gap). Giunchiglia et al. [21] proposes to use both syntactic and semantic retrieval to overcome this drawback. This axis deals clearly with indexing semantic granularity. The *Word to Phrase* second axis ranges from words to phrases to overcome complex concept expression. This dimension deals with the indexing structure. A 0 value on this axis corresponds to single indexing units (word or concept) while 1 represents complex ones. The *Knowledge* axis goes from string similarity to semantic similarity to achieve a relatedness estimation of indexing units.

Organizing hybrid IRSs in such a 3D Cartesian space provides a simple and relatively intuitive characterization of these IRSs but is hampered by some limitations. Indeed, three dimensions are insufficient to fully describe all kinds of indexing process implemented in IRSs, the complexity of their relevance calculus and user interaction. Moreover, there is no proof of the independence of the chosen dimensions. Finally, a linear axis is not sufficient to represent a complex index

structure since the information on how units are linked and organized is not taken into account.

11.2.2.2 Hybrid Approach for User Interaction Improvement

Information retrieval is often an iterative process where the user refines his/her query to dig out highly relevant documents. But this process implies that the end-user has a precise understanding of the results proposed by the search engine and that interaction techniques allow him/her to reformulate the query, select interesting documents and give some hints to the system about his/her preferences. Visualization techniques may thus be considered as key components of this process since they play a mediating role in this understanding. There are many specifications that characterize IRS result visualization interfaces but two of them are of particular interest:

- Cognitive aspects. In consideration of users' cognitive limits, it is important to enable users: to identify relevant documents at a glance, to focus on a specific section of the visualization interface, and to intuitively understand any action results [54].
- Colors. Visual color scanning requires less time than visualizing words [13]. Colors may highlight some semantics and reflect the relative importance of displayed elements: e.g. green may denote the presence of a query term in a document indexing unit, while blue may indicate the presence of another more generic related term.

Dimensionality is also an important feature of IRS visualization interfaces. However, most IRSs display results in a 2D space.

The simplest and most common way to display query results is in a list, where each item includes the retrieved document's title and its snipet with query terms highlighted (concept by label identification or words) in the document context. However, this type of presentation does not meet the above requirements. When passage level description is available, hybrid IRSs are able to show result explanations at the text level, thus synthesizing relevant information by highlighting relevant passages. Many such systems propose a range of result displays, from traditional document lists to passage visualization [33]. In *K-search* [5], retrieved documents are displayed in a list and document details are available in a separate panel when one of them is selected: keywords and RDF triples are thus highlighted. *K-search* also allows summarization of results using bi-dimensional graphs where different variables (e.g. retrieved document location) can be plotted. This graph is used to filter results. With the *Ontopassage* search engine [33], long and multitopic documents are fragmented as sets of passages and used as collection units. These passages are indexed using an ontology constructed from domain resources (e.g. relevant technical books of a domain). The system allows users to implement different relevance models in the same query session (vector space or probabilistic model). Users can switch from a traditional display mode (list of

retrieved documents) to a passage display mode. In the latter, most relevant passages (w.r.t. the user query) of each retrieved document are displayed. A small concept hierarchy is also displayed for each document, allowing users to explore related query concepts.

Hence, in the *CoLexIR* approach, we solely use a semantic relevance assessment model such as the one implemented in OBIRS and detailed in Sect. 11.4.1. The relevances of retrieved documents using different IR models are not comparable. Therefore, we consider that allowing users to switch between these different models introduces confusion with IR visualization interfaces.

11.2.2.3 Ontology and Lexical Resources Interfacing

The ontology has to be supplemented with lexical resources so as to be able to identify document passages that are related to domain ontology concepts. Most domain ontology construction methods do not hold lexical information from which its concepts are taken. This issue is known as the missing link between ontology and lexical layers [2]. Indeed, the formalisms used to represent an ontology, such as OWL, mainly focus on the intrinsic description of concepts, property classes and logical constraints on them and many initiatives[4] have been conducted to go beyond the label systems they implement. In order to deal with an ontology lexical component, those initiatives mainly rely on its representation models [2, 6, 11] and interfacing techniques to build it [52].

Prévot et al. [41] distinguishes three different approaches for those interfacing techniques. The first one aims at structuring lexical resources using ontological principles without ontological category or relation. The second uses lexical information to enrich an ontology by adding lexical entries to the ontology (populating) [39] or by adding lexical information to concepts. Adding lexical entries to an ontology may increase the ontology size and coverage, whereas enriching ontology concepts with lexical information does not change the ontology structure even if the coverage is increased. The last way of interfacing ontology and lexical resources combines the two previous approaches. Staab and Maedche [50] provides a definition of a lexical component of an ontology O:

Definition 11.3. A lexical component L of an ontology O is defined as: $L := \{L_C, L_R, F, G\}$ where L_C, L_R are disjoint sets of lexical entries respectively related to concepts and relations; F (resp.G) provides correspondence between concepts (resp. relations) and their lexical entries.

In the *CoLexIR* approach, we attach lexical information to ontology concepts and use such lexical information to determine passages that deal with each query concept in the returned documents. Our enrichment methodology therefore does not change the ontology structure and thus refers to the *enriching* option that has been

[4]Ontology-Lexica Community Group: http://www.w3.org/community/ontolex/

previously described as "enriching ontology concepts with lexical information". We propose an unsupervised method to build lexicons related to each ontology concept. The lexicons that are built must be relevant for both general and specific ontologies. A related issue is that the vocabulary associated with a concept depends on the level of expertise of the person who performs the term-concept matching. In other words, our approach should provide several granularity levels for the description of a same concept, suited to those different levels of expertise.

The next subsection details the basic notions of text segmentation, particularly to identify parts which deal with a specific concept in a document. Our approach is closely connected to the text partitioning process and thematic extraction process.

11.2.3 Concept Identification Through Lexical Analysis

A text partitioning process is based on the analysis of thematic breakdowns in a document in order to subdivide the document into semantically homogeneous parts. These parts are considered as "text portions" (passages) which have very strong semantic coherence and are clearly disconnected from adjacent parts [47]. Thematic text segmentation may also be seen as a process of grouping basic units (words, sentences, paragraphs, etc.) in order to highlight local semantic coherence [29]. From a global standpoint, thematic structure search [36, 37] is a first crucial analysis step in many applications such as text segmentation, text summarization, or information retrieval [3].

Among the approaches described in the literature, two categories may be distinguished:

- *Lexical cohesion based approaches*. Several approaches measure this cohesion via term repetitions, semantic similarity, context vector entity repetition, word frequency models or word distance models. The re-occurrence of specific terms may indicate the presence of a common topic [1, 23, 27]. Lexical chains and their extension, the so-called weighted lexical links approach, are two identification techniques often used in a huge collection. The topic unigram language model is the most frequently used technique [40]. Most lexical cohesion based techniques are linear topic segmentation algorithms. These algorithms set boundaries inside a text at positions where a topic shift is identified. This process is performed in a (fixed size) sliding window. Lexical variation often results in dropping an employed similarity measure. Many methods use this process: TextTiling [23], C99 [8], Dotplotting [45], and Segmenter [27].
  There are also other statistical approaches that use the overall information in the text [26]. Text segmentation is based on analysis of the whole text, contrary to lexical cohesion based approaches that analyze a text on the fly. Malioutov [34] presents a graph-theoretic framework. The text is converted into a weighted undirected graph in which the nodes represent sentences and the edges quantify thematic relations between them. Text segmentation is performed by maximizing

the similarity within each partition and minimizing dissimilarity across the partition [49]. Lamprier et al. [31] offers a statistical linear segmentation based on genetic algorithms.

- *Natural language processing techniques*. Linguistic methods introduce a set of specific rules that link words to each other (e.g. N-grams). These rules are dependent on the corpus. Linguistic methods still use external semantic information resources such as thesauri and ontologies. Information resulting from the association rules and from external semantic sources may then be combined through statistical techniques [38], which are highly dependent on available resources. Caillet proposes an automatic segmentation method based on term clustering [7]. This approach discovers the different themes in a text and extracts their related representative terms. Clifton et al. [12] proposes an algorithm to recombine segments according to their content.

Note that segmentation approaches all have the same weakness: they do not allow precise identification of the themes (labeling) of a text portion, they only detect semantic breaks in a text without providing labels. To solve this labeling issue, some studies, based on text summary [20] and key phrase extraction approaches [25], identify text portions or key phrases according to their major theme [10]. Other methods focus on the identification of text portions related to the document title [30]. Most automatic text summary methods are based on a supervised learning process, that requires human intervention to set an adequate training corpus [9, 55]. Riedhammer et al. [46] proposes an unsupervised method to extract key phrases in a summarization context.

Similar to segmentation methods, the approach presented in the following uses statistical information to identify, in a non-supervised context, text portions related to a given concept. The next section proposes an implementation of the F correspondence function (Definition 11.3), thereby producing a lexicon for concepts and a thematic extraction process.

11.3 Concept Identification Through Lexical Analysis: The "Synopsis" Approach

The aim is to automatically identify document passages that are related to a given concept. This section describes an adaptation of the *Synopsis* approach [17] involving tagging of text items according to predefined concepts (e.g. those expressed in the user query). For each concept, the *Synopsis* process starts by building a lexicon L containing a set of words that characterize it and a set of words that do not. This is performed by processing a significant number of documents that are downloaded through a Web search engine (e.g. *Google*). Then, based on the learned lexicon, *Synopsis* identifies text portions according to the given concepts.

This section describes the two main phases of this process: (i) generation of the learning dataset and elaboration of concept lexicons (Sect. 11.3.1) and (ii) extraction

of topics related to the concepts from textual data (Sect. 11.3.2). Note that the first step is time consuming and has been preprocessed once and for all, while the second step is fast enough to be done on the fly on retrieved documents.

As our hybrid approach evaluation (see Sect. 11.5) relies on *Cancer* related scientific publications, some vocabulary in this domain will be used hereafter to illustrate our approach. The scientific publications are indexed by the *MeSH* ontology concepts.

11.3.1 Concept Characterization

As a start, lexicons related to some concepts in a domain have to be built. There are four steps in this process: acquisition of relevant corpus for each concept, significant words learning, representativity calculus for each of these words and lexicon elaboration.

11.3.1.1 Acquisition of Relevant Corpus

The first objective is to automatically build a training corpus for each concept of interest in a specific domain. For our purposes, these concepts are those in the *MeSH* ontology and the domain is *cancer*. A set of *root-words* (also called *germs*) has to be attached to each concept. Here we rely on the *MeSH* ontology to automatically obtain n root-words, which are the label of the concept of interest and those of its hyponyms. For example, regarding the "dna" concept, the following root-words may be identified thanks to its label and its hyponym ones: "dna", "dna, z-form", "dna, satellite", "dna, intergenic", "dna, plant"... For each root-word r related to a concept C, the *Synopsis* system, via a Web search engine, searches for 300 documents that contain both the root-word r and the name of the domain (e.g. "dna, z-form" and "Cancer" in our case). Together, these texts will form the *class* of C.

Similarly, the system searches for 300 documents of the domain that do not include any root-words of the concept C. Together, these texts are called the *anti-class* of C. This set constitutes the second part of the corpus related to C. It obviously improves characterization of the concepts: a domain term that appears frequently in the class as well as in the anti-class for a concept is not discriminating (not representative) for this concept.

The class related to C thus contains $n * 300$ documents (where n is the number of root-words). Its anti-class contains 300 documents. The union of the class and the anti-class of C constitutes the corpus related to C. The second step involves searching any words significantly related to the root-words within these documents.

11.3.1.2 Significant Word Training

First of all, HTML tags, advertising and other noisy contents are removed from the documents of the corpus related to C. These filtered documents are then transformed using a morpho-syntactic analyzer and lemmatization techniques [48]. This step identifies the representative (respectively non-representative) words for C. This is achieved by occurrence frequency analysis, assuming that the probability that a word characterizes a concept is proportional to its occurrence frequency in the immediate neighborhood of one of the concept's root-words. This occurrence frequency is computed over the whole corpus of concept C and is used to quantitatively assess the representativity score Sc of a word W w.r.t. C. At the end of this step, lexicon L related to a concept C is formed with a set of words and their representativity w.r.t. C.

Two categories of words are distinguished: i.e. those prevailing in the class and those prevailing in the anti-class.

More formally, the words in the immediate neighborhood of a concept's root-word r are first selected inside a window $\mathscr{F}$ of size sz in a document doc:

$$\mathscr{F}(r, sz, doc) = \{w \in doc / d_{noun}(r, w) \leq sz\} \tag{11.1}$$

with $d_{noun}(r, w)$ being the distance corresponding to the number of nouns (considered as meaningful words [28]) separating a word w from r in the document doc [17].

11.3.1.3 Representativity of Words

It is now possible, for each word W of the corpus, to define its representativity in the class of the concept C. It is denoted $X(W)$ and defined as the sum of occurrences of a word W in a window $\mathscr{F}(r, sz, d)$ for all the root-words of C and all the documents of the corpus. Note that for the anti-class, there is a single "root-word" which is the domain itself. The representativity in the anti-class is denoted $\overline{X}(W)$.

11.3.1.4 Lexicon Elaboration

From the representativity of a word W in the class and in the anti-class, a score is established for this word using the following discrimination function [17]:

$$Sc(W, sz) = \frac{(X(W) - \overline{X}(W))^3}{(X(W) + \overline{X}(W))^2} \tag{11.2}$$

Fig. 11.3 Example of sliding window $\mathscr{F}'$ (with size = 1). The *dots* between two nouns symbolize the possible presence of any word that is not a common noun

The cubic numerator function allows a signated discrimination: words of the domain that are non-representative of the concept get negative scores, while representative words of the concept get positive scores. The square denominator function allows a normalized score. It is now possible to build a concept-specific lexicon which include all nouns encountered in either the class or the anti-class documents of the concepts with their respective score (either positive or negative).

11.3.2 Thematic Extraction

Finally, this section explains how the achieved lexicon can be used to obtain thematic segmentation of any document. A sliding window $\mathscr{F}'$ is introduced: it is successively centered on each occurrence of nouns in the processed document *doc*.

From lexicon L of a concept C, a score is computed as follows for each sliding window $\mathscr{F}'$ in a document *doc* (c.f. Fig. 11.3):

$$Score_{doc}(\mathscr{F}') = \sum_{W \in \mathscr{F}'} Sc(W, sz) \tag{11.3}$$

In document *doc*, the sliding window $\mathscr{F}'$ is said to be related to a concept C as soon as its score is higher than a predetermined threshold. Roughly speaking, the higher this threshold, the more reliable the matching between the selected sliding windows and the concept C. The number of words that can be linked to the concept C is a function of the threshold value. The number of words slowly evolves with the threshold value, except for some singular values that correspond to rough changes in semantic points of view, i.e. significant breaks in the granularity description (Fig. 11.4). The choice of the threshold can be supported by sensitivity analysis of the function. It finally allows allocation of a lexicon with a parameterized granularity to a concept.

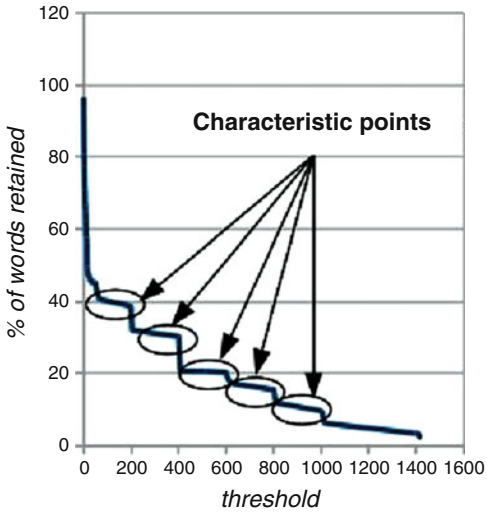

Fig. 11.4 Example of sensitivity analysis. Five discontinuities are observed, they correspond to rough changes of semantic level in the description of a document

11.4 Human Accessibility Enhanced at the Crossroads of Ontology and Lexicology

As previously stated, the *CoLexIR* architecture relies on a concept-based IRS. The one that is used in the following, called *OBIRS*, is detailed here.

11.4.1 An Example of Concept-Based IRS: OBIRS

The use of domain ontology semantics is known to improve IRS effectiveness. Sy et al. [53] proposes an ontological-based information retrieval system (*OBIRS*) using semantic proximities and aggregation operators to assess document adequacy w.r.t a user query. Since *OBIRS* methodological details and validation protocols are available in [53], it is only outlined in this section.

OBIRS allows assisted query formulation based on domain ontology concepts and implements a relevance model using semantic proximities. The proposed relevance score computation (also called retrieval status value [RSV]) consists of three stages of the aggregation process:

- The first stage computes a similarity measure (denoted π) between two concepts of the ontology O. Several semantic proximity measures may be used here, that can be based on calculation of the shortest path, on use of the information content (IC) [32, 44] or on set based measures [43]. In order to favor user interactions, concept proximities must be intuitive (so that the end-user can easily interpret them) and fast enough to compute (to ensure that the IRS remains efficient even

in case of large ontologies). By default, *OBIRS* relies on Lin's proximity for this step [32].

- Then a proximity measure is computed between each concept of the query and a document indexing. Let d_i denotes the i^{th} element of the list $C(d)$ of concepts indexing a document d, the similarity between a concept Q_t of a user query Q ($t = 1..|Q|$) and d is defined as:

$$\pi(Q_t, d) = \max_{1 \leq i \leq |C(d)|} \pi(Q_t, d_i) \tag{11.4}$$

- Finally, the relevance score of a document w.r.t a query is assessed using the family of aggregation operators proposed by Yager [16]. Each query concept is considered as a criterion to be satisfied and corpus documents as alternatives. The assessment of such alternatives with regard to the criteria is given by:

$$RSV(Q, d) = \left(\frac{\left(\sum_{t=1}^{|Q|} p_t . \pi(Q_t, d)^q \right)}{|Q|} \right)^{\frac{1}{q}}, q \in \Re, \sum_{t=1}^{|Q|} p_t = 1 \tag{11.5}$$

This aggregation model takes into account the user model preference about the kind of aggregation that has to be introduced to compute the overall relevance of a document w.r.t his/her query. When the above weighted operators' family is used, the user has the opportunity to fit both q parameter and the p_t weight distribution upon the query terms. The weights characterize the relative importance granted to each of the query terms in the preference model, whereas the q parameter sets the extent to which the simultaneous satisfaction of all criteria is required to assign a high RSV score to a document. Indeed, in Eq. 11.5, when q has very small values ($-\infty$) the query tends to be conjunctive (aggregation involves the MIN operator) whereas when q gets close to $+\infty$, the query tends to be disjunctive (aggregation involves the MAX operator). By default, *OBIRS* uses equal query term weights and $q = 2$.

This last stage synthesizes document relevance w.r.t. users' preferences and ranks the collection of retrieved documents according to their RSV. The aggregation model enables restitution of the contribution of each query concept to the overall relevance of a document. Hence it provides our system with explanatory functions that facilitate man-machine interaction and assists end-users in iterating their query.

OBIRS has been implemented and a web-based client is available.[5] Although users want IRSs to return good relevant documents at the top of the result list,

[5] www.ontotoolkit.mines-ales.fr/ObirsClient/

to ensure fast grasp of relevant information, they also need explanations about why documents have been chosen, and indications about the most interesting document passages [24]. *OBIRS and Synopsis* have been combined into the *CoLexIR* hybrid IRS for user interaction improvement according to the definition given in Sect. 11.2.2.

11.4.2 Ontology and Lexical Resource Interfacing Within Hybrid IRSs

In the *CoLexIR* visualization interface, retrieved documents are displayed in a semantic map. The higher their scores, the closer the documents are to the query, which is represented as a probe (symbolized as a question mark). The result explanation focuses on both document and passage levels. Each document is represented on the map by a pictogram which details its match with the query. The contribution of each query concept to the overall score assessment is summed up in a histogram where a bar is associated with each concept Q_t of the query. This bar is colored depending on whether the closest (according to the chosen semantic similarity measure) concept of the document indexing is exactly Q_t (green), a hyponym (red) or a hypernym (blue) of Q_t. The bar is purple in other cases. The height of the bar associated with Q_t is proportional to the elementary score of the document w.r.t. Q_t (i.e. $\pi(Q_t, D)$). A deeper analysis of document relevances is facilitated by their lexical analysis. Passages that deal with each query concept are identified by the segmentation process and highlighted at the text level. Double clicking on a document shows passages related to each query concept. These passages do not necessarily contain any query concept labels but rather terms that have been related to the concept lexicons in the segmentation step. In this way, users may see their query concepts instances within each document and also other concepts that the document deals with and that could be used to refine their information needs (reformulation support). Figure 11.5 shows an overview of the *CoLexIR* visualization interface.

Harvesting the vocabulary attached to a concept may take a while (about 20 min per concept). This learning phase is done once. Lexical supplements are cached as well as document segments related to each concept within their indexing. This approach ensures the responsiveness of the system, and is relevant since both document collection and domain ontology are relatively stable through time and partial update (e.g. collection or ontology size increase) may be done rapidly in background.

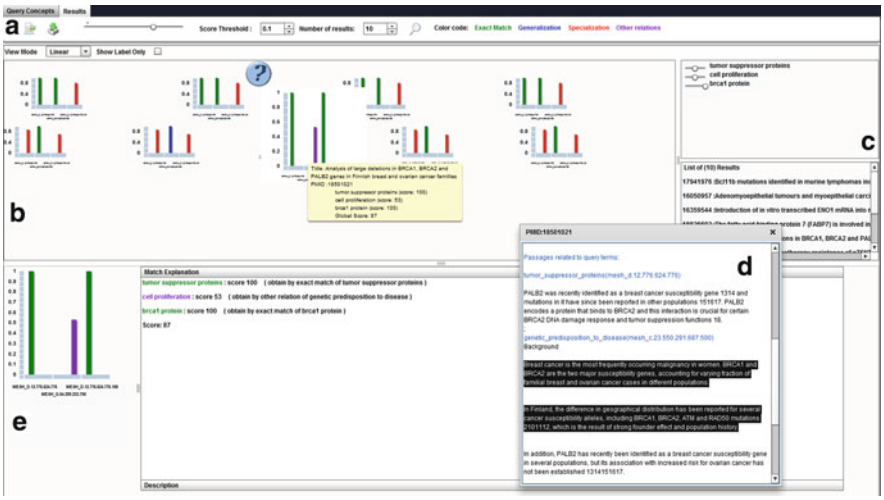

Fig. 11.5 *CoLexIR* interface displays selected document histograms in a semantic map according to their relevance scores w.r.t the query (symbolized by the *question mark*) (*B*). The query concepts and their weights are provided (*C*) as well as query parameters and color code legend (*A*). Match explanation of a document is proposed as well as a link towards the whole document (*E*). Document passages related to the query concepts are available in a pop-up (*D*)

11.5 Evaluation: User Feedback on a Real Case Study

Our system validation implies experts who assess both the relevance and man machine interactions of our system. These tests especially focus on document personalized previews.

Here we describe a biological case study in which the *CoLexIR* system is used to carry out a bibliographical study of proteins that could prevent cell proliferation induced by the BRCA1 protein. A first query of the three MeSH terms "tumor suppressor proteins", "cell proliferation" and "brca1 protein", respectively weighted (100, 100, 100), was submitted to *CoLexIR*. *CoLexIR* detailed scoring of the retrieved documents enabled us to quickly determine that most of these documents did not often deal with the "brca1 protein" MeSH term (low elementary score). A quick scan of *CoLexIR* excerpts of some of these retrieved articles confirmed that our query did not sufficiently stress our specific interest in BRCA1. We thus reformulated our query with adjusted weight, thus using "tumor suppressor proteins" (50) + "cell proliferation" (50) + "brca1 protein" (100). This new formulation generated several relevant papers.

For most of the selected articles, the segmentation process highlighted some relevant pieces of information, w.r.t. query terms, that sometimes did not appear in the title or in the detailed abstract published by BMC Cancer. For example, in [42], the *founder effect* noted in previous studies was not mentioned in the abstract, but retrieved by the segmentation process. The same was true for the

fact that genomic rearrangement between BRCA1 and BRCA2 was not a major determining factor of breast cancer susceptibility in Finland, although this might be useful information for anyone interested in the genomic distribution of BRCA alleles in breast cancers. Similarly, in [19], several key results regarding *leukemia* and *lymphoma* associated genes were retrieved that were absent from the relatively long abstract of an article reporting the role of the BRCA1 gene in non-breast cancer. On the same lines, the excerpt concerning the interaction between BRCA1 and Fanconi proteins was valuable, and could provide researchers working in breast and immunological cancer fields with an opportunity to look for this interaction in either cancer type.

11.6 Conclusion and Perspectives

Although lexical and conceptual approaches are mostly considered to be concurrent and exclusive strategies, hybrid IRSs can benefit from their complementarity to enhance information retrieval and presentation. Indeed, as stressed in the review proposed in this chapter, these two strategies are tailored to different kinds of system (open or closed), different granularity (document or sentences), and hybrid IRS aims to pull together their strengths. A review of these hybrid IRSs shows that most of them use different strategies to combine the results of the two approaches so as to rank documents according to both view-points. They thus somehow still consider these two approaches as competitive solutions. We describe an alternative combination that we implement in a hybrid IRS dedicated to scientific article retrieval. Relevant documents are retrieved via their conceptual indexing and then segmented to highlight passages that could be of particular interest for users.

The idea is to use each approach where it excels rather than to somehow average their points of view at each step of the search process. We thus propose to first use a conceptual model for document retrieval. The relevance of documents w.r.t. a query is then computed using both semantic similarity based on the conceptual model and users' preferences through a weight distribution over query concepts. Secondly, an explanation step, based on an original visualization system, helps users gain insight into the results and facilitates interaction with the search engine for query reformulation. In addition to this relevance map, the user may require a more precise analysis of the document relevancies. Each relevant document is thus segmented to highlight all the text portions related to the query concepts. The text portions do not necessarily contain any query concept labels but rather terms that have been related to the concept lexicons in the segmentation step.

The resulting *CoLexIR* system was evaluated through a case study based on a corpus of BMC Cancer papers. This case study highlights the usefulness of the *CoLexIR* functionalities and illustrates how its rendering and segmentation of retrieved papers allow users to rapidly identify relevant documents and grasp their key information (w.r.t. user needs) by reading sentences focused on the query terms. As expected, the main conclusions of the papers, as they appeared in the abstract,

were actually selected by the segmentation process. In addition, the excerpts help to place these conclusions in their context and to retrieve additional relevant information scattered throughout the paper.

Excerpts selected by *CoLexIR* generally ranged from technical information (as found in "material and methods" sections or in figure legends) to general information (as found in the abstract, introduction, discussion and conclusion sections). From a scientific standpoint, the technical information was generally of relatively low interest. The general approach of *CoLexIR* does not take advantage of the fact that BMC papers are strongly structured documents. It could be worth taking this information into account so as to enable end-users to select sections of scientific papers from which *CoLexIR* should extract excerpts. Further integration of lexical and conceptual approaches in *CoLexIR* could thus be beneficial. When scientific reviews do not specify that the article must be structured using pre-defined sections, preprocessing of the corpus could be carried out in order to identify sections from which technical details are derived using a supervised lexical approach.

Acknowledgements This work is partially supported by the AVieSan national program (French national alliance for life sciences and health) and by the French Agence Nationale de la Recherche 'Investissements d'avenir/Bioinformatique' [ANR-10-BINF-01-02 'Ancestrome'].

References

1. An, R.A., Morris, J., Hirstt, G.: Lexical cohesion computed by thesaural. Comput. Linguist. **17**, 21–48 (1991)
2. Badra, F., Despres, S., Djedidi, R.: Ontology and lexicon: the missing link. In: Slodzian, M., Valette, M., Aussenac-Gilles, N., Condamines, A., Hernandez, N., Rothenburger, B. (eds.) Workshop Proceedings of the 9th International Conference on Terminology and Artificial Intelligence, pp. 16–18. INALCO, Paris (2011)
3. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM, New York; Addison-Wesley (1999)
4. Baziz, M., Boughanem, M., Pasi, G., Prade, H.: An information retrieval driven by ontology: from query to document expansion. In: RIAO. ACM, pp. 301–313. New York (2007)
5. Bhagdev, R., Chapman, S., Ciravegna, F., Lanfranchi, V., Petrelli, D.: Hybrid search: effectively combining keywords and semantic searches. In: Proceedings of the 5th European semantic web conference on the Semantic Web: Research and Applications, ESWC'08, pp. 554–568. Springer, Berlin/Heidelberg (2008)
6. Buitelaar, P., Cimiano, P., McCrae, J., Montiel-Ponsada, E., Declerck, T.: Ontology lexicalisation: the lemon perspective. In: Slodzian, M., Valette, M., Aussenac-Gilles, N., Condamines, A., Hernandez, N., Rothenburger, B. (eds.) Workshop Proceedings of the 9th International Conference on Terminology and Artificial Intelligence, pp. 33–36. INALCO, Paris (2011)
7. Caillet, M., Pessiot, J.F., Reza Amini, M., Gallinari, P.: Unsupervised learning with term clustering for thematic segmentation of texts. In: Proceedings of RIAO, pp. 648–656. CID, Paris (2004)
8. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics, vol. 23, pp. 26–33. ACL, Stroudsburg, PA, USA (2000)

9. Christensen, H., Kolluru, B., Gotoh, Y., Renals, S.: From text summarisation to style-specific summarisation for broadcast news. In: Proceedings of ECIR 2004: European conference on IR research No27, Sunderland, ROYAUME-UNI (05/04/2004), vol. 2997, pp. 223–237, ISBN 3-540-21382-1. Springer, Berlin, Germany (2004)
10. Chuang, W.T., Yang, J.: Extracting sentence segments for text summarization: a machine learning approach. In: Proceedings of the 23rd ACM SIGIR, pp. 152–159. ACM, New York (2000)
11. Cimiano, P., Buitelaar, P., McCrae, J., Sintek, M.: Lexinfo: a declarative model for the lexicon-ontology interface. Web Semant. Sci. Serv. Agents WorldW. Web **9**(1), 29–51 (2011)
12. Clifton, C., Cooley, R., Rennie, J.: Topcat: data mining for topic identification in a text corpus. In: Proceedings of the 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases. Springer, Berlin/New York (2002)
13. Cockburn, A., McKenzie, B.: 3D or not 3D?. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM New York, NY, USA (2001)
14. Cohen, A.M., Hersh, W.R.: A survey of current work in biomedical text mining. Brief. Bioinform. **6**(1), 57–71 (2005)
15. Dragoni, M., Pereira, C.D.C., Tettamanzi, A.G.B.: An ontological representation of documents and queries for information retrieval systems. In: Proceedings of the 23rd International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems – Volume Part II, IEA/AIE'10, pp. 555–564. Springer, Berlin/Heidelberg (2010)
16. Dubois, D., Prade, H.: A review of fuzzy set aggregation connectives. Inf. Sci. **36**(1-2), 85–121 (1985)
17. Duthil, B., Trousset, F., Roche, M., Dray, G., Plantié, M., Montmain, J., Poncelet, P.: Towards an automatic characterization of criteria, DEXA '11. In: Proceedings of the 22nd International Conference on Database and Expert Systems Applications DEXA 2011, p. 457. Springer, Berlin/New York (2011)
18. Fox, C.J.: Lexical analysis and stoplists. In: Frakes, W.B., Baeza-Yates, R. (eds.) Information Retrieval: Data Structures & Algorithms, pp. 102–130. Prentice-Hall, Inc. Upper Saddle River, NJ, USA (1992)
19. Friedenson, B.: The BRCA1/2 pathway prevents hematologic cancers in addition to breast and ovarian cancers. BMC Cancer **7**, 152 (2007)
20. Gillick, D., Favre, B., Hakkani-tür, D.: The icsi summarization system at tac 2008. In: Proceedings of the Text Analysis Conference Workshop, pp. 801–815. National Institute of Standards and Technology Gaithersburg, Maryland, USA (2008)
21. Giunchiglia, F., Kharkevich, U., Zaihrayeu, I.: Concept search. In: ESWC, pp. 429–444. Springer Berlin Heidelberg (2009). http://link.springer.com/chapter/10.1007/978-3-642-02121-3_33?null
22. Haav, H., Lubi, T.: A survey of concept-based information retrieval tools on the web. In: 5th East-European Conference, ADBIS 2001, Vilnius. Springer, Berlin/New York (2001)
23. Hearst, M.A.: Texttiling: segmenting text into multi-paragraph subtopic passages. ACM **23**, 33–64 (1997)
24. Hersh, W.: Evaluation of biomedical text-mining systems: lessons learned from information retrieval. Brief. Bioinform. **6**(4), 344–356 (2005)
25. Hulth, A., Megyesi, B.B.: A study on automatically extracted keywords in text categorization. In: Proceedings of 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (CoLing/ACL). ACL, Stroudsburg, PA, USA (2006)
26. Joris, D., Paul-Armand, V., Joris, V., Dirk, C., Joost, R.D.: Topic identification based on document coherence and spectral analysis. Inf. Sci. **181**, 3783–3797 (2011)
27. Kan, M.Y., Klavans, J.L., McKeown, K.R.: Linear segmentation and segment significance. In: Proceedings of the 6th International Workshop of Very Large Corpora, Montreal, pp. 197–205 (1998)
28. Kleiber, G.: Noms propres et noms communs: un problème de dénomination. Meta, **41**, 567–589 (1996)

29. Kozima, H.: Text segmentation based on similarity between words. In: ACL, pp. 286–288. ACL, Morristown (1993)
30. Kupiec, J., Pedersen, J., Chen, F.: A trainable document summarizer. In: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 68–73. ACM, New York (1995)
31. Lamprier, S., Amghar, T., Levrat, B., Saubion, F.: Seggen: a genetic algorithm for linear text segmentation. In: IJCAI'07, pp. 1647–1652. AAAI, Menlo Park, California, USA (2007)
32. Lin, D.: An Information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning, pp. 296–304. Morgan Kaufmann, San Francisco, California, USA (1998)
33. Lin, H.T., Chi, N.W., Hsieh, S.H.: A concept-based information retrieval approach for engineering domain-specific technical documents. Adv. Eng. Inf. **26**, 349–360 (2012)
34. Malioutov, I., Barzilay, R.: Minimum cut model for spoken lecture segmentation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006), pp. 25–32. ACL, Stroudsburg (2006)
35. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
36. McDonald, D., Hsinchun, C.: Using sentence-selection heuristics to rank text segments in txtractor. In: JCDL'02, pp. 28–35. ACM, New York (2002)
37. Misra, H., Yvon, F., Cappé, O., Jose, J.: Text segmentation: a topic modeling perspective. Inf. Process. Manag. **47**, 528–544 (2011, in press). Corrected Proof
38. Moens, M.F., De Busser, R.: Generic topic segmentation of document texts. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01, pp. 418–419. ACM, New York (2001)
39. Niles, I., Pease, A.: Towards a standard upper ontology. In: Proceedings of the International Conference on Formal Ontology in Information Systems – FOIS '01, Ogunquit, pp. 2–9. ACM, New York (2001)
40. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval – SIGIR '98, Melbourne, pp. 275–281. ACM, New York (1998)
41. Prévot, L., Borgo, S., Oltramari, A.: Interfacing ontologies and lexical resources. In: Ren Huang, C., Calzolari, N., Gangemi, A., Lenci, A., Oltramari, A. Prévot, L. (eds.) Ontology and the Lexicon, a Natural Language Processing Perspective, Studies in Natural Language Processing, pp. 185, 200. Cambridge University Press, Cambridge/New York (2010)
42. Pylkas, K., Erkko, H., Nikkila, J., Solyom, S., Winqvist, R.: Analysis of large deletions in BRCA1, BRCA2 and PALB2 genes in Finnish breast and ovarian cancer families. BMC Cancer **8**, 146 (2008)
43. Ranwez, S., Ranwez, V., Villerd, J., Crampes, M.: Ontological distance measures for information visualisation on conceptual maps. In: Meersman, R., Tari, Z., Herrero P. (eds.) On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops. Lecture Notes in Computer Science, vol. 4278, pp. 1050–1061. Springer, Berlin/Heidelberg (2006)
44. Resnik, P.: Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. J. Artif. Intell. Res. **11**, 95–130 (1999)
45. Reynar, J.C.: Topic segmentation: algorithms and applications. Ph.D. thesis, Computer and Information Science. University of Pennsylvania, Pennsylvania, USA (1998)
46. Riedhammer, K., Favre, B., Hakkani-Tür, D.: Long story short? Global unsupervised models for keyphrase based meeting summarization. Speech Commun. **52**(10), 801–815 (2010)
47. Salton, G., Singhal, A., Buckley, C., Mitra, M.: Automatic text decomposition using text segments and text themes. In: Hypertext'96, pp. 53–65. ACM, New York (1996)
48. Schmid, H.: Treetagger. In: TC project at the institute for Computational Linguistics of the University of Stuttgart (1994). http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/
49. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 888–905 (1997)

50. Staab, S., Maedche, A.: Ontology learning for the semantic web. IEEE Intell. Syst. **16**(2), 72–79 (2001)
51. Stokoe, C., Oakes, M.P., Tait, J.: Word sense disambiguation in information retrieval revisited. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval – SIGIR '03, Toronto, p. 159. ACM, New York (2003)
52. Supekar, K., Chute, C.G., Solbrig, H.: Representing lexical components of medical terminologies in OWL. AMIA Annu. Symp. Proc. **2005**, 719–723 (2005)
53. Sy, M., Ranwez, S., Montmain, J., Regnault, A., Crampes, M., Ranwez, V.: User centered and ontology based information retrieval system for life sciences. BMC Bioinformatics **13**(Suppl 1), S4 (2011)
54. Wiss, U., Carr, D.: A cognitive classification framework for 3-Dimensional information visualization. Research report LTU-TR-1998/4-Lulea University of Technology (1998)
55. Xie, S., Hakkani-tür, D., Favre, B., Liu, Y.: Integrating prosodic features in extractive meeting summarization. In: Proceedings IEEE Workshop on Speech Recognition and Understanding (ASRU). IEEE, Piscataway (2009)
56. Zheng, H., Borchert, C., Jiang, Y.: A knowledge-driven approach to biomedical document conceptualization. Artif. Intell. Med. **49**(2), 67–78 (2010)

Part IV
Sentiment Analysis Thorugh Lexicon and Ontologies

Chapter 12
Detecting Implicit Emotion Expressions from Text Using Ontological Resources and Lexical Learning

Alexandra Balahur, Jesús M. Hermida, and Hristo Tanev

Abstract In the past years, there has been a growing interest in developing computational methods for affect detection from text. Although much research has been done in the field, this task still remains far from being solved, as the presence of affect is only in a very small number of cases marked by the presence of emotion-related words. In the rest of the cases, no such lexical clues of emotion are present in text and special commonsense knowledge is necessary in order to interpret the meaning of the situation described and understand its affective connotations. In the light of the challenges posed by the detection of emotions from contexts in which no lexical clue is present, we proposed and implemented a knowledge base – EmotiNet – that stores situations in which specific emotions are felt, represented as "action chains". Following the initial evaluations, in this chapter, we describe and evaluate two different methods to extend the knowledge contained in EmotiNet: using lexical and ontological knowledge. Results show that such types of knowledge sources are complementary and can help to improve both the precision, as well as the recall of implicit emotion detection systems based on commonsense knowledge.

12.1 Introduction

Research in affect has a long established tradition in many sciences, such as Philosophy, Psychology, Socio-psychology, Linguistics, Cognitive Science, Pragmatics, Marketing or Communication Science. In Artificial Intelligence (AI), although

A. Balahur (✉) · H. Tanev
European Commission Joint Research Centre, Via E. Fermi 2749, 21027 Ispra (VA), Italy
e-mail: alexandra.balahur@jrc.ec.europa.eu; hristo.tanev@ext.jrc.ec.europa.eu

J.M. Hermida
Department of Software and Computing Systems, University of Alicante, Ap. de Correos 99, 03080 Alicante, Spain
e-mail: jhermida@dlsi.ua.es

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_12,

different computational approaches to spot affect from text have been proposed since the 1970s, the field of "Affective Computing" has only been consecrated recently [30]. The need to study computational approaches to detect affect in text has become more evident in the past years, together with the development of technology and the creation of virtual environments that require the interaction between humans and machines.

Although many distinct methods were proposed for the automatic detection and classification of affect in text, the complexity of emotional phenomena and the fact that the majority of existing approaches contemplate only the word level have led to a low performance of the systems implementing the emotion detection task – e.g., the ones participating in the SemEval 2007 Task No. 14 [39]. The explanation for these results is given by the fact that such methods can only account for direct expressions of emotion, through specific affect-related words (e.g., "I am sad" contains the word "sad", which is directly related to the emotion "sadness"). However, most of the times, texts contain only indirect expressions of emotions, in descriptions of situations that based on commonsense knowledge can be interpreted as leading to an emotion (e.g., "I was away when I heard the news about my grandfather's death", which implies that the person in this situation was, most probably, experiencing the emotion of "sadness", triggered by the news of somebody's death).

In a first effort to overcome the issue of emotion detection from texts in which no or little lexical clues exist to mark the presence of a specific emotion (i.e., presence of words such as "joy", "happy", "angry", etc.), in [4, 5], we proposed a method to build a commonsense knowledge base (KB), which we called "EmotiNet" (EN), storing situations that trigger emotions, based on the principles of the Appraisal Theories [34]. The main idea behind our approach, is that situations trigger emotions based on the result of the individual evaluation of their components, in accordance to "appraisal criteria". In order to detect the values of such criteria, each such situation was represented in EmotiNet as a chain of actions, with their corresponding actors, objects, their properties and the associated emotion. In order to be able to evaluate our proposed approach, we initially concentrated only on situations that deal with family contexts. We subsequently demonstrated [4, 5] that by using this resource, we are able to detect emotion from examples in ISEAR describing family-related situatons in which little or no explicit mention of affect is present.

However, due to the fact that the knowledge contained in the EmotiNet KB is still limited, the evaluations we have conducted so far [4, 5] show that the recall of the approach still requires improvements. The latter can be two-fold: on the one hand, additional knowledge is required about situations and the emotions they trigger; on the other hand, additional information is required to be able to handle different surface realizations of the examples (e.g., "The man wept when he heard the news." is the same as "The man cried when he heard the news.", but the knowledge about the fact that "cry" is a synonym of "weep" must be obtained from an external source).

In this chapter, we present an overview of the EmotiNet construction process, as well as an analysis of the different approaches we have proposed so far, in terms of knowledge extension and emotion computation heuristics.

Subsequently, we propose two new methods to extend the knowledge contained in EmotiNet: (a) The first one uses Ontopopulis (Tanev and Magnini 2008) – a system that is able to learn semantic dictionaries based on a small set of seeds – to learn more surface realizations for the known situations and knowledge about new situations and the emotions they trigger. (b) The second method relies on new examples of situations related to specific emotions extracted using the API that is made available by the "wefeelfine.org" portal, that are processed accordingly and included as new examples in the EmotiNet KB.

Subsequent to the extension of the KB with new information from these sources, we performed additional evaluations using different heuristics and analyzed the impact of including information from structured versus unstructured sources on the automatic detection of implicit expressions of emotion.

12.2 Related Work

The work described in this chapter is related to five different areas: the Appraisal Theories, the issue of emotion detection in Natural Language Processing, the problem of knowledge base population, lexical learning and the linking of ontologies with lexical resources.

12.2.1 Appraisal Theories

The Appraisal Theories [14, 18, 20, 26] state that emotions are elicited and differentiated on the basis of the cognitive evaluation of the personal significance of a situation, object or event. In the light of these theories, the nature of the emotional reaction can be best predicted on the basis of the individual's appraisal of an antecedent situation, object or event. Thus, having a sufficiently large set of representations of situations when a specific emotion was felt, it can be possible to predict the emotional reaction based on the similarity of that situation to previous ones. The ideas underlying these theories have been implemented in automatic systems detecting and/or simulating human affective reactions, obtaining encouraging results (GENESIS – [19, 23, 36]).

12.2.2 Affect Detection and Classification in Natural Language Processing

In Natural Language Processing (NLP), previous approaches to spot affect in text include the use of models simulating human reactions according to their needs and

desires [15], fuzzy logic [41], lexical affinity based on similarity of contexts – the basis for the construction of WordNet Affect [40] or SentiWordNet [16], detection of affective keywords [32] and machine learning using term frequency [27,45] and the creation of syntactic patterns and rules for cause-effect [24]. Significantly different proposals for emotion detection in text are given in the work by Liu et al. [22] and the recently proposed framework of "Sentic Computing" [10], whose scope is to model affective reaction based on commonsense knowledge. These approaches however only aim at detecting the emotion related to separate concepts and do not take into consideration the context in which these concepts appear.

12.2.3 Knowledge Bases for NLP Applications

As far as knowledge bases are concerned, many NLP applications have been developed using manually created knowledge repositories such as WordNet [17], Cyc,[1] ConceptNet [21] or SUMO (Suggested Upper Merged Ontology[2]). Some authors tried to learn ontologies and relations automatically, using sources that evolve in time – e.g., Yago [42] which employs Wikipedia to extract concepts, using rules and heuristics based on the Wikipedia categories. Other approaches to knowledge base population were by Pantel and Ravichandran [28], and for relation learning [6]. DIPRE [9] and Snowball [1] label a small set of instances and create hand-crafted patterns to extract ontology concepts. It has been shown that a great advantage of using ontologies is the easiness with which they are employable and extendable with external sources of knowledge, as well as to other languages [3].

12.2.4 Lexical Learning

There are different approaches which perform learning of semantic classes. These approaches have been used mostly in the domain of information extraction, dictionary creation and ontology population. Bourigault [7] presented an approach for extracting of clusters of terms, based on their structure. In the domain of information extraction a dictionary learning approaches were presented by Riloff and Jones [31] and Yangarber et al. [46] – the NOMEN algorithm. In a more recent work [43] present a syntactic-based approach for ontology population, based on distributional similarity between names. Never Ending Language Learning (NELL) [11] is a project for massive bootstrapping of semantic concepts and relations from the Web. It uses distributional similarity to cluster extracted noun phrases. These approaches proved to be efficient for semi-automatic creation of lexical resources,

[1] http://www.cyc.org

[2] http://www.ontologyportal.org/

however they use language-specific parsing and cannot be applied for languages other than English. Moreover, they are specialized in the acquisition of categories of noun phrases, but not verbs and modifiers.

12.2.5 Linking Ontologies with Lexical Resources

Ontologies and lexical resources are complementary. Whereas the first can describe the semantic relations in an abstract manner and can be used to perform inferences, they lack linguistic expressivity and cannot capture the surface realizations of the concepts they contain. As such, in order to perform reasoning over natural language, these two resources must be combined [33]. An example of such an application was proposed within the the KYOTO project, in which a 3 layered-model for vocabularies and ontologies was proposed (see Chap. 1 of this book) and by Scheffczyk [33], who translate FrameNet to OWL-DL and subsequently perform inferences over FrameNet-annotated sentences and show a manner in which FrameNet can be linked to the Suggested Upper Merged Ontology (SUMO) to perform more general inferences over any natural language text. Representing lexical knowledge formally using ontologies has also been shown to be useful for creating a bridge across multilingual WordNets. In this sense [29] show how SUMO – the formal ontology that formally represents WordNet can be used as an interligua to link various WordNets in different languages, while being able to accurately verify their cross-language links by testing them against the logical definitions described by the ontology. Other applications that use ontologies and lexical resources are presented by Speranza and Magnini [38], who employ ontologies combined with lexical resources for advanced content indexing in the information retrieval both from local collections and the Web. Finally, another application combining this types of resouces is presented by Andrew Philpot and Pantel [2], who employ ontologies (linked to lexical forms) for multilingual question answering, as well as for information integration across databases.

12.3 The EmotiNet Knowledge Base

EmotiNet [4, 5] is a KB aiming to be a resource for detecting emotions in text. EmotiNet captures and stores emotional reaction to real-world situations in which commonsense knowledge plays a significant role in the affective interpretation, such as the ones presented in ISEAR. Within the KB, each situation is specified as chains of actions and their corresponding emotional labels from several situations in such a way that it facilitates the extraction of general patterns of appraisal. Action chains are sequences of action links, or simply actions, that trigger an emotion on one or more subjects. Each specific action link is described with a tuple *(actor, action type, patient, emotional reaction)*. For example, for the situation “I failed my exam

because I did not study enough", the action chains are (I, fail, exam, anger), (I, study, ?, guilt){not, enough} and the final emotion label of the situation is "guilt".

In order to test our approaches of implicit emotion detection, we have chosen a data set that contains examples of such situations, where emotions are described within situations that triggered them. These types of descriptions are called self-reported affect. In the following section, we present the data set we employed.

12.3.1 Self-Reported Affect and the ISEAR Data Set

Self-reported affect is the most commonly used paradigm in Psychology to study the relationship between the emotional reaction and the appraisal preceding it [35]. ISEAR[3] [37] (International Survey on Emotion Antecedents and Reactions), a corpus of self-reported affect, contains examples of situations in which their participants had experienced all of seven major emotions (joy, fear, anger, sadness, disgust, shame, and guilt), without mentioning the emotion explicitly. An example of entry in the ISEAR databank is: "I lent my car to my brother and I had to pay the fine for the speeding ticket he got." Each example is attached to one single emotion (e.g., "anger" in the case of the previous example).

For our experiments, we employed the 1,081 examples used in the previous work by Balahur et al. [4,5] that relate to family situations. As 175 were used to construct the core knowledge in EmotiNet, we will only use for testing the remaining 895 examples.[4]

12.3.2 Building the EmotiNet Knowledge Base

The process followed in the development of EmotiNet [4, 5] comprised the next stages: (1) the design of the EmotiNet ontology, which specifies the main concepts, properties and relations managed by the knowledge base (KB), which combines and extends three ontologies (Family, ReiAction, Emotion – see Fig. 12.1 and [4, 5] for further details); (2) the extension and population of this ontology using the situations stored in ISEAR database (and thus creation of the EmotiNet KB – Fig. 12.2); and (3) the expansion of the EmotiNet KB using different existing or created resources: commonsense knowledge bases – ConceptNet – and lexical resources – VerbOcean [12], "Core" WordNet Boyd-Graber [8], WordNet Affect [40] and SentiWordNet [16].

[3] http://www.unige.ch/fapse/emotion/databanks/isear.html

[4] For 11 examples, the Semantic Role Labeling system employed – proposed by Moreda et al. [25] had a void output.

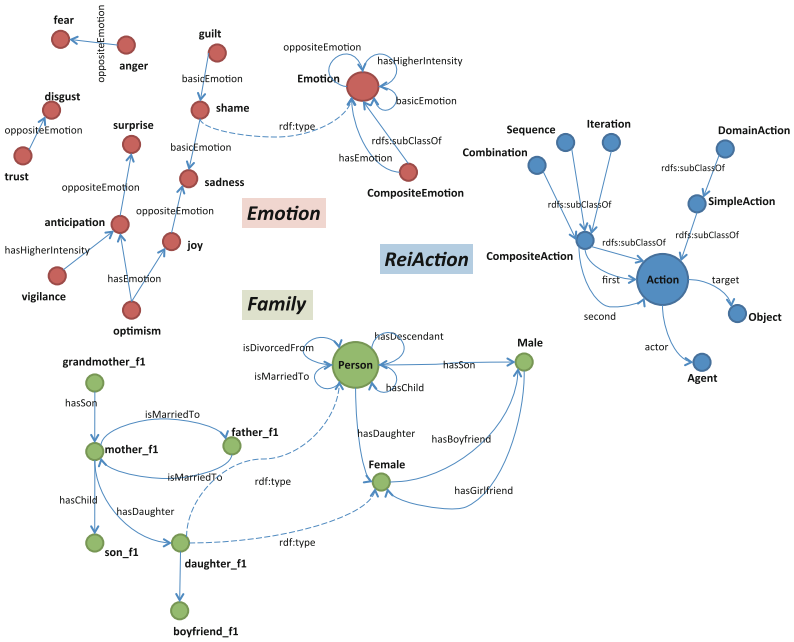

Fig. 12.1 EmotiNet ontology cores

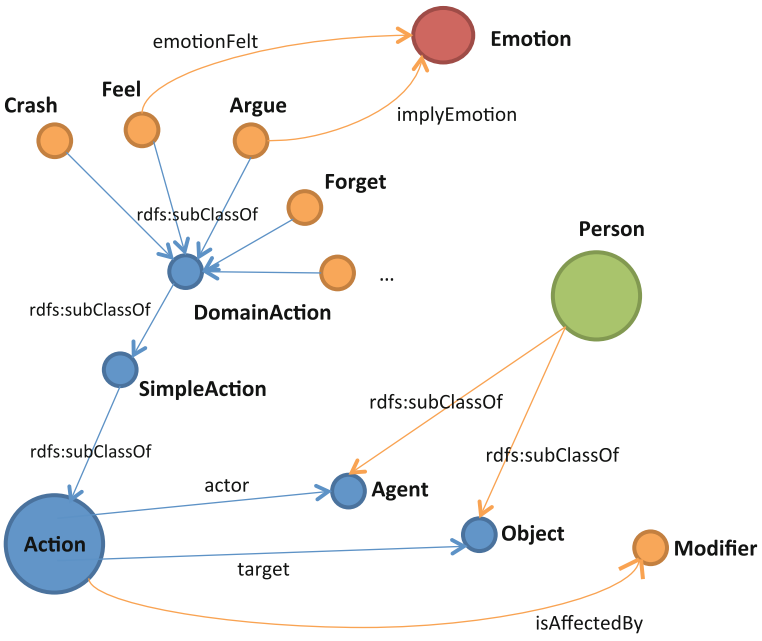

Fig. 12.2 Main concepts and examples of instances in the EmotiNet KB

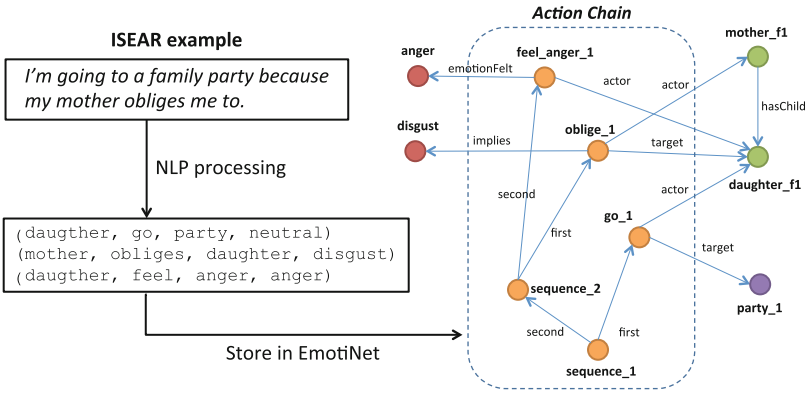

Fig. 12.3 Example of action chain extracted from the ISEAR corpus and added to the EmotiNet KB

At the end of the process, each such situation from ISEAR was represented in EmotiNet as a chain of actions, with their corresponding actors, objects, their properties and the associated emotion (see Fig. 12.3).

The following section briefly presents the different lexical and ontological resources that were used to extend EmotiNet.

12.3.3 Preliminary Extensions of EmotiNet

In our preliminary experiments, we extended EmotiNet using different types of resources. Some of them served for enriching the lexical knowledge about the terms included in the EmotiNet chains (i.e., to find similar or synonymic terms). Other resources have been used to assign emotions to the different actions involved in the chains. In the follwing subsections, we briefly present the manner in which EmotiNet was extended with each of these resources. It should be noted that these extensions have been done in different phases, and are shown in the order in which they were accomplished.

1. **VerbOcean.** In order to extend the coverage of the resource and include certain types of interactions between actions, we firstly expanded the ontology with the actions and relations from VerbOcean. In particular, 299 new actions were automatically included as subclasses of DomainAction, which were directly related to any of the actions of our ontology through three new relations: can-result-in, happens-before and similar. This knowledge extracted from VerbOcean is the basis of inferences when the information extracted from new texts does not appear in our initial set of instances.
2. **ConceptNet.** Further to the expansion of the EmotiNet core with VerbOcean, each action was associated with an emotion, using ConceptNet relations and

concepts. Action chains were represented as chains of actions with their associated emotion, as found in the ConceptNet resource.

3. **SentiWordNet.** SentiWordNet has been used to assign the same emotion to similar actions, i.e., if one action had been assigned a specific emotion label in the EmotiNet core of knowledge and we found an action that was synonymic according to this resource, we assigned it the same emotion label.
4. **Core WordNet.** The "Core" WordNet data were employed to extend EmotiNet at a lexical level. Specifically, we searched this resource for synsets containing actions that were stored in the EmotiNet core and linked the synonyms of these actions using the (*CWN_similar* relation) to the actions imported from "Core" WordNet.
5. **WordNet Affect.** This extension aimed at extending the EmotiNet KB with knowledge on the affective value of the actions contained. Specifically, the actions found in EmotiNet that were also found in the WordNet Affect (WNA) resource were assigned the corresponding affective category in this resource.
6. **LIWC.** The LIWC resource has been employed in the same manner in as WordNet Affect and ConceptNet, i.e., to assign the emotions associated to each action in EmotiNet that was found pertaining to it. However, as opposed to the case of WordNetAffect and ConceptNet, the emotions associated to the actions that could be found in LIWC have been automatically extracted and were added to EmotiNet using the *infer* relationship. More specifically, three word categories from LIWC have been employed, i.e., Anx (LIWC code 128), Anger (LIWC code 129) and Sad (LIWC code 130), as LIWC only contains words associated to to anxiety (as a subtype of fear), anger and sadness.

Although each of these resources brought an improvement in some sense to EmotiNet, the recall of the KB still remained low after the extensions. The explanation for this is that, although most of these collections are high in accuracy and some of them have a high recall, their impact on the performance of EmotiNet is limited by the small number of action chains contained in the core of this resource. Another reason for the lower impact is also that some of the resources are obtained automatically or semi-automatically and they contain a lot of noise (e.g. ConceptNet, SentiWordNet, VerbOcean) and some of the terms they contain are conceptually incorrect, incorrectly linked, too fine-grained, coarse-grained or they contain terms whose granularity is different, so ideally they should not be added as instances on the same conceptual level. Another problem is also that mapping words to ontological concepts requires disambiguation, which is a difficult problem in Natural Language Processing. Thus, even if some resources are highly accurate (LIWC, WNA, Core WN), adding the related terms to a concept from EmotiNet based only on the word form may lead to ambiguity issues. Finally, there are only a few existing lexica and, as we have seen in our previous experiments [4, 5], the extension of EmotiNet with the words they contain is not enough. Therefore, there is a need to obtain new resources which can be added to EmotiNet, in order to, on the one hand, extend its collection of action chains that lead to a specific emotion and, on

the other hand, extend its knowledge on individual actions found in these chains that lead to a specific emotion. These extensions are presented in the following section.

12.4 Further Extensions of EmotiNet with Lexical and Ontological Resources

12.4.1 Extending EmotiNet with Additional Emotion-Triggering Situations

In order to extend the knowledge on emotion-triggering situations in EmotiNet (whole action chains, as the ones contained in the EmotiNet core), we employed the API that is put at the disposal of the public by the *wefeelfine.org*' portal.[5] "Since August 2005, We Feel Fine has been harvesting human feelings from a large number of weblogs. Every few minutes, the system searches the world's newly posted blog entries for occurrences of the phrases "I feel" and "I am feeling". When it finds such a phrase, it records the full sentence, up to the period, and identifies the "feeling" expressed in that sentence (e.g., sad, happy, depressed, etc.).[6]

In order to have a sufficiently large number of examples, but also deter the introduction of a too high quantity of noise, we have extracted a maximum number of 1,500 examples of emotions per year, from 2004 to 2011. This was done invoking the *wefeelfine.org* web service using a collection of queries (the example[7] is for the year 2004 and the emotion "joy", which in this collection can be found within examples grouped under "happy" and "joyful").

By executing these queries for all the 7 emotions found in ISEAR, we obtained 10,114 examples of situations when anger was felt, 4,769 examples of disgust, 8,925 examples of fear, 12,024 examples of guilt, 11,554 examples of joy, 21,041 examples of sadness and 7,020 examples of situations when shame was felt.

In order to populate EmotiNet with these examples, we processed them in the same manner as the initial chains that are present in the EmotiNet core [4, 5]. Thus, each sentence corresponding to each emotion was processed by the SRL system by Moreda et al. [25]. Subsequently we extracted from the output of this system the triplets corresponding to the actor, action and object related to each verb. Further on, we applied the same heuristics presented by Balahur et al. [4, 5] to order the actions on a temporal line. Due to the fact that the examples originate from blogs, they contain many spelling mistakes and/or sometimes lack punctuation signs. In order to extend EmotiNet with quality examples, we cleaned the resulting chains,

[5] http://wefeelfine.org/api.html

[6] http://wefeelfine.org/mission.html

[7] http://api.wefeelfine.org:8080/ShowFeelings?display=xml&returnfields=sentence&postyear=2004&feeling=happy&limit=1500

eliminating the ones that contained only auxiliary verbs or the verb "feel", and also the incorrect spellings of negated modals (e.g., "shouldn t"). After this process of cleaning, we obtained the following number of action chains per emotion: anger – 6,931; disgust – 3,631; fear – 7,065; guilt – 8,866; joy – 7,768; sadness – 5,551; shame – 4,925. This collection of examples, which we subsequently use in our evaluations, is denominated **WFF**.

12.4.2 Extending EmotiNet Using Ontopopulis

In order to add more knowledge on the affective connotations of the actions contained in the chains in EmotiNet, we expanded the lexica used in EmotiNet using Ontopopulis, a weakly-supervised multilingual system for learning of semantic classes. This system uses Harris' distributional hypothesis, according to which words with similar meaning tend to appear in similar contexts. Ontopopulis is based on ideas described earlier by Tanev and Magnini [43]; it is also similar in spirit to the NOMEN algorithm [46] and close to Mitchell's NELL (Never Ending Language Learning).[8] It is known that such approaches are error prone, since the syntactic context alone cannot define unambiguously the semantics of a concept (Mitchell talks about the so called "semantic drifting").

The Ontopopulis system takes as input a small set of seed terms for each semantic category under consideration and and an unannotated corpus of news articles. Then, it proposes candidate terms, which most likely belong to this category or at least are closely related to it.

In our experimental settings, for each emotion we provided related seed words. Ontopopulis performs two learning steps – feature extraction and term extraction. In the feature extraction phase the system finds typical contextual features for the seed words. In the term extraction phase, it learns new terms which co-occur with the same contextual features.

For each considered emotion, we constructed a seed set of mostly verb phrases, but also nouns and adjectives, which describe situations which trigger this emotion. We used three sources to construct our seed sets: the dictionaries of the LIWC – Linguistic Inquiery and Word Count Software (http://www.liwc.net/), the ConceptNet ontology and our in-house term extraction algorithm.

LIWC provides word classes for three of the emotions, in which we are interested: anger, sadness, and fear (in LIWC the words about this last emotion are in the more generic class Anxiety). We manually chose the words which best represent each category. Moreover, we tried to give more verbs, since our model uses verbs in the action chains.

In ConceptNet we searched for action concepts which are connected to the concept nodes of the considered emotions via the relation Causes, we also considered

[8] http://rtw.ml.cmu.edu/rtw/

Table 12.1 Top five verbs and top five base forms learned by Ontopopulis for each emotion

Emotion	Top 5 acquired terms	Top 5 acquired verbs (base forms)
Anger	Sexually assaulted, sexually, robbed, sexually abused, stabbed	Assault, rob, abuse, stab, abduct
Disgust	Tortured, beaten, sexually abused, angry, assaulted	Torture, beat, abuse, assault, sadden
Fear	Anxiety, angry, pain, shocked, saddened	Pain, shock, sadden, worry, frighten
Guilt	Commit, oath, commit genocide, abuse, murder	Commit, abuse, murder, distribute, tenure
Joy	Singing, listen, perform, music, watch	Sing, listen, perform, watch, laugh
Sadness	Grief, sadness, pain, saddened, shocked	Pain, sadden, shock, suffer, mourn
Shame	Disclose, acknowledge, revealing, conceal, confirm	Disclose, acknowledge, reveal, conceal, confirm

subevents of these action concepts. Then, we manually selected actions, which were not very ambiguous with respect to their emotional effect, neither they were too specific. Following this procedure, we succeeded to identify good action concepts for the guilt and joy emotions. For the other emotions, we were not able to find relevant enough actions.

Regarding the remaining two emotions – disgust and shame, we created semi-automatically the seed sets. First we found few words, which were directly related to the considered emotions. We run an in-house-built term-extraction algorithm which found terms, which co-occur with these words. Then, we manually selected the relevant words.

We run Ontopopulis with each of the obtained seed sets and obtained a ranked list of candidate terms for each of the seven emotions we considered. Further on, we use the Freeling POS-tagger and filter out all the words in the output which are not verbs.

The top five learned terms and the top five verbs in base form that were learned for each emotion are presented in Table 12.1. As we can see from this table, some of the terms we acquired, such as "abuse", are acquired for more than one category of affect, as they are terms that relate to different emotions, depending on the context and/or usually such contexts trigger not only one, but several related emotions (e.g. anger with disgust and guilt). Such terms can be considered to be related potentially to more than one emotion and, in these cases, either the context can differentiate among different affective interpretations or a larger set of potential affective connotations can be assigned to the situation.

Subsequently, we created different sets, which we will separately use to extend EmotiNet. The first collection contains all the terms returned by Ontopopulis for all the emotions (we denote the collection **Onto**). The second collection contains all the terms that have a score that is higher than 5 % of the highest score obtained by a word retured for the corresponding emotion (we denote the collection **Onto5**). The third collection contains all the terms that have a score that is higher than 10 %

Table 12.2 List of anchors used to compute the NGD scores for Onto10 web validation

Emotion	Anchor1	Anchor2	Anchor3	Anchor4	Anchor5	Anchor6
Anger	Anger	Angry	Fury	Rage	Madness	Irritate
Disgust	Disgust	Disgusting	Repulsion	Repel	Sicken	Nauseate
Fear	Fear	Afraid	Fright	Scare	Panic	Horror
Guilt	Guilt	Guilty	Remorse	Regret	Sorry	Lament
Joy	Joy	Happy	Joyful	Happiness	Excitement	Cheerful
Sadness	Sadness	Sad	Sorrow	Desolation	Unhappiness	Grief
Shame	Shame	Ashamed	Disgrace	Embarrasement	Humiliation	Dishonor

Table 12.3 Number of terms in each Ontopopulis collection

Emotion	Total (#)	Top scored (5 %) (#)	Top scored (10 %)	Top scored NGD validated (#)
Anger	4,287	70	32	3
Disgust	4,031	81	44	25
fear	3,999	41	23	12
Guilt	3,650	29	12	2
Joy	7,092	58	22	8
Sadness	4,630	111	30	3
Shame	4,536	87	28	2

of the highest score obtained by a word retured for the corresponding emotion (we denote the collection **Onto10**). With the fourth collection of terms that we built from the output of Ontopopulis, we aimed to automatically assess the quality of the obtained lists of words. In order to accomplish this, we applied a method based on web validation, using the Normalized Google Distance [13] score (NGD). The complete list of anchors used is presented in Table 12.2. For each of the terms found in **Onto10**, we computed the NGD score between the word and a set of six anchor words for each emotion. The anchors were established in agreement by two persons, and were determined using WordNet synonyms. The queries for computing the score were launched using the Bing API.[9] The "validated" words were considered to be the ones whose cumulated NGD scores for the emotion under which they had been assigned by Ontopopulis were lower than the cumulated NGD scores for each of the other emotions. We denominated the set of validated terms **OntoNGD**.

Table 12.3 shows the number of words contained, for each of the emotions, by each of the four Ontopopulis-based collections.

Finally, EmotiNet is extended using the four collections of terms derived from the Ontopopulis output, using the "implies" relation, obtaining four different test collections. Each of them contains the EmotiNet version in which each action in an existing chain that has an emotion associated in Ontopopulis is linked to it through the "implies" relation.

[9] http://www.bing.com/developers/s/APIBasics.html

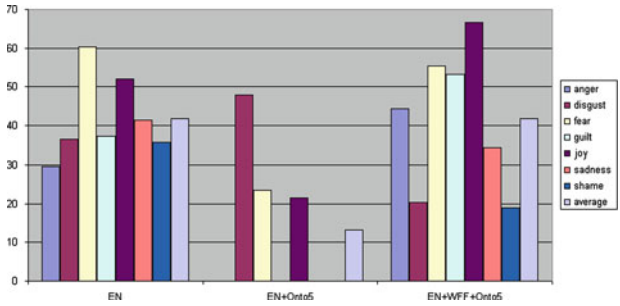

Fig. 12.4 Precision action similarity

12.5 Evaluation

In order to assess the accuracy of the EmotiNet KB in the context of the implicit emotion detection task, we performed two types of experiments:

- *Experiments using the EmotiNet action chains and* ***action*** *similarity.* When a new situation is assessed, we automatically obtain an action chain of the text and, subsequently, we compute the similarity between the emotion chain of the new situation and the EmotiNet emotion chains. The resulting emotion has the same label as the EmotiNet action chain with the highest similarity score. This type of experiments were performed by Balahur et al. [4, 5] on EN, and on EN with different combinations of resources – VerbOcean (VO), WordNet Affect (WNA) and Core WordNet (WNC). The best results were obtained using the combination of EN with all these resources. In the present approach, we added the WFF action chains to EmotiNet and evaluated the KB thus obtained in combination with the VO, WNA and WNC lexical resources. A comparative summary of the results obtained using this method is presented in Fig. 12.4 (in terms of Precision) and in Fig. 12.5 (in terms of Recall).
- *Experiments using the EmotiNet emotion chains and* ***emotion*** *similarity.* This second set of experiments is based on the use of the *implies* relationship, which associates an action to the possible emotions felt by the agents of that action. We have performed different experiments in which we have automatically annotated the actions contained in EmotiNet using a different resource in each experiment. At this stage, for each action chain in EmotiNet it is possible to obtain an emotion chain associated to the first one.

  When a new situation is assessed, we automatically obtain an action chain of the text and the emotion chain associated to this new action chain using one of the available resources. Subsequently, we compute the similarity between the emotion chain of the new situation and the EmotiNet emotion chains.

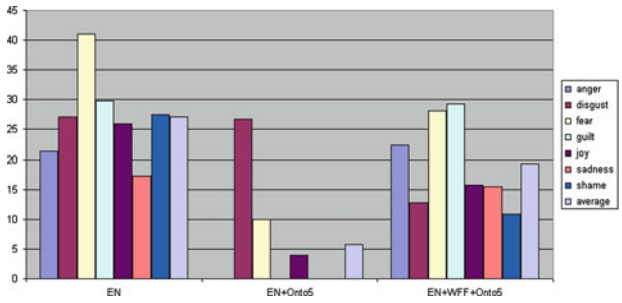

Fig. 12.5 Recall action similarity

Table 12.4 Emotion similarity (a): EmotiNet + FilesOntopopulis

	Experiment							
	e1		e2		e3		e4	
Emotion	Precision (%)	Recall (%)	P	R	P	R	P	R
Anger	0	0	0	0	0	0	0	0
Disgust	85.54	82.55	47.91	26.74	82.60	22.09	0	0
fear	13.00	11.81	23.40	10.00	0	0	11.11	0.90
Guilt	0	0	0	0	0	0	0	0
Joy	0	0	21.42	3.94	28.57	2.63	0	0
Sadness	0	0	0	0	0	0	0	0
Shame	0	0	0	0	0	0	0	0

[e1] Emotion similarity: EmotiNet + Onto
[e2] Emotion similarity: EmotiNet + Onto5
[e3] Emotion similarity: EmotiNet + Onto10
[e4] Emotion similarity: EmotiNet + OntoNGD

The resulting emotion has the same label as the EmotiNet action chain with the highest similarity score. This type of experiments were performed on the EmotiNet core (EN) by Balahur et al. [4, 5]. Subsequently, after the extension of EN with the four different collections of ontopopulis terms, a comparative evaluation has been performed on these four new resources (Table 12.4). Finally, the EmotiNet core has been first extended with the WFF chains of actions and the actions in the resulting resource have been assigned to emotions in according to the four collections of terms created with Ontopopulis. The results of this latter approach are summarized in Table 12.5. The best performing resources, EN, EN in combination with Onto5 and EN in combination with WFF (Table 12.6) and Onto5 are represented in Fig. 12.6 (in terms of Precision) and Fig. 12.7 (in terms of Recall).

Table 12.5 Emotion similarity (b): EmotiNet + FilesOntopopulis

	Experiment							
	e5		e6		e7		e8	
Emotion	Precision (%)	Recall (%)	P	R	P	R	P	R
Anger	14.45	13.79	44.31	22.41	58.82	17.24	52.38	6.32
Disgust	0	0	20.37	12.79	3.12	1.16	0	0
Fear	29.00	26.36	55.35	28.18	51.72	13.63	64.70	10.00
Guilt	98.57	93.69	53.27	29.27	41.53	12.16	90.00	8.10
Joy	90.00	71.05	66.66	15.78	55.55	6.57	100.00	5.26
Sadness	62.55	45.20	34.35	15.41	55.22	12.67	55.26	7.19
Shame	0	0	18.84	10.92	37.14	10.92	16.66	1.68

[e5] Emotion similarity: EmotiNet + WFF + Onto
[e6] Emotion similarity: EmotiNet + WFF + Onto5
[e7] Emotion similarity: EmotiNet + WFF + Onto10
[e8] Emotion similarity: EmotiNet + WFF + OntoNGD

Table 12.6 Action similarity: Emotinet + WFF

Emotion	Total (#)	Result (#)	Result (%)	Correct result (#)	Precision (%)	Recall (%)
Anger	174	169	97.12	55	32.54	31.60
Disgust	86	85	98.83	12	14.11	13.95
Fear	110	105	95.45	54	51.42	49.09
guilt	222	218	98.19	156	71.55	70.27
Joy	76	68	89.47	61	89.70	80.26
Sadness	292	220	75.34	132	60.00	45.20
Shame	119	115	96.63	26	22.60	21.84

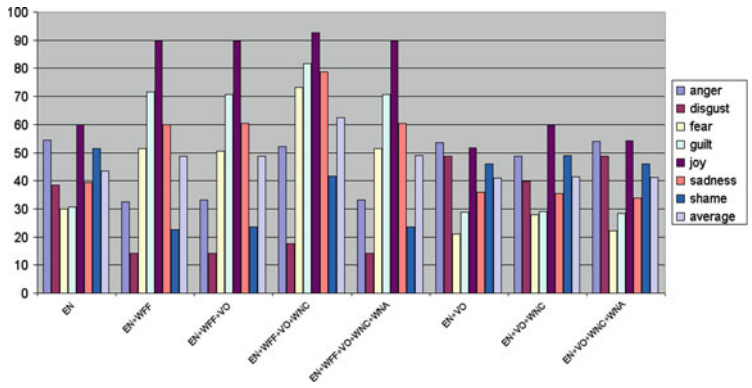

Fig. 12.6 Precision emotion similarity

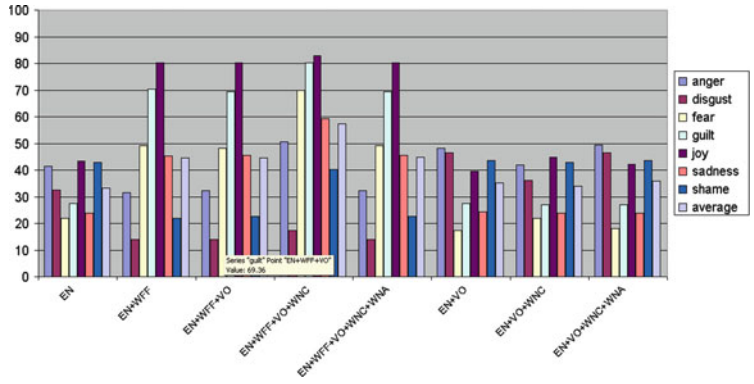

Fig. 12.7 Recall emotion similarity

12.6 Discussion, Conclusions and Future Work

From the results of the experiments, we can draw several conclusions. First of all, we can see that there are significant differences between the performance obtained using different resources. Secondly, we can notice that lexical and ontological resources are complementary and that adding both types of knowledge helps to improve both the precision, as well as the recall of the approach for some emotions (see approach using EmotiNet, WFF and Ontopopulis together). This is especially true for the action similarity-based approach, in which we can see that the use of WFF together with the lexical resources (VO, WNA, WNC) has led to improved results, both in precision, as well as recall, in four of the emotions – fear, guilt, joy and sadness. The same phenomena can be noticed from the results of the combinations of EN with Onto5, in which improvements can be noticed in precision, for anger, guilt and joy. This improvement, as can be seen from the results, comes mainly from the use of new examples from WFF, and the precision of the action-emotion associations learned by Ontopopulis. This demonstrates that new knowledge on situations helps to better discriminate among the different affective results of chains of actions and eliminates the ambiguity of actions that can be associated to multiple emotions. The best average results for all emotions are obtained using EmotiNet and WFF. Given the fact that WFF is a resource we have produced on the fly, using a web portal, this improvement in fact demonstrates the validity of our approach using EmotiNet, showing that it is possible to extend the resource without any human intervention and with large quantities of information. In this case, we have only used a small number of examples from the "wefeelfine.org" portal, but in the future, the number of situations to be imported can be increased substantially. The evaluations have shown that, although such a resource is noisy, the manner in which EmotiNet stores and deals with the action chains makes it stable against the aquired noise. As far as the anger, disgust and shame emotions, we can see that the introduction of new examples leads to a decrease in the results. This is mainly due to the fact that they

were more difficult to distinguish among themselves, as the situations in which these emotions were described contained similar vocabulary (e.g., people being angry at themselves for eating too much when they have problems with weight). These findings point to the fact that some of the imported examples could benefit from human annotation, in the sense of confirmation of this existing ambiguity and of a possible multiple emotion annotation of these examples.

Looking at Figs. 12.6 and 12.7, we can see that the Ontopopulis-obtained resources alone improve just the precision of the emotion disgust. On the other hand, combining WFF and Ontopopulis has a positive impact on the precision of guilt and joy detection. Unfortunately, this is compensated by the decreased performance on the other emotions and the overall effect of the resources, obtained by this lexical learning algorithm, is negative. After investigating this fact, we reached the conclusion that the decreased performance was mostly due to the fact that word lists overlapped between emotions and this had negative effects on our action chain selection algorithm. The terms, learned by Ontopopulis, in general seemed related to the corresponding emotions (see Table 12.1). However, there was significant overlap, since one word can be related to situations which imply more than one emotion. In our future experiments we can improve the learning algorithm, so that it automatically chooses the best emotion, to which a word is related. In this way we can resolve the problems with the overlapping lists.

Other future work includes the study of alternative methods to integrate terms learned with Ontopopulis in EmotiNet, additional extensions of EmotiNet using more examples extracted from "wefeelfine" and other web sources and the evaluation of EmotiNet on the entire dataset of ISEAR examples, as well as on other emotion-annotated datasets (e.g., the SemEval 2007 data, that has multiple emotions annotated). Finally, EmotiNet will be extended to take into account the modifiers and properties of the actors and actions involved in the action chain, which are elements that have proven to be necessary at the time of distinguishing among connected emotions (e.g., anger with sadness). At present, the model already captures the concept of modifier in order to represent elements that modify the effect of the action on the actor or object. Within this wide concept, we aim to create a new hierarchy of modifiers (or reuse an existing one) grouping elements into different categories such as intensifiers, diminishers (i.e. "valence shifters") or negations. This new hierarchy will be populated with the examples extracted from ISEAR. Subsequently, we can propose variations of our method for emotion detection that take into account the effect of the modifiers in the action chains. Regarding the properties of the actors (e.g., gender, age, culture, background), we plan to study the manner in which they are related to the EmotiNet action chains and analyse if it is possible to create different collections of action chains for different countries or cultures.

Acknowledgements The work by Jesús M. Hermida has been supported by the Spanish Ministry of Education under the FPU Program (ref. AP2007-03076).

References

1. Agichtein, E., Gravano, L.: Snowball: extracting relations from large plain-text collections. In: Proceedings of the 5th ACM International Conference on Digital Libraries (ACM DL), pp. 85–94. ACM, New York (2000)
2. Andrew Philpot, E.H., Pantel, P.: The Omega Ontology. Studies in Natural Language Processing, chap 15, pp. 258–270. Cambridge University Press, Cambridge (2010)
3. Aviv, S., Gal, A.: Enhancing portability with multilingual ontology-based knowledge management. Decis. Support Syst. **45**(3), 567–584 (2008)
4. Balahur, A., Hermida, J.M., Montoyo, A.: Building and exploiting emotinet, a knowledge base for emotion detection based on the appraisal theory model. IEEE Trans. Affective Comput. **3**(1), 88–101 (2012a)
5. Balahur, A., Hermida, J.M., Montoyo, A.: Detecting implicit expressions of emotion in text: A comparative analysis. Decis. Support Syst. (2012b). doi:10.1016/j.dss.2012.05.24
6. Berland, M., Charniak, E.: Finding parts in very large corpora. In: Proceedings of ACL 1999, College Park, pp. 57–64 (1999)
7. Bourigault, D., Jacquemin, C.: Term extraction plus term clustering. In: Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics (EACL-99), Bergen (1999)
8. Boyd-Graber, J., Fellbaum, C., Osherson, D., Schapire, R.: Adding dense, weighted connections to WordNet. In: Proceedings of the Third Global WordNet Meeting, Jeju Island, Korea (2006). http://wordnetcode.princeton.edu/standoff-files/core-wordnet.txt as footnote
9. Brin, S.: World-extracting patterns and relations from the wide web. In: Proceedings of the 1998 International Workshop on Web and Databases (WebDB 1998), New York, pp. 172–183 (1998)
10. Cambria, E., Hussain, E., Havasi, C., Eckl, C.: Affective space: blending common sense and affective knowledge to perform emotive reasoning. In: Proceedings of the 1st Workshop on Opinion Mining and Sentiment Analysis (WOMSA), Seville (2009)
11. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, Atlanta (2010)
12. Chklovski, T., Pantel, P.: Verbocean: mining the web for fine-grained semantic verb relations. In: Proceedings of EMNLP 2004, Barcelona, pp. 33–40 (2004)
13. Cilibrasi, R.L., Vitanyi, P.M.B.: The google similarity distance. IEEE Trans. Knowl. Data Eng. **19**, 370–383 (2007). doi:10.1109/TKDE.2007.48
14. De Rivera, J.: A Structural Theory of the Emotions. Psychological Issues, Monograph 40, vol. 10(4). International Universities Press, New York (1977)
15. Dyer, M.: Emotions and their computations: three computer models. Cognit. Emot. **1**(1), 323–347 (1987)
16. Esuli, A., Sebastiani, F.: Determining the semantic orientation of terms through gloss analysis. In: Proceedings of CIKM 2005, Bremen, pp. 617–624 (2005)
17. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT, Cambridge (1998)
18. Frijda, N.: The Emotions. Cambridge University Press, Cambridge (1986)
19. Gratch, J., Marsella, S., Wang, N., Stankovic, B.: Assessing the validity of appraisal-based models of emotion. In: Proceedings of ACII, Amsterdam (2009). http://www.ict.usc.edu/~marsella/publications/ACII09-appraisal.pdf
20. Johnson-Laird, P.N., Oatley, K.: The language of emotions: an analysis of a semantic field. Cognit. Emot. **3**, 81–123 (1989)
21. Liu, H., Singh, P.: Conceptnet: a practical commonsense reasoning toolkit. BT Technol. J. **22**, 211–226 (2004)
22. Liu, H., Lieberman, H., Selker, T.: A model of textual affect sensing using real-world knowledge. In: Proceedings of IUI 2003, Miami, pp. 125–132 (2003)

23. Marsella, S., Gratch, J., Petta, P.: Computational models of emotion. In: Scherer, K.R., Banziger, T., Roesch, E. (eds.) A Blueprint for a Affective Computing: A Sourcebook and Manual. Oxford University Press, Oxford (2010)
24. Mei Lee, S., Chen, Y., Huang, C.: Cause event representations of happiness and surprise. In: Proceedings of PACLIC 2009, Hong Kong (2009)
25. Moreda, P., Navarro, B., Palomar, M.: Corpus-based semantic role approach in information retrieval. Data Knowl. Eng. (DKE) **61**(3), 467–483 (2007)
26. Ortony, A., Clore, G., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press, New York (1988)
27. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques. In: Proceedings of EMNLP 2002, Philadelphia, pp. 79–86 (2002)
28. Pantel, P., Ravichandran, D.: Automatically labeling semantic classes. In: Proceedings of HLT-NAACL-04, New York, pp. 321–328 (2004)
29. Pease, A., Fellbaum, C.: Formal Ontology as Interlingua: The SUMO and WordNet Linking Project and Global WordNet. Studies in Natural Language Processing, chap 2, pp. 25–35, Cambridge University Press, Cambridge (2010)
30. Picard, R.: Affective computing, vol. technical report. MIT Media Laboratory (1995)
31. Riloff, E., Jones, R.: Learning dictionaries for information extraction by multi-level bootstrapping. In: Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), Orlando (1999)
32. Riloff, E., Wiebe, J., Wilson, T.: Learning subjective nouns using extraction pattern bootstrapping. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 – Volume 4, Edmonton, Canada. CONLL '03. Association for Computational Linguistics, Stroudsburg, pp. 25–32 (2003)
33. Scheffczyk, J., Baker, C.F., Narayanan, S.: Reasoning over Natural Language Text by Means of FrameNet and Ontologies. Studies in Natural Language Processing, Expanded version of paper at OntoLex, 2006, chap 4, pp 53–71. Cambridge University Press, Cambridge (2010). ISBN-13: 9780521886598
34. Scherer, K.: Handbook of Cognition and Emotion, Wiley, Chichester (1989). Chap Appraisal Theory
35. Scherer, K.R.: Toward a dynamic theory of emotion: the component process of affective states. Geneva Studies in Emotion and Communication **1**, 1–98 (1987)
36. Scherer, K.R.: Studying the emotion-antecedent appraisal process: an expert system approach. Cognit. Emot. **7**(3–4), 323–355 (1993)
37. Scherer, K., Wallbott, H.: The ISEAR Questionnaire and Codebook. Geneva Emotion Research Group (1997)
38. Speranza, M., Magnini, B.: Merging Global and Specialized Linguistic Ontologies. Studies in Natural Language Processing, chap 13, pp 224–238, Cambridge University Press, Cambridge (2010)
39. Strapparava, C., Mihalcea, R.: Semeval-2007 task 14: Affective text. In: Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), Association for Computational Linguistics, Prague, pp. 70–74. http://www.aclweb.org/anthology/S/S07/S07-1013 (2007)
40. Strapparava, C., Valitutti, A.: Wordnet-affect: an affective extension of WordNet. In: Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004), Lisbon, pp. 1083–1086 (2004)
41. Subasic, P., Huettner, A.: Affect analysis of text using fuzzy semantic typing. IEEE Tras. Fuzzy Syst. **9**, 483–496 (2000)
42. Suchanek, F., Kasnei, G., Weikum, G.: Yago: A core of semantic knowledge unifying wordnet and wikipedia. In: Proceedings of the World Wide Web Conference 2007, Banff, pp. 697–706 (2007)
43. Tanev H, Magnini, B.: Weakly supervised approaches for ontology population. In: Proceedings of the European Chapter of the Association of Computational Linguistics, Bergen, pp 55–67 (2006)

44. Tanev, H., Magnini, B.: Weakly supervised approaches for ontology population. In: Proceedings of the 2008 Conference on Ontology Learning and Population: Bridging the Gap Between Text and Knowledge, Amsterdam, The Netherlands, pp. 129–143. IOS Press (2008)
45. Wiebe J, Riloff, E.: Creating subjective and objective sentence classifiers from unannotated texts. In: Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing), Mexico City, pp. 73–99 (2005)
46. Yangarber, R., Lin, W., Grishman, R.: Unsupervised learning of generalized names. In: Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002), Taipei (2002)

Chapter 13
The Agile Cliché: Using Flexible Stereotypes as Building Blocks in the Construction of an Affective Lexicon

Tony Veale

Abstract Our affective perspective on a word is heavily influenced by the context in which it is used and by the features it is typically perceived to exhibit in that context. A nuanced model of lexical affect thus requires a feature-rich representation of each word's potential to mean different things in different contexts. To this end, we present here a two-level model of lexical affect. At the first level, words are represented as bundles of the typical properties and behaviors they are commonly shown to exhibit in everyday language. To construct these bundles, we present a semi-automatic approach to harvesting stereotypical properties and behaviors from the Web. At the second level, these properties and behaviors are related to each other in a graph structure that captures how likely one is to reinforce the meaning of another. We present an effective means of constructing such a graph from a combination of text n-grams and queries to the open Web. We calculate positive and negative potentials for each property in the graph, and show how these potentials can be used in turn to calculate an overall affective value for the higher-level terms for which they are considered stereotypical.

13.1 Introduction

Hamlet tells us that "there is nothing either good or bad, but thinking makes it so." This reasoning applies just as much to the words we use to label things as it does to the things themselves. In some contexts, for instance, "pride" denotes an admirable quality, but in others it denotes a deadly sin. Likewise, we praise go-getters, entrepreneurs and aspiring champions for their aggressiveness, but in many other contexts "aggression" denotes an unpleasant trait. Some words in English

T. Veale (✉)
Web Science and Technology Division, KAIST, Yuseong, Korea
e-mail: tony.veale@gmail.com

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8_13,

seem inherently positive or negative, but in reality there are very few words that cannot be given a reverse spin in the right context. Thus, words like *crazy, bad, wicked, sick* and *evil* have all been re-engineered as positive descriptors in the vernacular of youth culture.

The sense inventories that lexicographers compile for a polysemous word offer a good approximation of the word's potential to convey meaning, but affect can operate across sense boundaries and even within individual senses, at the sub-sense level. Consider the word "baby", used to denote a human infant. In some contexts the word carries a positive affect: babies can be cute and adorable, curious and trusting, and an obvious target of love and affection, especially when asleep. Crying babies, however, can be selfish, whining, drooling, hissing, tantrum-throwing little monsters. Both views are stereotypical of human babies, and either can be intended when a speaker uses the term "baby" figuratively, whether to describe a beloved partner or an annoying colleague. This is a matter of conceptual perspective, not of lexical sense, and many other words exhibit a similar affective duality; "teenager" for instance can mean "whining brat" just as easily as "growing adolescent". The concepts *Baby* and *Teenager* are complex and multifaceted, and different uses in context may highlight different stereotypical behaviors of each. Their affective meaning in context is therefore not so much a function of which lexical sense is intended but of which behaviors are highlighted, and of the perceived affect of those behaviors.

Context can change the way we perceive the affect of a word or concept, and the language we use in context can reinforce this shift in perception, for language provides various means of putting an appropriate contextual spin on the perceived affect of a word. We might use an adverb with a strong affect of its own, as when we say someone is "impressively aggressive" or "disgustingly rich." We might tack the caveat "in a good way" or "in a bad way" onto the end of a description, or say "for better or worse" if we want to highlight both the positive and negative aspects of a word's meaning. Conversely, in ambiguous cases the addressee may seek clarification using the construction "good X or bad X?", as in "good strange or bad strange?" when someone has been described as strange. A pleasantly strange person may be novel, mysterious, exciting and unpredictable, whereas an unpleasantly strange person may be incomprehensible, troubling, alien and freaky.

Affective ambiguity is also found at the level of complex objects that are described in terms of these basic properties. President Barack Obama, for instance, is often criticized for acting like a "professor", though it would be an unusual dictionary that assigned a negative sense to this word. In this case, one assumes that it is the negative qualities of the stereotypical professor that are highlighted by the criticism. In turn, these negative qualities are the stereotypical traits that can be given the most negative spin, such as when scholarly objectivity and logicality are taken to be signs of emotional detachment.

In cases like "professor", we cannot rely on the lexicon to provide appropriate positive or negative senses for our words, for in practice most words can be given an affective spin in the right linguistic context. Rather, we should instead attempt to model and represent the stereotypical properties and behaviors on which different

uses of the same word will derive their affective value in context. With a sufficiently rich behavioral model, we can determine the affect of a word like "baby" or "teenager" on a case-by-case and context-by-context basis, rather than wiring a one-size-fits-all measure of average affect directly into the lexicon. In short, we propose a two-level structure for a context-sensitive affective lexicon: a mapping of word-concepts to their normative stereotypical behaviors (e.g. *mewling, shrieking, drooling, sleeping* and *smiling*); and an affective profile of those behaviors (e.g. indicating the degree to which *shrieking* is unpleasant and *smiling* is pleasant). The affect of a word/concept in context can then be calculated as a function of the affect of its stereotypical behaviors that are primed in that context. We describe the construction of this two-level model of lexical affect in this paper. At the first level we capture the stereotypical properties and behaviors of commonplace ideas and the words that denote them. At the second level, we then calculate the perceived affect of a complex object – like *baby* or *professor* – as a function of those properties that are primed in context.

With these goals in mind, the rest of the paper assumes the following structure. We begin in Sect. 13.2 with a discussion of related work in the field of lexical affect. In Sect. 13.3 we then present a computational means of acquiring the stereotypical knowledge on which the current model is predicated. This knowledge is used in Sect. 13.4 to estimate an affective value for each property and behavior in our representation, and for each complex object for which these properties and behaviors are considered stereotypical in everyday language. We outline how the two-level model can be used in an affective search application in Sect. 13.5. An empirical evaluation of the model is presented in Sect. 13.6, showing that good results are achieved on both of its levels. The paper concludes with a discussion of key issues in Sect. 13.7.

13.2 Related Work and Ideas

In its simplest form, an affect lexicon assigns an affective score – along one or more dimensions – to each word or sense. The underlying lexicon may be a pre-existing resource that covers the bulk of a language , such as WordNet [7], or it may be a collection of sentiment-bearing words that aims to cover a small but relevant subset of the language. For instance, Whissell's *Dictionary of Affect* (or *DoA*) [23] assigns a trio of numeric scores to each of its 8,000 + words to describe three psycholinguistic dimensions: *pleasantness, activation* and *imagery*. In the DoA, the lowest pleasantness score of 1.0 is assigned to words like "abnormal" and "ugly", while the highest, 3.0, is assigned to words like "wedding" and "winning". Less extreme words are assigned pleasantness scores closer to the DoA mean of 1.84. Though Whissell's DoA is based on human ratings, Turney [19] shows how affective scores can be assigned automatically, using statistical measures of word association in Web texts.

Liu et al. [10] also present a multidimensional affective model that uses the six basic emotion categories of Ekman [4] as its dimensions: *happy, sad, angry, fearful, disgusted and surprised*. These authors base estimates of affect on the contents of Open Mind, a common-sense knowledge-base [17] that was harvested from the factual contributions of volunteers on the Web. These contents are treated as sentential objects, and a range of NLP models is used to derive affective labels for the subset of contents (approx. 10 %) that appear to convey an emotional stance. These labels are then propagated to related concepts (e.g., excitement is propagated from rollercoasters to amusement parks) so that the implicit affect of many other concepts can be determined.

For reliable results on a large-scale, Mohammad and Turney [13] and and Mohammad and Yang [14] used the *Mechanical Turk* to elicit human ratings of the emotional content of different words. Ratings were sought along the eight primary emotional dimensions identified by Plutchik [16]: *anger, anticipation, disgust, fear, joy, sadness, surprise* and *trust*. Automated tests were used to exclude unsuitable raters, and in all, 24,000 + word-sense pairs were annotated by five different raters. Thus, words that suggest fearful contexts, like "threat", "hunter" and "acrobat", are all assigned a significant score on the *fear* dimension, while "disease" and "rat" score highly on the *disgust* dimension.

Strapparava and Valitutti [18] provide a set of affective annotations for a subset of WordNet's synsets [7] in a resource called *Wordnet-affect*. The annotation labels, called *a-labels*, focus on the cognitive dynamics of emotion, allowing one to distinguish e.g. between words that denote an emotion-eliciting situation and those than denote an emotional response. Esuli and Sebastiani [5] also build directly on WordNet as their lexical platform, using a semi-supervised learning algorithm to assign a trio of numbers – *positivity, negativity* and *neutrality* – to word senses in their newly derived resource, SentiWordNet. (Wordnet-affect also supports these three dimensions as a-labels, and adds a fourth, *ambiguous*). Esuli and Sebastiani [6] improve on their affect scores by running a variant of the PageRank algorithm (see also [12]) on the implicit graph structure that tacitly connects word-senses in WordNet to each other via the words used in their textual glosses.

These lexica attempt to capture the affective profile of a word/sense when it is used in its most normative and stereotypical guise, but they do so without an explicit model of stereotypical meaning. Veale and Hao [20] describe a Web-based approach to acquiring such a model. They note that since the simile pattern "as ADJ as DET NOUN" presupposes that NOUN is an exemplar of ADJness, it follows that ADJ must be a highly salient property of NOUN. The authors of [20] harvested tens of thousands of instances of this pattern from the Web, to extract sets of adjectival properties for thousands of commonplace nouns. They show that if one estimates the pleasantness of a term like "snake" or "artist" as a weighted average of the pleasantness of its properties (like *sneaky* or *creative*) in a resource like the DoA [23], then the estimated scores show a reliable correlation with the DoA's own scores. It thus makes computational sense to calculate the affect of a word-concept as a function of the affect of its most salient properties. Veale [21] later built on

this work to show how a property-rich stereotypical representation could be used for non-literal matching and retrieval of creative texts, such as metaphors and analogies.

Both Liu et al. [10] and Veale and Hao [20] argue for the importance of commonsense knowledge in the determination of affect. We incorporate ideas from both while choosing to build mainly on the latter in this paper, to construct a two-level model of the affective lexicon. We focus chiefly on the determination of positive/negative affect, but we will also show how the two-level model can use the *halo effect* [3] to support an open-ended range of affective connotations. This will prove especially useful in tasks such as affective text retrieval (e.g. Veale and Hao [22] describe an affective news retrieval system), as it allows users to concoct their own ad-hoc mood filters to suit the needs of a particular query.

Veale and Hao [20] make the simplifying but unjustified assumption that all stereotypical properties are adjectival in nature, and work from adjectival properties (as inventoried by WordNet) to the nouns that exemplify them by successively binding ADJ in the Web query "as ADJ as a NOUN" to different adjectives. The resulting enfilade of queries is sent in rapid succession to the search engine Google. All bindings for NOUN are then automatically extracted from the results before being manually inspected. Here we instead use the *like*-simile patterns"VERB+*ing* like a NOUN" and "VERB+*ed* like a NOUN", the preferred simile patterns to describe behavior. At the first level, these patterns are used to acquire a model of stereotypical properties and behaviors from the Web. At the second level, a graph organization for these properties and behaviors is also derived, again using the Web as a corpus, and this graph is used to estimate the pleasantness and unpleasantness of each vertex as a function of the pleasantness and unpleasantness of its adjacent vertices. The resulting structure can be used as a richly-featured affective lexicon that supports different kinds of stereotypical reasoning, or it can be used to augment existing ontologies – whether those built on formal foundations such as DOLCE [8], or those built on more lightweight semantic foundations such as WordNet [7] – with an additional layer of commonsense knowledge as to how everyday word-concepts are stereotypically and affectively understood.

13.3 Finding Stereotypes on the Web

Similes leverage the evocative power of stereotypes to exemplify a descriptive property. Conversely, stereotypes are learned, spread and perpetuated by their constant use in similes. Veale and Hao [20] exploit this symbiotic relationship to acquire a feature-rich representation of many everyday concepts from the Web.

Before performing another large-scale trawl of the Web, we first conduct a pilot study on the Google n-grams [2], a database of contiguous n-word strings ($1 \leq n \leq 5$) with a Web frequency of 40 or higher. For example, the pattern "VERB+*ing* like a NOUN" matches over 8,000 4-g, while "VERB+*ed* like a NOUN" matches almost 4,000. However, we find here a good deal of empty behaviors, such as *acting* (as in "acting like a baby" rather than "acting like an actor") and *looking* (as in

"looking like a fool"). Indeed, just three empty behaviors – *looking/looked* and *seemed* – account for almost 2,000 n-gram matches. Others, like *walking* and *eating*, are too general and merely allude to a stereotypical behavior (as in "walking like a penguin") rather than explicitly providing the specific behavior (e.g. *waddling*). Sifting through the n-gram matches yields a few hundred nuggets of stereotypical insight, such as "circling like a shark", "salivating like a dog" and "clinging like a leech". Our pilot study reveals that most instances of the *like*-simile patterns are not so specific and informative, making a large-scale Web trawl with these patterns impracticable.

Instead we use a hypothesis-driven approach by first looking for attested mentions of a specific behavior with a given noun. Consider the noun *zombie*: searching the Google 3-g for matches to the patterns "DET VERB+*ing zombie*" and "DET VERB+*ed zombie*" yields the following hypotheses for the stereotypical behavior of zombies (numbers in parentheses are frequencies of matching 3-g):

{decomposing(1454), devastating(134), shambling(115), rotting(103), ravaged(98), brainwashed(94), drooling(84), freaking(83), attacking(80), crazed(79), obsessed(73), infected(72), marauding(71), disturbed(65), wandering(64), reanimated(54), flying(52), flaming(52), revived(47), decaying(41), unexpected(40)}

For each attested behavior in the Google n-grams we generate the corresponding *like*-simile, such as "decomposing like a zombie", and determine its frequency on the open Web. The corresponding non-zero frequencies for these behaviors in *like*-similes on the Web, obtained using Google, are as follows for *zombie*:

{drooling(4480), wandering(3660), shambling(1240), revived(860), rotting(682), brainwashed(146), reanimated(141), infected(72), flaming(52), decaying(46), decomposing(8), attacking(7), flying(6), freaking(2), obsessed(3)}

We also harvest all three-word phrases that match the pattern "DET ADJECTIVE NOUN" in the Google 3-g, where ADJECTIVE can match any adjective in WordNet. For each property ADJECTIVE that is attested for given noun NOUN in these 3-g patterns, we generate the Web query "as ADJECTIVE as a NOUN" and dispatch that to Google also. Thus, for example, the 3-g "a mindless zombie" yields the Web query "as mindless as a zombie", which occurs in hundreds of documents on the Web. The corresponding non-zero Google frequencies for *zombie* properties in *as*-similes on the Web are as follows:

{slow(18200), scary(6550), hungry(3320), lifeless(2840), creepy(2710), mindless(890), emotionless(827), brainless(155), ravenous(81), strange(8), soulless(6), powerful(6), bizarre(2), bloody(2), brutal(2), unstoppable(2), cheesy(1), supernatural(1)}

Unlike Veale and Hao [20] then, we do not use a relatively small (approx. 2000) set of queries that are made wide-ranging through the use of wild-cards, but generate a very large set of specific queries (with no wild-cards) that each derive from an attested combination of a specific property or behavior and a specific noun in the Google n-grams. We are careful not to dispatch queries that contain empty behaviors like "looking" or "acting", a list of which is determined during our initial pilot study with the Google n-grams. In all, we dispatch over 500,000 queries to Google, for

the same number of attested combinations. No parsing of the Web results is needed, and we need record only the total number of returned hits per query/combination.

13.3.1 Web-derived Models of Typical Behavior

The 3-g patterns "DET VERB+*ing* NOUN" and "DET VERB+*ed* NOUN" attest to the plausibility of a given noun-entity exhibiting a specific behavior, but they are only weakly suggestive about what is actually typical. As a basis for generating hypotheses about stereotypical behavior these patterns over-generate significantly, and less than 20 % of our queries yield non-zero result sets when sent to the Web.

As shown by the *zombie* example above, some Web-attested behaviors are best judged as idiosyncratic rather than stereotypical. While *rotting, decaying* and *shambling* are just the kind of behaviors we expect of zombies, *freaking, flying* and *flaming* are ill-considered oddities that our behavior model can well do without. As one might expect, such oddities tend to have lower Web frequencies than more widely-accepted behaviors (like *drooling*), yet raw Web frequencies can be an unreliable guide to what is typical [9]. Note for instance how *decomposing* has a low frequency of just eight uses on the Web (as indexed by Google).

Our Web data exhibits another interesting phenomenon. Consider the noun-entities for which the behavior *brainwashed* is attested, both in the 3-g ("a brainwashed NOUN") and on the Web ("brainwashed like a NOUN"):

{cult(1090), zombie(146), robot(9), child(7), fool(4), kid(4), idiot(3), soldier(2)}

Since cults often use brainwashing, we can consider *cult* to be a stereotypical exemplar for this behavior. Zombies and robots, however, are not typically brainwashed, nor indeed are they even brainwash*able*. Rather, it is more accurate to suggest that the victims of brainwashing often resemble *robots* and *zombies*, and to the extent that brainwashing is made possible by being weak-minded, they can also resemble *fools, idiots, kids* and *children*. This appears to be an example of *ataxis* [1], insofar as *brainwashed* is a "migrant modifier" that more aptly describes the target of the simile than it does the vehicle (*robot* or *zombie*). In this case we can sensibly conclude that *brainwashed* is a figurative behavior of *robots* and *zombies* (since they typically act like a brainwashed person) and is the kind of association we want in our behavioral model. In contrast, it would not be sensible to include *brainwashing* as part of the behavioral description of *fools, idiots, kids, children* or even *soldiers* (though the latter is perhaps debatable).

Ultimately, the stereotypicality of a behavioral association is a pragmatic *gut* issue for the designer of a lexico-semantic resource, one that cannot be automatically resolved by considering Web frequency (or other statistical quantities) alone. As with the design of resources like WordNet, it is best resolved by asking and answering the question "is this an association that I would want in my lexicon?". For this reason, we filter the results of the Web harvesting process manually, to ensure that the final model contains only those qualities that a human would consider

typical. In the end then, our approach is a semi-automatic one: automated processes scour the Google n-grams for hypotheses about typical behaviors and properties, and then seek supporting evidence for these hypotheses on the Web (in the form of *as*-similes and *like*-similes). Finally, a manual pass is conducted to ensure the model has the hand-crafted quality of a resource like WordNet.

It takes a matter of weeks to perform this manual filtering, but the stereotype lexicon that results from this effort has 9,479 different stereotypes, and ascribes to each a selection of 7,898 different properties and behaviors. In all, the new resource contains over 75,000 unique noun-to-property/behavior associations, which represents a significant extension to the 12,000 + associations first harvested for Veale and Hao's original resource [20]. The term *baby*, for instance, is associated with the following 163 properties and behaviors in this new, more comprehensive resource:

{delicate, squalling, weeping, baptized, adopted, startled, attentive, blessed, teeny, rocked, adorable, whining, bundled, toothless, placid, expected, rescued, treasured, new, sleepy, indulged, slumbering, weaned, pure, supple, helpless, small, sleeping, animated, vulnerable, wailing, cradled, kicking, soft, rested, bellowing, blameless, grinning, screaming, orphaned, cherished, reliant, thriving, loveable, guileless, mute, inexperienced, harmless, dribbling, unthreatening, nursed, angelic, bawling, beaming, naked, spoiled, scared, weak, squirming, blubbering, contented, smiling, wiggling, mewling, blubbing, sniffling, overtired, dimpled, loving, dear, tired, powerless, bewildered, peaceful, distressed, naive, wee, soiled, sucking, fussy, gurgling, vaccinated, heartwarming, pouting, constipated, drooling, quiet, wiggly, lovable, bare, weaning, suckling, cute, bald, whimpering, tender, pampered, incontinent, fleshy, charming, dependent, artless, fussing, flabby, babbling, warm, giddy, crawling, snoozing, hairless, cuddled, sweet, sobbing, squealing, wrapped, tiny, cooing, swaddled, laughing, toddling, fragile, innocent, moaning, gentle, terrified, precious, cranky, giggling, confused, pink, cuddly, fat, ignorant, snoring, young, howling, screeching, shrieking, trusting, shivering, napping, resting, frightened, fresh, loved, demanding, chubby, adored, appealing, happy, tame, relaxed, wriggly, rocking, wriggling, conceived, clean, content, smooth, crying, submissive, bumbling, sniveling}

A cursory glance at this list reveals a rich description of the stereotypical baby, one that incorporates pleasant and unpleasant behaviors in ample numbers. It makes little sense to reduce such a nuanced description to a single measure of lexical affect, or to parcel the description into separate senses, each with its own subset of behaviors. Instead, the partitioning of the description can be done on demand, and in context, to suit the speaker's meaning: if a term is used pejoratively, we focus on those qualities that are typically unpleasant (*sniveling, submissive, cranky, whimpering*, etc.); if the term is used affectionately, we focus instead on those that typically convey affection (*blessed, delicate, pure*, etc.); and so on. The affective rating of different qualities can be ascertained from any of the existing resources discussed earlier, with more or less success. Whissells DoA is perhaps the most limited, while Mohammad and Turneys eight-dimensional model of emotion [13] seems to possess the most nuance and power.

However, even basic properties and behaviors can be construed differently from one context to another. In some settings, for instance, *cunning* may be a positive description; in most others, it will likely be seen as negative. Many adjectival properties exhibit this duality of affect, such as *proud, tough, tame* and *fragile*, and

the description of the stereotypical baby above contains many that could be used to compliment in one context and to insult in another.

For this reason, we concentrate next on the construction of a nuanced model of behavioral interaction, in which the affective profile of a behavior or adjectival property (and thus of the entity that exhibits that property or behavior in context) changes in response to how it is used by the speaker. This model, which forms the second stage of the two-level affective lexicon outlined in the introduction, will allow us to see the positive in properties like *trusting, cunning* and *demanding*, and the negative in properties like *proud, unthreatening* and *innocent*, as the context demands.

13.3.2 Mutual Reinforcement Among Properties

In a representation as feature-rich as that for *baby* above, few features stand apart as truly unique. Some seem to mean much the same thing, while others form clusters of coherent, mutually-reinforcing properties and behaviors. Thus, *fat* reinforces *cuddly*, which reinforces *cute*, which reinforces *adorable*, which reinforces *lovable*, and so on. Intuitively, properties and behaviors that reinforce each other in this way are much more likely to share the same affective signature than those that clearly stand apart.

Yet to construct a support graph of mutually-reinforcing properties and behaviors, we need more than mere co-occurrence in the same stereotypical representation. We also need linguistic evidence to be certain of a link. Conveniently, this evidence can often be found in the Google n-grams, and if not there, then we may be forced to look for evidence on the open Web.

We begin by finding all Google 3-g of the form "ADJECTIVE and ADJECTIVE" or "BEHAVIOR and BEHAVIOR", such as "cuddly and cute" or "swaggering and strutting". We then consider the number of stereotypes that contain both terms in their representation. If this number is non-zero, a bidirectional link is added between both in the support graph. If the number is zero, we try one more test on the open Web; this test, though time-consuming, is well-motivated, since the n-gram data attests to the possibility of a relationship. We generate the *as*-bracketed query "as ADJECTIVE and ADJECTIVE as" and use Google to determine how many times this pattern occurs in similes on the Web. This pattern works only for adjectival properties, and should be attested by Web evidence only if both adjectives work well together in the description of the same target concept.

Once constructed in this way, every vertex in the resulting graph structure, which we denote N, represents a different property or behavior. The neighboring vertices of a property or behavior p – which we denote $N(p)$ – constitute a set of similar, mutually-reinforcing properties or behaviors that occur in one or more of the same affective contexts as p. For example, the vertex corresponding to the property *cunning* has the following neighbors in N:

{insidious, cruel, shrewd, devious, daring, audacious, evil, powerful, artful, clever, strategic, dangerous, charming, calculating, farsighted, strong, wary, subtle, manipulative, wise, conniving, convincing, pragmatic, quick, fast, experienced, diabolical, mighty, greedy, swift, articulate, avaricious, determined, patient, canny, vicious, detailed, curious, deadly, resourceful, resilient, intelligent, cool, treacherous, beautiful, brutal, skilled, bloodthirsty, resolute, wicked, poisonous, dastardly, dishonest, deceitful, sexy, unfeeling, sneaky, mean, sly, smart, agile, bold, aggressive, graceful, deceptive, ingenious, insightful, selfish, unprincipled, inventive, shameless, good, secretive, careful, neurotic, heartless, despicable, brave, convoluted, slimy, sophisticated, exploitative, vindictive, disloyal, fluid, machiavellian, towering, brilliant, keen, violent, feared, suspicious, sinister, energetic, scheming, savage, merciless, cowardly, silent, tricky, astute, witty, nasty, free, pretty, lucid, unscrupulous, evocative, precise, seductive, cheating, nimble, versatile, malicious, courageous, virulent, playful, cautious, skillful, untrustworthy, uncaring, amoral, unmerciful, coarse, underhanded, spry, awesome, original, angry, devilish, vile, duplicitous, venomous, obnoxious, bland, fantastic, reclusive, cynical, shifty, stunning, relentless, crazy, funny, wry, loyal, reliable, twisted, effective, prepared, capable, dexterous, adroit, methodical, beguiling}

Guided by our intuition that the affective profile of p should be heavily influenced by the affect of its neighbors, we now consider whether the affect of p can be reliably estimated as a function of $N(p)$.

13.4 Estimating Lexical Affect

Since every edge in N represents an affective context, we can estimate the likelihood that a property p is ever used in a positive or negative context if we know the positive or negative affect of enough members of $N(p)$. Thus, if we label enough vertices of N with $+$ or $-$ labels, we can interpolate a positive/negative affect score for all vertices p in N.

To do this, we build a reference set $-R$ of typically negative words, and a set $+R$ of typically positive words. Given a few seed members of $-R$ (such as *sad, disgusting, evil*, etc.) and a few seed members of $+R$ (such as *happy, wonderful, pretty*, etc.), we easily find many other candidates to add to $+R$ and $-R$ by considering neighbors of these seeds in N. Veale [21] shows how large ad-hoc word-categories like these can quickly be constructed using flexible pattern-matching over the Google n-grams. After just three iterations in this fashion, we populate $+R$ and $-R$ with approx. 2,000 words each.

For a property or behavior p can now define $N^+(p)$ and $N^-(p)$ as follows:

$$N^+(p) = N(p) \bigcap +R \tag{13.1}$$

For example, $N^+(\textit{cunning})$ denotes the following set of properties and behaviors:

{shrewd, powerful, strong, subtle, wise, quick, mighty, articulate, intelligent, beautiful, daring, experienced, patient, fast, curious, cool, swift, detailed, skilled, resolute, witty, free, artful, careful, agile, brave, cute, canny, graceful, sophisticated, versatile, inventive, spry, fun, precise, bold, resourceful, keen, courageous, playful, determined, stunning, smart,

seductive, astute, clever, strategic, towering, charming, ingenious, skillful, insightful, intricate, reliable, good, pretty, farsighted, nimble, pragmatic, lucid, brilliant, loyal, evocative, adroit, resilient, audacious, effective, awesome, capable, sexy, convincing, funny, fantastic, dexterous, methodical, beguiling, original, prepared, fluid, energetic, wry}

$$N^{-}(p) = N(p) \bigcap -R \tag{13.2}$$

Thus, $N^{-}(cunning)$ denotes the following set of properties and behaviors:

{insidious, cruel, devious, evil, dangerous, calculating, wary, manipulative, conniving, diabolical, greedy, avaricious, vicious, deadly, treacherous, brutal, bloodthirsty, wicked, poisonous, dastardly, dishonest, deceitful, unfeeling, sneaky, mean, sly, aggressive, deceptive, selfish, unprincipled, shameless, secretive, neurotic, heartless, despicable, convoluted, slimy, exploitative, vindictive, disloyal, machiavellian, violent, feared, suspicious, sinister, scheming, savage, merciless, cowardly, silent, tricky, nasty, unscrupulous, cheating, malicious, virulent, cautious, untrustworthy, uncaring, amoral, unmerciful, coarse, underhanded, angry, devilish, vile, duplicitous, venomous, obnoxious, bland, reclusive, cynical, shifty, relentless, crazy, twisted}

That is, $N^{+}(p)$ is the set of neighbors of p that are known to be positive, and $N^{-}(p)$ is the set of neighbors of p that are known to be negative. We can now assign positive and negative scores to each vertex p in N by interpolating from the reference values in $+R$ and $-R$ to their neighbors in N:

$$pos(p) = \frac{|N^{+}(p)|}{|N^{+}(p) \bigcup N^{-}(p)|} \tag{13.3}$$

$$neg(p) = \frac{|N^{-}(p)|}{|N^{+}(p) \bigcup N^{-}(p)|} \tag{13.4}$$

For instance, the set $N^{-}(aggressive)$ contains 230 elements, while $N^{+}(aggressive)$ contains 201 elements. Thus, $pos(aggressive)$ is calculated to be 0.466 while $neg(aggressive)$ is calculated to be 0.534. In other words, *aggressive* is deemed to be more positive than negative, or to be more precise, (13.3) and (13.4) estimate that *aggressive* is more likely to occur in a negative descriptive context than in a positive descriptive context. In contrast, *cunning* is deemed to be slightly more positive than negative, given the number of positive and negative descriptive contexts that are captured in $N^{-}(cunning)$ and $N^{+}(cunning)$ as shown earlier. The properties *aggressive* and *cunning* are borderline cases, since each evokes a large number of descriptive contexts in which each could be viewed either positively or negatively. A property like *cynical*, however, is much more clear-cut: with 258 neighbors in $N^{-}(cynical)$ and just 38 in $N^{+}(cynical)$, $neg(cynical)$ is 0.87 while $pos(cynical)$ is just 0.13.

If a term S denotes a stereotypical idea and is described via a set of typical properties and behaviors $typical(S)$ in the lexicon, then:

$$pos(S) = \frac{\sum_{p \in typical(S)} pos(p)}{|typical(S)|} \tag{13.5}$$

$$neg(S) = \frac{\sum_{p \in typical(S)} neg(p)}{|typical(S)|} \tag{13.6}$$

Thus, (13.5) and (13.6) calculate the mean affect of the properties and behaviors of S, as represented via $typical(S)$. We can now use (13.3) and (13.4) to separate $typical(S)$ into those qualities that are more negative than positive (putting a negative spin on S) and into those that are more positive than negative (putting a positive spin on S):

$$posTypical(S) = \{p | p \in typical(S) \wedge pos(p) > neg(p)\} \tag{13.7}$$

$$negTypical(S) = \{p | p \in typical(S) \wedge neg(p) > pos(p)\} \tag{13.8}$$

Formulae (13.7) and (13.8) can be used to "spin" a concept positively or negatively in given context, to highlight only those qualities of S that support the chosen positive or negative viewpoint on S. For instance, the stereotype *terrorist* has the following salient positive properties (numbers in parentheses indicate the value assigned by $pos(p)$ for each property p):

posTypical(*Terrorist*) = {*committed(.826), daring(.733), networked(.8), sponsored(.833)*}

As we should expect, there are many more negative properties that are salient for the stereotype *terrorist* (numbers in parentheses indicate the value assigned by $neg(p)$ for each property p):

negTypical(*Terrorist*) =
{hateful(.978), bad(.951), despicable(.98), harmful(.95), inhuman(.972), irrational(.916), odious(.97), horrid(.97), irresponsible(.916), depraved(.945), murdering(1.0), heinous(.951), hostile(.92), guilty(.954), misguided(.92), damaging(.89), bloodthirsty(.93), suspicious(.94), bigoted(.952), hated(.962), sickening(.969), callous(.928), raging(.917), appalling(.906), vicious(.86), deranged(.93), barbarous(.93), mindless(.866), unscrupulous(.89), threatening(.826), indiscriminate(.96), demonic(.98), wicked(.83), convicted(.96), destructive(.817), condemned(.97), pitiless(.9), crazed(.845), twisted(.815), alarming(.844), insidious(.84), merciless(.82), accused(.978), sinister(.79), dreadful(.89), diabolical(.85), devastating(.756), remorseless(.875), brainwashed(.893), shocking(.74), ruthless(.733), infamous(.828), menacing(.743), unforgiving(.768), evil(.904), hunted(.93), dreaded(.83), hardened(.764), disgruntled(.954), suspected(.925), fearsome(.712), armed(.685), imprisoned(.97), chilling(.638), prohibited(1.0), hating(1.0), criminal(.968), lethal(.628), wanted(.72), clandestine(.758), incendiary(.688), inflamed(.826), arrested(.961), captured(.888), masked(.78), feared(.627), shooting(.639), killing(.95), branded(.77), hooded(.916), banned(1.0), fanatic(.7), jailed(1.0), concealed(.68), targeted(.7), bombing(.961), fighting(.66), radical(.53), proscribed(.857)}

Note how properties such as *armed*, *shooting*, *fighting* and even *feared* have lower negativity than unremittingly negative qualities like *murdering* and *prohibited*. These lower scores reflect the greater possibilities for using these properties to impart a positive view of a topic, as when e.g. one desires to be *feared* and *respected*.

We estimate a positive and negative affect score for each stereotype (using (13.5) and (13.6)) and for each of their properties and behaviors (using (13.3) and (13.4)), which produces an affect lexicon of over 16,000 words. For instance, $pos(Terrorist) = 0.178$ and $neg(Terrorist) = 0.822$. Overall, the mean positivity score is 0.517 (standard deviation = 0.313), while the mean negativity score is 0.483. In contrast, the mean positivity of the 1,977 words in $+R$ is 0.852 (standard dev. = 0.127), while the mean negativity of the 2,192 words in $-R$ is 0.813 (standard dev. = 0.154).

13.5 In the Mood for Affective Search

Thus far we have focused on a rather reductive view of affect as the potential of words to convey a positive or negative meaning. As shown in [13, 14, 16] other emotional dimensions can meaningfully be used to describe our affective perception of a word. Those authors show e.g., that some words convey sadness and fear to different degrees, while others suggest a degree of joy and even trust. While we do not explicitly distinguish different dimensions of mood or emotionality in the two-level model, the model does capture, through its network N of mood-bearing words, the emotional influence that the perception of one kind of property or behavior can have on the perception of other properties and behaviors. The effect is often called the *halo effect* in the psychological literature, wherein the perception of one positive quality, such as physical beauty or strength, can influence our perception of other qualities such as intelligence, honesty and leadership [3]. Conversely, the network structure of N also supports reasoning under the so-called *devil effect*, wherein the perception of one negative quality (such as $angry$) can lead us to view the possession of related qualities (such as *aggressive*) more negatively [15]. As such, the two-level model implicitly supports a whole lexicon of diverse but interconnected mood types: every node in N evokes a halo of associated positive nodes, and a penumbra of associated negative nodes as well.

Consider that a word like *aggressive* implies a range of positive qualities that are captured in $N^+(aggressive)$ and a range of negative qualities that are captured in $N^-(aggressive)$. The halo of words in $N^+(aggressive)$ helps to convey the up-side of aggressive behavior (e.g. to be *aggressive* often implies that one is also *quick*, *energetic*, *vigorous* and *determined*) while the penumbra of words in $N^-(aggressive)$ evokes the down-side of aggressiveness (e.g. *aggressive* people are often *violent*, *angry*, *hostile* and *abusive*). We can, in effect, allude to a whole family of affective words with a single term like *aggressive*, or we can focus exclusively on wholly positive or negative word halos with polarizing labels such as $+aggressive$ and $-aggressive$. There is little need then to build aggressiveness or other moods into the lexicon as explicit dimensions of affect if we can use these labels to evoke the same word sets. Any one of 1,000s of different words in the lexicon – or a combination thereof – can be used as mood filters to refer to ad-hoc families of affective words as the need arises.

Stereotypes themselves can also be used as powerful and expressive mood filters in the affective retrieval and ranking of documents. For instance, we can use the mood filter $+$*leader* to rank documents by their relative density of words that convey positive leadership qualities, or $-$*terrorist* to rank documents by their use of words that convey the many negative qualities of terrorists (which are enumerated above).

Pursuing this theme, we are now using the two-level lexicon to support affective text search over news content on the Web. In this application, users may use $+$*aggressive*, or $-$*sad*, or any property, behavior or stereotype (e.g., $+$*genius*, $-$*terrorist*) they consider apt, as mood filters to organize the retrieved document set. Currently, news articles are crawled from a dozen news sites (this number will grow) and their textual content is indexed using the *Lucene* system [11]. To allow for efficient document-level affect determination, any words that occur in the affect lexicon are stored in a separate document field. Queries to the system are separated into two kinds of query terms: regular query terms, which are unadorned words or phrases; and mood filters, which are terms prefixed either with $+$ or $-$ to indicate their affective polarity (such as $-$*proud* or $+$*exciting*). Regular query terms are used to retrieve matching documents from the indexing engine, before the mood filters are used to rank these documents by mood. The retrieved documents may be ranked by their relevance to the regular query term (e.g. as calculated by *Lucene*) or by their mood density (the proportion of words in a document that match the mood filter of the query), or by a weighted combination of both measures.

13.6 Empirical Evaluation

We shall now take a closer look at the affect scores for properties and behaviors in Sect. 13.6.1, before considering the scores estimated for stereotypes in Sect. 13.6.2. We then evaluate the performance of the affect lexicon on an affective separation task – in which the properties and behaviors of stereotypes in the reference set are separated into distinct positive and negative subsets – in Sect. 13.6.3.

13.6.1 Bottom Level: Properties and Behaviors of Stereotypes

If the intuition behind formulae (13.1)–(13.4) is valid, then we should expect that for every property or behavior p in $+R$, $pos(p) > neg(p)$, and conversely, $neg(p) > pos(p)$ for every p in $-R$. Recall that with (13.3) and (13.4) we model the problem of estimating affect as an interpolation task rather than as a learning task. Therefore, the *pos* and *neg* affect scores that are calculated for p are independent of whether or not p is in $+R$ or $-R$. So if we add p to a reference set, or remove p from a reference set, the same values for $pos(p)$ or $neg(p)$ will be estimated. Only the neighboring vertices of p can possibly be influenced by such a move, and the affect scores that are calculated for those neighbors cannot feed back into the scores

calculated for p. It is thus reasonable to evaluate the intuition behind (13.1)–(13.4) using $+R$ and $-R$ as a gold standard.

When affect scores are calculated for the complete set of properties, behaviors and stereotypes in the lexicon, just five properties in $+R$ are given a positivity score of less than 0.5, leading those words to be wrongly classified as more negative than positive. The misclassified properties/behaviors are: *evanescent, giggling, licking, devotional*, and *fraternal*. These five words account for approx. 0.4 % of the 1,314 adjectival properties in $+R$.

At the same time, 26 properties in $-R$ are given a negativity score of less than 0.5, leading those words to be wrongly classified as more positive than negative. The misclassified properties/behaviors are: *cocky, dense, demanding, urgent, acute, unavoidable, critical, startling, gaudy, decadent, biting, controversial, peculiar, disinterested, strict, visceral, feared, opinionated, humbling, subdued, impetuous, shooting, acerbic, heartrending, ineluctable*, and *groveling*. These 26 words account for approx. 1.9 % of the 1,385 adjectival properties in $-R$.

Though these results are not very surprising– after all, the elements of $+R$ and $-R$ were chosen to have an obviously positive or negative affect – they do validate the intuition in (13.1)–(13.4) that the affect of a property or behavior can be consistently estimated as a function of the other properties and behaviors with which it is used to form a coherent description.

13.6.2 Top Level: Stereotypical Concepts

The reference sets $+R$ and $-R$ also contain a significant number of nouns. The positive reference set $+R$ contains 478 nouns for which the stereotype lexicon provides a feature-level description, while $-R$ contains 677 nouns that are associated with specific stereotypical properties and behaviors. We can thus use these reference cases to evaluate the mean affect scores estimated for stereotypes in (13.7) and (13.8). If it is indeed sensible to average the positive and negative scores of the elements in $typical(S)$ to estimate a positive and negative score for a stereotype S, then we should observe $pos(S) > neg(S)$ for almost all stereotypes S in $+R$, and $neg(S) > pos(S)$ for almost all stereotypes S in $-R$.

When affect scores are calculated for the complete set of properties, behaviors and stereotypes in the lexicon, just 16 positive stereotypes in $+R$ are assigned a positivity of less than 0.5, leading these stereotypes to be classified as more negative than positive. The misclassified stereotypes are: *patient, innocent, stable, rustic, giant, desire, expectation, heart, responsibility, sentiment, infant, toddler, fruitcake, giggle, sitcom*, and *granny*. These 16 stereotypes account for approx. 3.3 % of the 478 stereotypes in $+R$.

At the same time, just 26 negative stereotypes in $-R$ are assigned a negativity of less than 0.5, leading these to be classified as more positive than negative. The misclassified stereotypes are: *penitent, fire, regret, trial, opposition, accomplice, revenge, rebellion, enmity, debt, illusion, protest, drill, hide, wetland, dogma,*

Table 13.1 Average P/R/F1 scores for the retrieval of pos. and neg. features from 6,230 stereotypes

Macro average	Positive properties	Negative properties
Precision	0.962	0.98
Recall	0.975	0.958
F-score	0.968	0.968

disregard, revolt, jihad, handgun, grenade, sorceress, grudge, inquisition, duel and *colonoscopy*. These 26 stereotypes account for approx. 3.8 % of the 677 stereotypes in $-R$.

These results validate the guiding intuition in (13.5) and (13.6), namely, that the overall affect of a stereotype S (in a null context) can be reliably estimated as a function of the affect of its most typical properties and behaviors as represented by $typical(S)$.

13.6.3 Separating Words by Affect: Two Views

The reference sets $+R$ or $-R$ contain many of the properties and behaviors that are ascribed to each stereotype S in the stereotype lexicon, via $typical(S)$. We can thus use $+R$ and $-R$ as a gold standard for evaluating the separation of the properties and behaviors of a stereotype S into distinctly positive and negative subsets of $typical(S)$, denoted $posTypical(S)$ and $negTypical(S)$ in formulae (13.7) and (13.8).

The stereotype lexicon contains 6,230 stereotypes with at least one property or behavior in $-R \bigcup +R$, and on average, $-R \bigcup +R$ contains 6.51 of the properties and behaviors of each of these stereotypes (on average, 2.95 are in $+R$, 3.56 are in $-R$).

In a perfect separation we should obtain a positive subset that contains only those properties and behaviors in $typical(S) \bigcap +R$ and a negative subset that contains only those in $typical(S) \bigcap -R$. Viewing the problem as a retrieval task, whose goal is the accurate retrieval of distinct positive and negative subsets of $typical(S)$ for a given stereotype S using (13.7) and (13.8), we report the P/R/F1 results of Table 13.1. Note that the reported results are calculated as the macro-average of P/R/F1 scores for the separation process applied to the properties and behaviors of all 6,230 stereotypes in the experiment.

In a complementary formulation of this problem, we must separate the list of stereotypes that exhibit a given property or behavior p into two distinct sets, the set of positive stereotypes that exhibit p and the set of negative stereotypes that exhibit p. The stereotype lexicon contains 4,536 properties and behaviors for which one or more of its associated stereotypes is in $-R \bigcap +R$. On average, each of these properties or behaviors is associated with 5.29 stereotypes in $-R \bigcap +R$ (on average, 2.06 are in $+R$ while 3.23 are in $-R$). Again viewing the problem as a retrieval task, of stereotypes rather than properties and behaviors, we report the

Table 13.2 Average P/R/F1 scores for the retrieval of pos. and neg. stereotypes for 4,536 features

Macro average	Positive stereotypes	Negative stereotypes
Precision	0.986	0.965
Recall	0.949	0.982
F-score	0.967	0.973

results of Table 13.2. Note that the reported results are calculated as the macro-average of P/R/F1 scores for the separation process applied to the stereotypes associated with all 4,536 properties and behaviors in the experiment.

As can be seen in Tables 13.1 and 13.2, the current model achieves very encouraging results, both on the property/behavior separation task and on the stereotype separation task. These tasks serve more than a purely evaluative function. The former (property/behavior separation) is performed whenever we wish to place a particular affective spin on a topic, such as when we use the words "baby" and "youngster" affectionately, or use the words "elite" and "professor" disapprovingly. The latter (stereotype separation) is often performed as part of the interpretation of a feature ascription : if someone describes you as "strange", what might they be implicitly calling you if "strange" is meant negatively (a *weirdo*, or a *freak*, perhaps?), and what might "strange" describe if generously given the most positive interpretation (an *eccentric* or a *rarity*, perhaps)?

13.7 Conclusions

The chief innovation in this work is the imposition of a two-level structure onto the affect lexicon: the first level represents stereotypes as bundles of their most salient properties and behaviors; the second represents the relationship of these properties and behaviors to each other. The result is a lexicon that incorporates a great deal of common-sense knowledge of the world, for it is only through common-sense that a language user can fully appreciate the variability of a word's affect from one context to another (see [10] for an expression of the same view).

The approach is a modular one, and one could in principle use any of the existing affect lexica to assign affect scores to the properties and behaviors of the second level. Nonetheless, we have shown that good results are achievable with the simple formulae in (13.3) and (13.4), and that these results are a good basis for estimating the affect of higher-level stereotypes in (13.5) and (13.6). In this first phase of the work, we have concentrated on building a stereotype lexicon in which each of the typical properties and behaviors ascribed to a word-concept are both justifiable and salient. That is, we have aimed for precision rather than recall in this foundation-building phase of development. However, coverage is also an important dimension of a stereotype lexicon. While we have shown that the two-level lexicon contains enough property ascriptions to ensure that the average overall affect of a stereotype

S can be reliably estimated from the set *typical*(S), we also need to quantify the likelihood that *typical*(S) will contain any property or behavior that is commonly and consensually held to be salient of S. This remains a challenging goal of the work, especially given the lack of a gold standard against which the coverage of a stereotype lexicon can be quantified. For now, the *halo effect* (or conversely, the *devil effect*) can be used to infer the salience of an arbitrary property p to a stereotype S given the contents of *typical*(S), since the relevance of p will be a function of $typical(S) \bigcap N(p)$. Interested readers who wish to exploit this early form of the lexicon can do so by contacting the author directly. As demonstrable improvements are made in the coverage of the lexicon, versions will be made publicly available for research purposes.

We have also demonstrated how a two-level affect lexicon might be used to understand how one topic can be viewed through the affective lens of another, as e.g. when we view *science as a religion*, *art as a science*, or *a leader* as a *pioneer* or a *tyrant*. In addition, the lexicon allows the users of an affective retrieval system to personalize their affective relationship to the words in a text or a query. For instance, a user can use $+$*cunning* or $-$*powerful* to specify that *cunning* should be viewed as a positive quality and *powerful* as a negative quality in the current retrieval context. The two-level lexicon represents a lightweight form of commonsense knowledge, albeit commonsense that is not explicitly axiomatized. Nonetheless, the affective and stereotypical dimensions of the two-level lexicon can always be used to enrich a more formal, axiomatized ontology like DOLCE (which, for instance, contains a rich hierarchy of social roles) with the kind of non-axiomatic pragmatic knowledge that one needs to reason effectively in social contexts.

Beyond the obvious applications of a fine-grained affective lexicon for classifying the polarity of texts – such as the texts retrieved using a Web search engine – the stereotypical perspective can also be used to find documents that exhibit a particular conceptual slant on a given topic (e.g. to retrieve documents that positively view *Apple as a cult*). In this sense our search engines really would support a form of a creative information retrieval (as defined in [21] and exploited in [22]), and allow us to see the best and worst in everything on the Web.

Acknowledgements This research was supported by the WCU (World Class University) program under the National Research Foundation of Korea, and funded by the Ministry of Education, Science and Technology of Korea (Project No: R31-30007).

References

1. Bolinger, D.: Ataxis. In: Rokko Linguistic Society (ed.), Gendai no Gengo Kenkyu (Linguistics Today), pp. 1–17. Kinseido, Tokyo (1988)
2. Brants, T., Franz, A.: Web 1T 5-gram Version 1. Linguistic Data Consortium, Philadelphia (2006)
3. Dion, K., Berscheid, E., Walster, E.: What is beautiful is good. J. Personal. Soc. Psychol. **3**(24), 285–290 (1972)

4. Ekman, P.: Facial expression of emotion. Am. Psychol. **48**, 384–392 (1993)
5. Esuli, A., Sebastiani, F.: SentiWordNet: a publicly available lexical resource for opinion mining. In: Proceedings of LREC-2006, the 5th Conference on Language Resources and Evaluation, Genoa, pp. 417–422 (2006)
6. Esuli, A., Sebastiani, F.: PageRanking WordNet synsets: an application to opinion mining. In: Proceedings of ACL-2007, the 45th Annual Meeting of the Association for Computational Linguistics. ACL, Prague, Czech Republic (2007)
7. Fellbaum, C. (ed.): WordNet: an electronic lexical database. MIT, Cambridge (1998)
8. Gangemi, A., Guarino, N., Masolo, C., Oltamari, A., Schneider, L.: Sweetening ontologies with DOLCE. In: Proceeddings of EKAW 2002, the 13th International Conference on Knowledge Engineering and Knowledge Management. Springer, London (2002)
9. Kilgarriff, A.: Googleology is bad science. Comput. Linguist. **33**(1), 147–151 (2007)
10. 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. ACM, New York (2003)
11. McCandless, M., Hatcher, E., Gospodnetić, O.: Lucene in Action, 2nd edn. Manning, Greenwich (2010)
12. Mihalcea, R., Tarau, P.: TextRank: bringing order to texts. In: Proceedings of EMNLP-04, the 2004 Conference on Empirical Methods in Natural Language Processing. ACL, Barcelona, Spain (2004)
13. Mohammad, S.F., Turney, P.D.: Emotions evoked by common words and phrases: using mechanical turk to create an emotional lexicon. In: Proceedings of the NAACL-HLT 2010 workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Los Angeles (2010)
14. Mohammad, S.F., Yang, T.W.: Tracking sentiment in mail: how genders differ on emotional axes. In: Proceedings of the ACL 2011 WASSA workshop on Computational Approaches to Subjectivity and Sentiment Analysis, Portland, Oregon, USA (2011)
15. Nisbett, R.E., Wilson, T.D.: The halo effect: evidence for unconscious alteration of judgments. J. Personal. Soc. Psychol. (American Psychological Association) **35**(4), 250–256 (1977)
16. Plutchik, R.: A general psycho-evolutionary theory of emotion. Emot. Theory Res. Exp. **2**(1–2), 1–135 (1980)
17. Singh, P.: The public acquisition of commonsense knowledge. In: Proceedings of AAAI Spring Symposium on Acquiring (and Using) Linguistic (and World) Knowledge for Information Access, Palo Alto (2002)
18. Strapparava, C., Valitutti, A.: Wordnet-affect: an affective extension of Wordnet. In: Proceedings of LREC-2004, the 4th International Conference on Language Resources and Evaluation, Lisbon (2004)
19. Turney, P.D.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of ACL-2002, the 40th Annual Meeting of the Association for Computational Linguistics, pp. 417–424. Morgan Kaufmann, San Francisco (2002)
20. Veale, T., Hao, Y.: Making lexical ontologies functional and context-sensitive. In: Proceedings of ACL-2007, the 45th Annual Meeting of the Association of Computational Linguistics, pp. 57–64. ACL, Stroudsburg (2007)
21. Veale, T.: Creative language retrieval: a robust hybrid of information retrieval and linguistic creativity. In: Proceedings of ACL-2011, the 49th Annual Meeting of the Association of Computational Linguistics. ACL, Portland (2011)
22. Veale, T., Hao, Y.: In the mood for affective search with web stereotypes. In: Proceedings of WWW-2012, the World Wide Web conference (demonstration track), Lyon (2012)
23. Whissell, C.: The dictionary of affect in language. In Plutchik, R., Kellerman, H. (eds.) Emotion: Theory and Research, pp. 113–131. Harcourt Brace, New York, USA (1989)

Index

A. Oltramari et al. (eds.), *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-31782-8, © Springer-Verlag Berlin Heidelberg 2013