The Combined Usage of Ontologies and Corpus Statistics in Information Retrieval

Thomas Vestskov Terney

AUGUST 2009 ROSKILDE UNIVERSITY COMPUTER SCIENCE RESEARCH REPORT #126

Thomas Vestskov Terney

Computer Science
Department of Communication,
Business and Information Technologies
Roskilde University
P. O. Box 260
DK–4000 Roskilde
Denmark

Telephone: +45 4674 3839
Telefax: +45 4674 3072
Internet: http://www.ruc.dk/dat_en/
E-mail: datalogi@ruc.dk

ISSN 0109–9779

Research reports are available electronically from:

http://www.ruc.dk/dat_en/research/reports/

The Combined Usage of Ontologies and Corpus Statistics in Information Retrieval

Thomas Vestskov Terney

A Dissertation Presented to the Faculties of Roskilde University in Partial Fulfillment of the Requirement for the Degree of Doctor of Philosophy

Department of Communication, Business and Information Technologies
Roskilde University, Denmark
August 2009

Abstract

This dissertation explores the question of how corpus statistics can be combined with ontological knowledge in information retrieval. The motivation for delving into this question lies in the intriguing possibilities the two different frameworks of semantic analysis and descriptive statistics offer individually in information retrieval. Using corpus statistics, the specific document collection at hand can be described and this description can be used for matching users' information needs. Ontologies, on the other hand, offer a semantic analysis based on world knowledge simply beyond the reach of statistical analysis.

Information retrieval can be divided into three general processes: The indexing of documents and queries, the matching of documents and queries, and, finally, the presentation of the result of the match. A single dominant ontology-based information retrieval model that covers all three processes does not yet exist; and this dissertation does not attempt to present one. Rather, the aim is to present a number of improvements in the three processes that can be integrated into already existing ontology-based or keyword-based information retrieval systems.

With respect to indexing, a preliminary machine learning approach for the analysis of the relations denoted by prepositions is presented aimed at improving the extraction of conceptual knowledge from a text. An extension of the vector space model that makes it possible to expand the index by means of an ontology is also presented. With regard to matching, arguments are presented showing how ontology-based similarity measures, to a larger extent, could incorporate a statistical element that modifies the similarity measure to reflect the document collection at hand. Finally, with an emphasis on the presentation of search results, two ontology-based clustering approaches are presented as means for navigating through a set of documents.

Resumé (in danish)

Denne afhandling udforsker spørgsmålet om, hvordan statistisk analyse af en dokumentsamling kan kombineres med ontologisk viden indenfor søgning. Motivationen for at dykke ned i dette spørgsmål udspringer af de besnærende muligheder, som semantisk analyse og statistisk analyse tilbyder hver for sig indenfor søgning. Ved brug af statistiske metoder kan en given dokumentsamling blive analyseret og brugt til at matche brugernes informationsbehov. På den anden side tilbyder ontologier en semantisk analyse baseret på omverdensviden, som simpelthen er udenfor rækkevidde ved anvendelse af statistisk analyse alene.

Søgning kan opdeles i tre generelle processer: Indekseringen af dokumenter og forespørgsler, matchning af dokumenter og forespørgsler og endelig præsentationen af resultatet af matchet. Der eksisterer ikke en dominerende model for ontologibaseret søgning, og denne afhandling forsøger ikke at præsentere en. Istedet er målet at præsentere en række forbedringer i de tre processer, som kan integreres i allerede eksisterende ontologibaserede eller nøgleordsbaserede søgesystemer.

I forhold til indeksering præsenteres en præliminær maskinindlæringstilgang til analyse af relationer udtrykt af præpositioner. Formålet er at forbedre udtrækningen af konceptuel viden fra tekst. En udvidelse af vektorrumsmodellen præsenteres også. Denne gør det muligt at ekspandere indekset ved hjælp af en ontologi. I relation til matchning fremføres en række argumenter, som viser, hvordan ontologibaserede similaritetsmål i højere grad kan inkorporere et statistisk element, der modificerer similaritetsmålet til at afspejle den givne dokumentsamling. Med særlig fokus på præsentationen af søgeresultater præsenteres til slut to ontologibaserede grupperingsmetoder som tilgange til at navigere gennem et sæt af dokumenter.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The company *Kids Arts* has produced a small package of toys consisting of *dough*, an *oven*, and kitchen utensils. Using several present-day search engines, finding this product in any store on the Internet using a query like, "kids arts dough oven" is almost impossible. Why is this the case? The answer is simple: Keyword based search; we are searching for a page containing the words in the query using simple string comparison. In the query *Kids Arts* is thus not in the normal sense of artistic masterpieces produced by kids, but rather it is in the sense of the company which produces goods that kids can use in their endeavor to develop artistic skills. *Oven* is used in the sense of a toy, and not a real oven for baking *dough*, which, by the way, is synonymous in this case with *clay* and *plasticine*. The problem with the words *kids*, *arts*, *dough*, and *oven* in connection with a simple keyword based information retrieval approach is that the meaning of the words and the relations between are ignored. Furthermore, the shear abundance of web pages with these words makes it impossible to browse through every document. If the search engine ranks relevant documents far down on the list, the search has brought us no closer to our goal.

Keyword based information retrieval is based on the assumption that we can capture, or to a large extent approximate, the semantic content of both queries and documents simply by looking at the lexical level of the text. In other words, the matching of documents and queries are performed on the surface level of the texts, so a search for, e.g. *hormone* will not match *insulin* or *thymosin* even though they are indeed both hormones. The prevalent use of search engines like Google, Yahoo, and MSN Live shows that in many cases the assumption behind keyword based search is unproblematic. But, as the hormone and toy example illustrate, in some cases the assumption leads to low quality results.

The topic of this dissertation is ontology-based information retrieval as an alternative or a supplement to keyword based information retrieval. In popular terms, an ontology is a taxonomic ordering of concepts, e.g. *insulin* is a special kind of *hormone*. Ontologies are interesting because they offer the possibility of introducing semantics at several of the processes involved in information retrieval. In the

initial content analysis of the text, ontologies can be used to ensure that sentences paraphrased differently but with identical conceptual content are indexed similarly so that a query for, e.g. *fast memory* will also match *high speed ram*. Moreover, ontologies enable a semantic expansion of the query with related concepts based on how similar they are to the concepts in the original query. For instance, *ram* is more closely related to *memory* than, e.g. *computer components*.

One of the challenges in ontology-based information retrieval is that available ontologies is a scarce resource for many languages and tasks. In addition, engineering an ontology that captures all the concepts and relations that can be expressed by natural language is impossible. Despite this challenge, ontology-based information retrieval offers a semantic matching which is not possible with information retrieval relying solely on a keyword based match. The schism between the possibilities offered by ontologies and the challenges in modeling domains and applying strictly ontology-based systems calls for research in the area of integrating ontology-based approaches with approaches focused on the lexical level of keywords. The research presented here has sought to explore how such approaches can be integrated in information retrieval.

1.1 Research Question

The purpose of the research presented in this dissertation was to explore the question:

> *How can corpus statistics be combined with ontological knowledge in information retrieval?*

Important sub questions to be examined in the study included:

1. *How can corpus staticstics improve ontology-based content analysis?*

2. *How can corpus statistics and ontologies be combined in semantic similarity measures?*

3. *How can term weighting take into account both the ontology and the frequency of the concepts in the documents?*

4. *How can corpus statistics and ontologies be used in the presentation of search results?*

Corpus statistics are here to be understood in a broad sense relating to descriptive statistical analysis of a document collection.

Given the wide ranging nature of the research, the aim has not been to excavate all possible solutions to the questions posed, but rather to present possible directions to pursue.

1.2 Outline

This dissertation is divided into a foundations part and a contributions part. Chapter two, three, and four are on the foundations of the work presented here, and chapters five, six, seven and eight are on the contributions. In this section the content of the chapters in these two parts will be outlined.

The **second chapter** will contain an introduction to the field of information retrieval. The chapter starts out with a description of the major processes in information retrieval systems and tries to give an intuitive as well as a more formal understanding of why some words are more interesting than others from an information retrieval perspective. The chapter will present the Boolean Model, the Vector Space Model and the Probabilistic Model, which are the most prominent models of information retrieval systems used today. In addition, an extension of the Boolean Model, the Fuzzy Model, will be presented since it provides an intuitive and an easy way of integrating ontologies in the retrieval process. The purpose of the chapter is to serve both as a presentation of the theoretical basis of the research presented in this dissertation and to give the reader an understanding of the sequencing of the chapters in the rest of the thesis. The chapter will also indicate why ontology-based information retrieval offer a valuable alternative to keyword based information retrieval.

Ontologies are introduced in the **third chapter**, which describes what an ontology is, what the different types of ontologies are, and how ontologies are represented. Special attention is given to the lattice algebraic language ONTOLOG, which serves as the formal framework for specifying ontologies in all subsequent chapters. The chapter ends with a presentation of examples of ontological resources. Like the previous chapter, the aim is to present the underpinning theoretical framework of this dissertation.

The **fourth chapter** is devoted to the ONTOQUERY project which the work presented here is a part of. The chapter will focus on content analysis; namely on how we get from running text to an ontological representation that can be used as an index later in the retrieval process. Also the work on semantic similarity done within the ONTOQUERY project will be described. These two aspects of the ONTOQUERY project are described because they are of particular relevance for the work presented here.

The contributions are presented in the subsequent four chapters starting with **chapter five**. This chapters presents an experiment with the semantic analysis of content; more specifically, how machine learning can be used to investigate an assumption of affinity between the relation denoted by a preposition and the concepts surrounding it. The chapter presents the nature of semantic relations, the effort involved in compiling the used corpus, an introduction to the basics of machine learning, and, naturally, the results of the experiments. The presented approach enables a more accurate ontological indexing than previously has been attempted.

The issue of similarity treated in chapter four on the ONTOQUERY project paves

the way for moving to a more thorough investigation of how to compute similarity between concepts. This is the topic of the **sixth chapter**. The first part of the chapter gives an introduction to ontology-based similarity measures and similarity measures based on distributional patterns of the co-occurrence of concepts. The second part of the chapter introduces the novel idea of measuring the similarty between two concepts by these two kinds of measures.

Chapter seven introduces an original approach to index expansion as an alternative to the term weighting applied in the Vector Space Model. A the heart of the model is an approach to term weighting based on the inclusion of an ontology. The chapter starts out with presenting some fundamental issues of query expansion.

Having presented the steps in ontology-based information retrieval from content analysis, to indexing and similarity measures, **chapter eight** presents two different approaches to conceptual summaries. A conceptual summary is as a way of presenting the conceptual content of the result of a query, i.e. a summary of the result set based on the ontology and a clustering of the concepts appearing in the set of results. The first approach, connectivity clustering, is based solely on the ontology itself and the second, similarity clustering, utilizes a similarity measure derived from the ontology. Central to both models is the novel idea of letting a set of concepts from the ontology be the summary rather than a summary in natural language.

Finally, the **nineth** draws conclusions and presents a discussion of further work.

1.3 Contributions

This dissertation is the result of an industrial PhD funded by Scan·Jour A/S under the industrial PhD program established by the Ministry of Science, Technology and Innovation. The aim of this industrial PhD has been to explore the options for combining corpus statistics with ontology-based information retrieval. The research has taken place within the framework of the ONTOQUERY project, which means that the research issues involved and the results presented are thus related to the work being done within this project.

As noted in connection with the research question, the aim has been to explore appropriate parts of an ontology-based information retrieval system that could benefit from introducing corpus statistics. In this endeavor, contributions within the following subjects have been made:

1. *Semantic analysis of the relations denoted by prepositions.* Joint work done with Tine Lassen, Roskilde University, which is presented in chapter 5. The chapter is to a large part a rendering of Lassen & Terney (2006*a*,*b*).

2. *A model of descriptor expansion in the Vector Space Model.* This previously unpublished work in chapter 7 is by this author alone.

3. *Approaches to combining semantic and distributional similarity*. What is presented here in chapter 6 is a further development of the ideas presented in Terney (2007).

4. *Conceptual summaries*. Joint work done with Troels Andreasen and Henrik Bulskov, Roskilde University, is presented in chapter 8. This is to a certain extend a rendering of Bulskov et al. (2007), Andreasen et al. (2008) except for the work on "Prioritized connectivity clustering" which subsequently to the submission of this dissertation has been published in Bulskov et al. (2008).

All the joint work has been characterized by an equal amount of work from all the participants.

Part I

Foundations

Chapter 2

Information Retrieval

Information retrieval deals with the representation, storage, organization and access to information items. Here, information items can, in principle, be any kind of objects. However, until now, developed information retrieval systems and research within the field have, to a large extent, focused on retrieval of documents with a textual content. Therefore, information retrieval is often used synonymously with text retrieval. In this context, documents can be of any type or structure, e.g. emails, web pages, books or fragments hereof, e.g. sections, paragraphs, and sentences.

The following example taken from the Text REtrieval Conferences (TREC) illustrates important processes in information retrieval. TREC is an annual conference where a variety of research groups competes and exchange ideas on text retrieval. One of the tracks that ran in 2003 and 2005, was the High Accuracy Retrieval (HARD) track, where the goal was high accuracy retrieval using additional information about the searcher and/or the search context captured using very targeted interaction with the searcher. One of the questions posed, or topics using TREC terminology, in 2005 was (NIST 2007):

> *Identify positive accomplishments of the Hubble telescope since it was launched in 1991*

Where the criteria for information being relevant was judged by:

> *Documents are relevant that show the Hubble telescope has produced new data, better quality data than previously available, data that has increased human knowledge of the universe, or data that has led to disproving previously existing theories or hypotheses. Documents limited to the shortcomings of the telescope would be irrelevant. Details of repairs or modifications to the telescope without reference to positive achievements would not be relevant*

The above question expresses an information need that has to be transformed into a language interpretable by the system. As formulated, the question above would

normally not be posed directly as a query to present-day search engines. Instead, the question would be reformulated into a set of keywords like “accomplishments Hubble” or “accomplishment Hubble telescope”. This is done both in order to reduce typing and because most search engines are keywords based.

A collection of documents also exists from which we wish to extract the needed information. The document collection can be the entire web, the contents of an enterprise document management system, a mailbox, a desktop or even a single document. Then, using some type of function, the system matches the documents, or a representation hereof, with the query and presents the result. Finally, the result of the retrieval process can then be presented, typically in the form of a list with the possible addition of small text excerpts from the documents. The text excerpt is presented in order for the user to be able to quickly assess the relevance of a retrieved document without actually having to look through it. This simplification of the retrieval process is shown in figure 2.1 (inspired by Ingwersen (1992)):

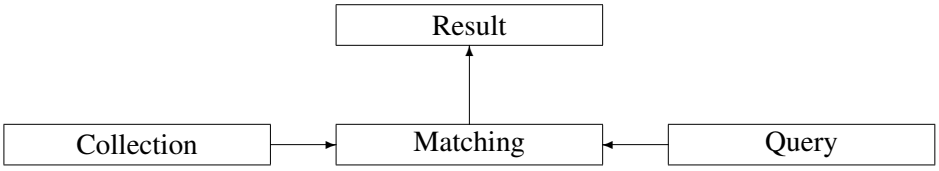


Figure 2.1: A simple model of the basic processes in information retrieval

The relevance judgment of the result made by the user also shows how information retrieval differs from standard data retrieval. In data retrieval, all objects satisfying clearly stated criteria must be retrieved and only these objects. This is made possible, for instance, in relational databases by a well-defined structure and semantics. However, for the most part, information retrieval deals with natural language that is not well-structured and that contains many semantic ambiguities. In other words, we move from matching by means of well-defined criteria to matching a more uncertain estimation of “what is most likely relevant”.

This chapter begins with a detailed view of the processes involved in information retrieval. With this view in place terms central to the field of information retrieval will be described followed by a section on indexing and term weighting. Hereafter, a brief overview of the three classical retrieval models will be presented: the Boolean model, the Vector Space model, and the Probabilistic model (Baeza-Yates & Ribeiro-Neto 1999). Also a Fuzzy model is presented. The chapter concludes with a summary and a discussion. The purpose of this chapter is to provide the fundamental framework for applying the approaches and models presented in subsequent chapters.

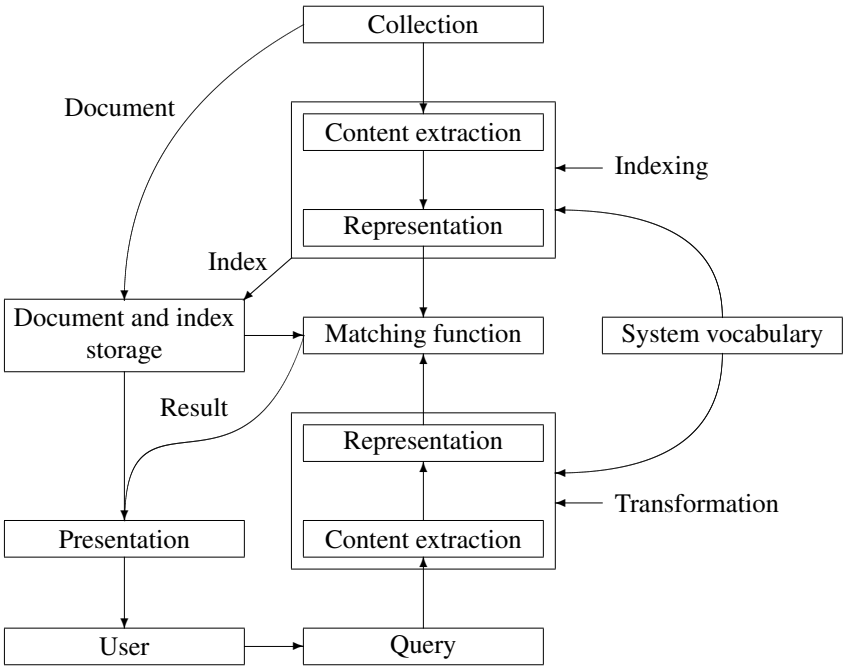


Figure 2.2: The major processes in an information retrieval system (a minor modified version of Lancaster's (1968) model.)

2.1 A Prototypical Information Retrieval System

Figure 2.2 illustrates in more detail the main processes in an information retrieval system (a minor modified version of Lancaster's (1968) model). At the top of the figure is a collection of documents that undergo a transformation from the original document format and structure into the system's internal representation or logical view of the documents. In classic keyword based information retrieval, this transformation of the documents results in a set of keywords that can be stored as indexes in addition to the original document for later fast retrieval. The content extraction, representation and storing of the indexes is called indexing. The system's vocabulary is the language used by the information retrieval system for describing documents and queries. For instance, the system's vocabulary can be the keywords in the collection.

The lower part of the figure is an almost mirror image of the upper part of the figure. The transformation of the queries, however, is reduced to an extraction of the content of the query and a representation of this content using the same system vocabulary as the indexing. By doing so, the system makes it possible to perform the matching of documents with the queries depicted at the center of the figure. Finally, the result of the matching is presented to the user, for instance, as a ranked list of references to the relevant documents, judged by the system as relevant, with accompanying text excerpts from the documents.

Before we proceed a definition of a few of the terms used in this dissertation is appropriate. A *term* is a lexical unit in the sense of a sequence of characters much similar to a *word*, and as noted in the beginning of the chapter word and terms are here used interchangeably. In information retrieval, however, *term* can also be an element in the system vocabulary which does not have to be be a whole word, in the sense of a meaningful unit of language that native speakers would use. Instead it can be a part of word like "inform" or "theor" (Jones 1972). The two, "inform" and "theor", are examples of the *stem* which is the part of a word that left when all affixes has been stripped (Jurafsky & Martin 2000). Stemming refers to the process of reducing a word to its stem and it has been widely used in information retrieval. Stem is sometimes confused with *lemma* which is the head word it appears as the entry in a lexicon. For instance *go*, *goes*, *going*, *went*, *gone* are all different forms of the same *lexeme* with *go* being the lemma. Neither stems or lexemes are of any particular interest in the context of this dissertation but the terms can be found in some of the subsequent chapters in the description of related work.

A *description* is the set of *descriptors* assigned to a given document. Accordingly the set of descriptors used to describe all documents constitute the system vocabulary. It is common to use term and descriptor interchangeably in the information retrieval literature because commonly terms are used as the system vocabulary. But there is a difference as later chapters on ontology-based indexing will illustrate. Finally, of special interest in ontology-based information retrieval, is of course *concept* in the sense of an abstract or general idea. A *concept* can be expressed by using different word which are then considered as synonyms.

Two important measures for retrieval evaluation are *precision* and *recall*. Precision is measured as the fraction of relevant documents in the set of documents retrieved. Recall, on the other hand, is measured as the fraction of relevant documents. High precision indicates that among the retrieved documents most of them were relevant. High recall indicates that most of the relevant documents were retrieved from the collection of documents. There is an inherent conflict between achieving both a high precision and a high recall, since it is difficult to find all the relevant documents while still maintaining high precision.

With these terms in place we can continue to how to deal with the optimal description of documents in keyword based information retrieval. We start out with a continuation of the TREC example from the beginning of the chapter.

2.2 Indexing and Term Weighting

In order to get from the original document to the systems internal representation, some kind of transformation is necessary. Table 2.1 shows part of speech tags for first sentence in the TREC evaluation criteria presented on page 9:

This table can be used to exemplify three important aspects in information retrieval. First, even though the table splits up the sentence, just looking at the different

Nouns	Verbs	Other
Documents$_{NN2}$	are$_{VBB}$	relevant$_{AJ0}$
Hubble$_{NP0}$	show$_{VVB}$	that$_{CJT}$
telescope$_{NN1}$	has$_{VHZ}$	the$_{AT0}$
data$_{NN0}$	produced$_{VVN}$	new$_{AJ0}$
quality$_{NN1}$	has$_{VHZ}$	better$_{AJC}$
data$_{NN0}$	increased$_{VVN}$	than$_{CJS}$
data$_{NN0}$	has$_{VHZ}$	previously$_{AV0}$
knowledge$_{NN1}$	led$_{VVN}$	available$_{AJ0}$
universe$_{NN1}$	disproving$_{VVG}$	that$_{CJT}$
data$_{NN0}$		human$_{AJ0}$
theories$_{NN2}$		of$_{PRF}$
hypotheses$_{NN2}$		the$_{AT0}$
		or$_{CJC}$
		that$_{CJT}$
		to$_{PRP}$
		previously$_{AV0}$
		existing$_{AJ0}$
		or$_{CJC}$

Table 2.1: The first sentence of the evaluation criteria from the TREC example on page 9. The sentence is organized by part of speech. The tagging was performed with the online CLAWS tagger using the C5 tagset, excluding punctuation (Garside & Smith 1997, CLAWS 2008).

words in the table provides a pretty good idea of what the sentence is about. So, instead of viewing the content of the sentence as a close-knit composition, we can, to an extent, extract and represent the content of the sentence as a set of words. This is behind the common assumption of *term independence*, that is, the distribution of one term a in the document collection is independent of the distribution of b (Robertson & Jones 1976). Though unrealistic it makes makes the information retrieval models, which we shall look at in the next section, much simpler (Salton et al. 1982). Also assuming term correlation does not necessarily improve retrieval performance (Baeza-Yates & Ribeiro-Neto 1999).

Second, the table exemplifies how the different part of speech contributes more or less to the content of the document. *Hubble* and *data* are, for instance, much better at capturing the semantics of the TREC sentence than *increased* and *show*, or *better* and *relevant*.

Third, the evaluation sentence and table 2.1 make clear that the word *data* is key if one were to describe the criteria, simply because *data* appears more frequently than the other words in the sentence. A description based on term independence should therefore stress the importance of the word *data* related to the other words in the criteria. In other words, the word *data* should be given a higher weight than, say, *theories*, which only occurs once in the sentence. With this intuitive understanding we can now proceed to how indexing and term weighting are treated in keyword based information retrieval.

In information retrieval two fundamental notions of indexing are *index exhaustivity* and *term specificity* (Salton & Yang 1973). Index exhaustivity denotes the coverage of the description with respect to the topics in a given document, and term specificity refers to the level of detail a given concept in a given document is described (Jones 1972). The more exhaustive a document description is the more likely it is that relevant documents are retrieved as a response to user queries, and similar the more specific the terms are the less likely it is that non-relevant documents are retrieved. The challenge is naturally to find an optimal level of exhaustivity and specificity in order to ensure that as many relevant documents are being retrieved and that non-relevant documents are not retrieved.

Besides having the above semantic interpretation the index exhaustivity and term specificity can also be interpreted as statistical properties of term use. Thus the exhaustivity of a description is a function of the number of terms it contains and the specificity of a term is a function of number of documents it pertains to (Jones 1972). Thus there is slight difference in the common semantic and statistical interpretation of the word. A term can be specific from a semantic point of view (e.g. *neuropsychological*) but if it is widely used in all the documents in the collection is non-specific from a statistical point of view. In general, we are looking for terms that describe the document well, taking into account the term's frequency in other documents. If a term is good at describing the entire corpus, i.e. it has a high overall frequency, then it is usually poor at *discriminating* among the different documents, giving it a low *resolving power*. The term resolving power was introduced by Luhn (1958), and Salton and his colleagues later introduced discrimination value (Salton & McGill 1982). Luhns notion of resolving power was based on the assumption that if your order terms in order of their decreasing frequency, the value of a terms as document descriptors has a Gaussian distribution. This is sought illustrated in in figure 2.3.

Where the resolving power of a term is based only on the frequency, the discrimination value proposed by Salton and his colleagues measures to what degree a terms increases or decreases the average document-pair similarity (Salton & McGill 1982). Thus a term with high discrimination value decreases the average document-pair similarity, and a term with a low discrimination value either increases the average document-pair similarity or leaves it unaffected. Though expressed differently resolving and discrimination value are both notions of the same fundamental concept.

In order to ensure a term weight which takes into account the above factors, a good term weighting scheme typically includes both a local and a global weight. Local means how well a given term describes the document and global means how well a given term describes the entire collection. A commonly used weight is the term frequency, tf, as the local weight combined with the inverse document frequency,

Figure 2.3: An illustration of Luhns notion of resolving power (Luhn 1958).

idf, as the global weight measured as Baeza-Yates & Ribeiro-Neto (1999):

$$tf_{i,j} = \frac{f_{i,j}}{\max_l(f_{l,j})} \tag{2.1}$$

$$idf_i = \log\left(\frac{N}{n_i}\right) \tag{2.2}$$

$$tfidf_{i,j} = \frac{f_{i,j}}{\max_l(f_{l,j})} \cdot \log\left(\frac{N}{n_i}\right) \tag{2.3}$$

where $f_{i,j}$ is the frequency of term t_i in document d_j, $\max_l(f_{l,j})$ is the maximum frequency computed over all terms which are in document d_j, and N is the total number of documents in the corpus/system, and n_i is the number of documents where t_i appears.

The weight, $tfidf$, can be measured in many ways and Salton & Buckley (1988) is a classic reference on the topic. In relation to figure 2.2, the main task in the two content extraction processes is often to tokenize the document into terms, perhaps excluding high frequent words by using of stop lists, and, finally, by adding weights to each term. In this way, the document and the query can be expressed using the same system vocabulary and representation, thereby making it possible for the matching function to perform the matching.

2.3 Retrieval Models

A common trait of all the retrieval models presented in this chapter is that they all assume term independence and document independence, i.e. a document's relevance

is not evaluated relative to the other documents retrieved. Since the information value of a second document being identical to a previously retrieved document is very small this independence assumption is clearly erroneous. However, like the term independence assumption, the simplification is made in order to get simple models both from a conceptual and a computational viewpoint. We will now present four different information retrieval models. The first three: the Boolean model, the Vector Space model, and the Probabilistic model are the so called classical models (Baeza-Yates & Ribeiro-Neto 1999). They are interesting in an account of information retrieval because they show different methods for representing and matching documents and queries. The Boolean model is characterized by the logical connectives that can be used in query formulation. The Vector Space model is characterized by its term weighting and how closeness in the vector space is used as a measure of relatedness between query and document. The Probabilistic model is especially characterized by ranking of document based on the probability of a document being relevant to a query. Besides the three models, fuzzy theory and a fuzzy information retrieval model is presented. The fuzzy retrieval model presented in this chapter is presented because it includes a thesaurus in the matching of document and queries.

2.3.1 Boolean Model

The framework of the Boolean information retrieval model is, as the name reveals, based on Boolean algebra. In the *Boolean model* the index terms are either present or absent, i.e. the weighting of the index terms is binary. A document can be represented as an element in the power set of the set of index terms in the corpus (the system vocabulary). A query, q, is represented by a set of index terms connected either explicitly by the user or implicitly by the system with the logical connectives, *and*, *or* and *not*. Any Boolean expression can be re-represented in disjunctive normal form, e.g. the query q_a "dogs *and* (mice *or not* cats)" can be represented in disjunctive normal form as:

$$\begin{aligned} dogs \wedge (mice \vee \neg cats) \quad \Rightarrow \quad & (dogs \wedge mice \wedge cats) \vee \\ & (dogs \wedge mice \wedge \neg cats) \vee \\ & (dogs \wedge \neg mice \wedge \neg cats) \end{aligned}$$

Any documents satisfying one of the conjunctive components of the query will be retrieved by the Boolean Model. A positive term is satisfied if the term is included in the set of terms representing the document. A negative term is satisfied if the term is not included. Documents are considered either to be relevant or non-relevant with no notion of partial match.

In many aspects, the Boolean model is the simplest model, and its straightforward formalism and precise and "intuitive" semantics have ensured its popularity

(Baeza-Yates & Ribeiro-Neto 1999). However, as with many other simple models, the Boolean model suffers from some major drawbacks. The intuitivity of the semantics is deceptive, e.g. most people with no background in logic would be surprised that with, e.g. the query "dogs *and* (mice *or not* cats)" will retrieve a document containing the term *dogs, mice* and *cats* as illustrated in figure 2.4.

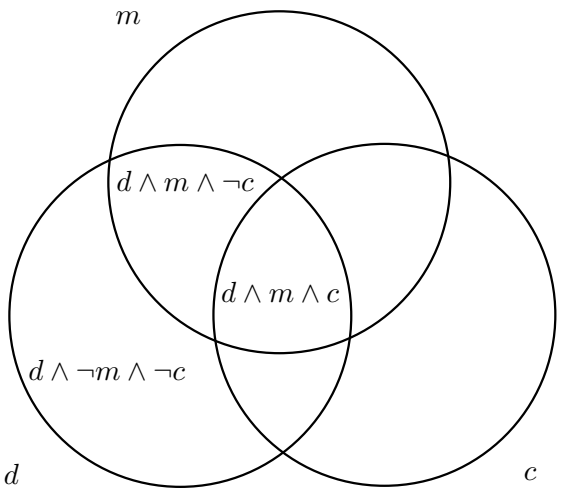


Figure 2.4: Venn diagram of the Boolean expression "dogs and (mice or not cats)" $d \wedge (m \wedge \neg c)$ that shows its disjunctive components.

In addition, the common interpretation of *and* is union not intersection, and users generally find it difficult to express their information needs in the Boolean model (Topi & Lucas 2005, Cooper 1997, Avrahami & Kareev 1993). Surveys indicate that end users queries might be getting more sophisticated (Jansen & Pooch 2001) though most studies show that less than 2% of end users use the *not* operator and less than 3% of the end users use the *or* operator (Markey 2007). Another problem with the Boolean model is the notion of relevance since documents are regarded as either relevant or not relevant with no ranking. Owing to the lack of partial match, systems based on the Boolean model often retrieve too many or too few results. To a certain extent, this can be remedied by ranking by the cardinality of the intersection of the query and the document. Though usually perceived as an information retrieval model, the Boolean model is therefore in fact much more of a data retrieval model.

2.3.2 Vector Space Model

Salton and his associates pioneered the vector space model working on the `SMART` retrieval system (Salton & Buckley 1988, Salton et al. 1975, Salton & Lesk 1968, Baeza-Yates & Ribeiro-Neto 1999). At the heart of the *Vector space model* is the possibility of representing the system vocabulary as an n-dimensional vector space,

where each document and query can be viewed as a vector in this space. Similarity or relatedness can then be evaluated as the closeness between the document vector and the query vector in the vector space. The vector space model opens up for the use of a more sophisticated representation since we can now add weights to each term, use partial matching, and rank the retrieved documents based on the similarity between the document and the query. A common measure of similarity between two vectors is the cosine. If $w_{i,d}$ and $w_{i,q}$ denote the weight of term i in document $\underline{d}$ and query $\underline{q}$ respectively, then the cosine can be expressed as:

$$sim(d,q) = \cos(d,q) = \frac{\sum_{i=1}^{n} w_{i,d} \cdot w_{i,q}}{\sqrt{\sum_{i=1}^{n} w_{i,d}} \cdot \sqrt{\sum_{i=1}^{n} w_{i,d}}} \tag{2.4}$$

In principle, one could chose to simply apply the inner product as a similarity measure, but the cosine takes into account the length of the vectors, and can thus be regarded as a normalization of the similarity. Assume, for example, that we have a query, q, a document, d, and three terms in our vocabulary. For simplicity, we presume each term is weighted by its number of occurrences in the document and in the query. The weights of the terms and the positions of the vectors in the space are illustrated in figure 2.5:

	d	q
t_1	1	1
t_2	2	0
t_3	1	1

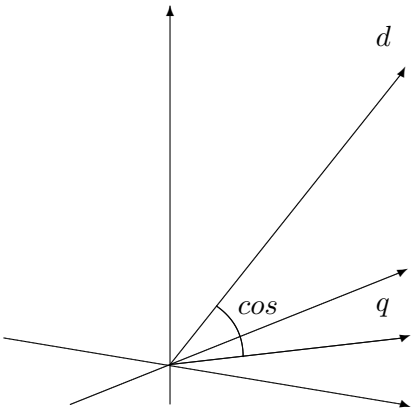


Figure 2.5: An illustration of a document, d, and a query, q, in the vector space model.

Based on this, the cosine can be calculated as:

$$\cos(d,q) = \frac{1 \times 1 + 2 \times 0 + 1 \times 1}{\sqrt{1^2 + 2^2 + 1^2} \cdot \sqrt{1^2 + 0^2 + 1^2}} = 0.58$$

The obvious advantages of the vector space model compared to the Boolean model is the option of introducing advanced term weighting, the ranking of results, and partial matching. The main drawback –and this goes for all models that are more advanced than the Boolean Model– is that the clear connection between the interpretation of the query and participation in the result is lost. This makes it more difficult for users to figure out how to restate their information needs, if the information is

not found immediately, unless e.g. relevance feedback is applied (Salton & Buckley 1990).

2.3.3 Probabilistic Model

This section will describe the classic probabilistic model introduced by Robertson & Jones (1976) known as the *binary independence model* (Baeza-Yates & Ribeiro-Neto 1999). The fundamental idea of the binary independence model is that the best ordering of documents, presented to the user as a response to a query, is a ranking where the documents most *likely* to be relevant are nearest the top (Robertson & Jones 1976). The challenge is naturally how to asses the probability that a document is relevant. In the binary independence model this challenge is met by some independence assumptions and an ordering principle. First the model assumes that the occurrences of different terms are independent within the set of relevant documents, and that the occurrences of different terms are independent within the set of non-relevant documents. Second the model assumes, as ordering principles, that the probability of a document being relevant should be calculated from the terms present in the document and from the terms absent in the document Robertson & Jones (1976).

Following Baeza-Yates & Ribeiro-Neto (1999) let the index term weight w_i in document d_j and query q be all binary i.e. $w_{i,j} \in \{0,1\}, w_{i,q} \in \{0,1\}$. Let R be the set of relevant documents and $\bar{R}$ be the set of non-relevant documents. Also let $P(R|d_j)$ be the probability that document d_j is relevant to the query and let $P(\overline{R}|d_j)$ be the probability that document d_j is not relevant to the query. The odds of document d_j being relevant to the query can be expressed as:

$$sim(d_j, q) = \frac{P(R|d_j)}{P(\overline{R}|d_j)}, \tag{2.5}$$

which also can be interpreted as the similarity between document d_j and query q. Using Bayes' rule:

$$P(R|d_j) = \frac{P(d_j|R) \times P(R)}{P(d_j)}, \tag{2.6}$$

equation 2.5 can be restated as:

$$sim(d_j, q) = \frac{P(d_j|R) \times P(R)}{P(d_j|\overline{R}) \times P(\overline{R})} \tag{2.7}$$

Since we are only interested in the ranking of the retrieved documents, and since $P(R)$ and $P(\overline{R})$ are constant for all documents, we can instead of equation 2.7 use:

$$sim(d_j, q) \approx \frac{P(d_j|R)}{P(d_j|\overline{R})} \tag{2.8}$$

The independence assumptions noted above enables us to rewrite the probability of d_j being in R and $\overline{R}$ respectively as

$$sim(d_j, q) \approx \frac{\prod_{i=1,w_{i,j}=1}^{n} P(t_i|R) \times \prod_{i=1,w_{i,j}=0}^{n} P(\overline{t_i}|R)}{\prod_{i=1,w_{i,j}=1}^{n} P(t_i|\overline{R}) \times \prod_{i=1,w_{i,j}=0}^{n} P(\overline{t_i}|\overline{R})} \tag{2.9}$$

where $P(t_i|R)$ expresses the probability of index term t_i is present in a document selected randomly from R, and $P(\overline{t_i}|R)$ expresses the probability of index term t_i is not present in a document selected randomly from R. The probabilities with $\overline{R}$ has analogous meaning. Here n denotes the cardinality of the set of terms. The similarity between d_j and q is thus the product of the probabilities of the index terms in document d_j being present in a document randomly selected from R, and the probabilities of the index terms not in d_j not being present in a randomly selected document from R. This is then divided by the analogous probabilities of for the index terms in $\overline{R}$. An implicit assumption is naturally, that the similarity is only based on the terms in the documents and in the query.

By ignoring factors which are constant to all documents given the same query the similarity expressed in equation 2.9 can be written as

$$sim(d_j, q) \approx \sum_{i=1}^{n} w_{i,q} \times w_{i,j} \left(\log \frac{P(t_i|R)}{1 - P(t_i|R)} + \log \frac{P(t_i|\overline{R})}{1 - P(t_i|\overline{R})} \right) \tag{2.10}$$

which is an important expression for ranking documents in the probabilistic model Baeza-Yates & Ribeiro-Neto (1999).

The major challenge in the model is that R is not known at query time, and thus a method is needed to estimate $P(t_i|R)$ and $P(t_i|\overline{R})$. One approach is to have the user identify a small subset of the relevant documents, and then iteratively refine the estimated probabilities. Alternatively assume that all terms in the query are equally likely to appear in a document randomly chosen from R, and to chose an initial estimate of this probability, say, 0.5 (Baeza-Yates & Ribeiro-Neto 1999). By these assumptions and initial ranking of the documents can be provided which again can be iteratively improved.

The advantage of the binary independence model is the ranking of results, but the challenge is to estimate the initial probability of a document being relevant or not. Also, it is not advantageous to only be able to interpret results and index terms within a binary framework. There has, to best of our knowledge, not been any attempts to create an ontology-based probabilistic model.

2.3.4 Fuzzy Information Retrieval

One of the important features of the Boolean model is the possibility to use the connectives, *and*, *or*, and *not*, which are not present in the vector space model or in

the Probabilistic Model. Being able to use logical connectives gives users a more advanced query language, which more experienced users especially can benefit from. A major problem, however, with the two-valued logic applied in the Boolean Model is the inability to handle borderline cases. If an element is (very) close to the precisely defined border, it will only be taken as evidence of one of the states.

Boolean logic operates with only two truth values, *true* and *false*. Fuzzy logic, on the other hand, operates with the *degree* of something being true. A common example is the set of tall men. When is a man tall? When he is taller than 1.8 meters or perhaps two meters? Most people would agree that a man who is two meters tall is tall, whereas a man who is 1.8 meters tall can be considered tall to a certain degree. Following this intuitive notion, a document can be considered relevant to a query to a certain degree.

Compared to the other models there is not a well established fuzzy model but rather different applications of fuzzy sets in information retrieval (see e.g. Miyamoto (1990),Kraft et al. (1999),Pasi (2008)). With respect to query languages for instance fuzzy logic offers a soft interpretation of the Boolean connectives and thus query languages that are more flexible and well suited for expressing imprecise user needs. However, here the focus will be on fuzzy information retrieval using a fuzzy thesaurus because it is of particular interest in the context of ontology-based information retrieval. Since fuzzy logic is also central to chapter 8 on conceptual summaries the following sections will briefly state the basics of fuzzy set theory based on Klir & Yuan (1995).

The Fuzzy Membership Function

In Boolean logic, elements can be either members or non-members of a given set. These sets are referred to as *crisp* sets. If X denotes the universe of discourse, then the members of a crisp set, A, can be defined by a characteristic function, χ_A. This characteristic function maps elements of X to the set $\{0,1\}$ declaring their membership or non-membership of A:

$$\chi_A : X \rightarrow \{0, 1\} \tag{2.11}$$

This characteristic function is generalized in fuzzy logic by the *membership function* μ_A which for all elements in X denotes their degree of membership of the fuzzy set A. The universal set is always a crisp set. The degree of membership can be expressed by a real valued number in the interval [0,1]:

$$\mu_A : X \rightarrow [0, 1] \tag{2.12}$$

where 1 denotes full membership, 0 denotes no membership and $0 < \mu_A(x) < 1$ denotes partial membership. Another notation of the membership function commonly used is simply A where there is no distinction between the set and the membership

function (Klir & Yuan 1995):

$$A : X \rightarrow [0, 1] \tag{2.13}$$

Similar to the characteristic function in classical set theory, the membership function represents a given concept, e.g. the set of tall men. The membership function could be designed in an infinite number of ways if it adheres to the fundamental properties of the concept (Klir & Yuan 1995).

Cardinality, $\alpha - cut$ and Strong $\alpha - cut$

Two important concepts in fuzzy logic are the $\alpha - cut$ and strong $\alpha - cut$. Two special crisp sets, ${}^{\alpha}A$ and ${}^{\alpha+}A$, result from applying the $\alpha - cut$ and strong $\alpha - cut$, respectively:

$$\begin{array}{rcll} {}^{\alpha}A & = & \{x|\mu_A(x) \geq \alpha\} & (\alpha - cut) \\ {}^{\alpha+}A & = & \{x|\mu_A(x) > \alpha\} & (\textit{strong } \alpha - cut) \end{array} \tag{2.14}$$

That is, ${}^{\alpha}A$ is the crisp set of all the elements in A having a degree of membership equal to or above the threshold $\alpha \in [0, 1]$. As a further restriction hereof, ${}^{\alpha+}A$ is the crisp set of all the elements in A having a membership strictly above the threshold $\alpha \in [0, 1]$. In the special case of $\alpha = 0$, ${}^{\alpha+}A$ is called the support of A. When $\alpha = 1$, ${}^{\alpha}A$ is called the core of A. For sets having a finite support set, the elements in A can be specified using the following notation, where a_i is the degree of membership in A of element x_i:

$$A = a_1/x_1 + a_2/x_2 + \ldots + a_n/x_n = \sum_{i=1}^{n} a_i/x_i \tag{2.15}$$

Scalar cardinality, or the sigma count as it is also referred to, is defined as:

$$|A| = \sum_{x \in X} \mu_A(x) \tag{2.16}$$

The scalar cardinality is thus the sum of all the degrees of membership (the a_i's in equation 2.15). Note the difference in the use of Σ here. In equation 2.15, Σ, denotes an enumeration whereas in 2.16, Σ, denotes the actual sum of the membership values of the elements.

Union, Intersection and Complement

The fuzzy generalizations of the Boolean intersection and union are the T-norms, (T), and the T-conorms, (S). There are several possible ways of defining union and

intersection that still follow the fundamental ideas about intersection and union (Klir & Yuan 1995). The standard definition of intersection and union in fuzzy logic is:

$$\mu_{A\cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (2.17)$$
$$\mu_{A\cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (2.18)$$

Like union and intersection, the complement can be defined in various ways. Though the standard complement is defined as:

$$\mu_{\overline{A}}(x) = 1 - \mu_A(x) \quad (2.19)$$

Inclusion and Subsethood

Given the two fuzzy sets A and B, A is a subset of B if for all x:

$$\mu_A(x) \leq \mu_B(x) \quad (2.20)$$

This corresponds to our intuitive notion of inclusion, since for all elements, x, they have a higher or equal degree of membership in B than their degree of membership in A. However, sometimes there will be cases where 2.20 is violated for some x. The degree of subset measures to what extend 2.20 is violated, i.e. to what degree elements in X have a higher degree of membership in A than in B:

$$subsethood(A, B) = \frac{1}{|A|}\left(|A| - \sum_{x\in X} \max[0, \mu_A(x) - \mu_B(x)]\right) \quad (2.21)$$

Clearly, equation 2.21 sums to one if equation 2.20 is not violated (everything after $\sum$ equals 0 and then only $|A|/|A|$ is left).

Information Retrieval Using a Fuzzy Thesaurus

Here we will use the notation and example of fuzzy information retrieval using a thesaurus given in Klir & Yuan (1995). Similar presentations can be found in Miyamoto (1990), and in Baeza-Yates & Ribeiro-Neto (1999).

There are two important relations in fuzzy information retrieval using a fuzzy thesaurus. First there is the indexing relation and second there is the fuzzy thesaurus.

Assuming we have a corpus of documents, D, and a vocabulary of terms, T, the relevance of index terms to individual documents can be expressed by a binary fuzzy indexing relation, I, as

$$I : T \times D \rightarrow [0, 1] \quad (2.22)$$

The membership value $I(t_i, d_j)$ specifies for each $t \in T$ and each $d \in D$ the degree to which term t_i describes document d_j. The degree could for instance be set to the $tfidf$ weight described previously in this chapter.

The fuzzy thesaurus is a reflexive fuzzy relation O defined on T^2. For each pair of index terms $\langle t_i, t_k \rangle \in T^2$, $O(t_i, t_k)$ expresses the degree of association between t_i and t_k:

$$O : T \times T \rightarrow [0, 1] \tag{2.23}$$

The degree of association can be asserted in numerous ways, and this topic will be treated in details in chapters to come. For now let us presume this degree of association is just given by the thesaurus.

Given the fuzzy indexing relation and the fuzzy thesaurus a model for a thesaurus based fuzzy information retrieval can now be expressed in the following manner. A query Q can be expressed as any fuzzy subset of the set of terms T. By using the fuzzy thesaurus the set of query terms can be expanded with associated terms. The expanded query E can be obtained by composing the query Q with the fuzzy thesaurus O:

$$Q \circ O = E, \tag{2.24}$$

where $\circ$ can be understood as the max-min composition, so that:

$$E(t_k) = \max_{t_i \in T}[\min(Q(t_i), O(t_i, t_k))] \tag{2.25}$$

forall t_k in T.The set of retrieved documents can now be expressed as the fuzzy result set R defined on D. R is obtained by composing the expanded query E with the relevance relation I:

$$E \circ I = R, \tag{2.26}$$

where $\circ$ again can be understood as the max-min composition.

As an illustration of the retrieval process consider the following simple example where the query only contains the three index terms:

$$\begin{aligned} t_1 &= \text{fuzzy logic} \\ t_2 &= \text{fuzzy relation equation} \\ t_3 &= \text{fuzzy modus ponens} \end{aligned}$$

Thus the support ${}^{0+}Q = \{t_1, t_2, t_3\}$ is the fuzzy set expressing the query. Let the vector representation of Q be

$$Q = \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & t_3 & \end{array} \\ \left[\begin{array}{cccc} 1 & .4 & 1 & \end{array}\right] \end{array}$$

Assume also that the thesaurus O looks like the following matrix restricted to the support of Q and nonzero columns:

$$O = \begin{array}{c} \begin{array}{cccccc} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 \end{array} \\ \begin{array}{c} t_1 \\ t_2 \\ t_3 \end{array}\left[\begin{array}{cccccc} 1 & .2 & 1 & 1 & .5 & 1 \\ .2 & 1 & .1 & .7 & .9 & 0 \\ 1 & .4 & 1 & .9 & .3 & 1 \end{array}\right] \end{array}$$

where the columns express the association to t_1, t_2, t_3 of the terms $t_1, t_2, \ldots, t_6$ with:

$$\begin{aligned} t_4 &= \text{approximate reasoning} \\ t_5 &= \text{max-min composition} \\ t_6 &= \text{fuzzy implication} \end{aligned}$$

By equation 2.25 the expanded query E can be expressed as a composition Q and O. With Q and O as given above this results in an E which in vector form can be expressed as:

$$E = [\ 1 \quad .4 \quad 1 \quad 1 \quad .5 \quad 1]$$

Assume now that the relevant part of the relevance relation I, that is, I restricted to the support of E and nonzero columns is given by the matrix:

$$I = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{array} \begin{array}{c} \begin{array}{cccccccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \end{array} \\ \left[\begin{array}{cccccccccc} .2 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & .3 & 0 & .4 & 0 & 0 & 1 & 0 \\ 0 & 0 & .8 & 0 & .4 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & .9 & .7 & .5 \\ 1 & 0 & .5 & 0 & 0 & .6 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & .2 & 0 & 1 & 0 & 0 & .5 \end{array}\right] \end{array}$$

In other words $d_1, \ldots, d_{10}$ are the only documents related to the terms $t_1, \ldots, t_6$. The fuzzy result set R can now be obtained by composing E and I with results in:

$$R = \begin{array}{c} \begin{array}{cccccccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \end{array} \\ [\ .5 \quad 1 \quad 1 \quad .3 \quad .4 \quad .5 \quad 1 \quad .9 \quad .7 \quad .5 \quad] \end{array}$$

The result for d_2 is at first surprising given that the document does not contain any of the query terms t_1, t_2, t_3. However, document d_2 contains the terms t_4 (approximate reasoning), and t_6 (fuzzy implication) which by thesaurus is associated with term t_1 (fuzzy logic) to a degree of 1.

2.4 Ontology-Based Information Retrieval

After having set the overall scene for information retrieval we now turn to ontology-based information retrieval as it will be treated in this dissertation. The introduction in the first chapter sought to give an intuitive understanding of the merits of ontology-based information retrieval. The motivation behind ontology-based information retrieval is that natural language is ambiguous. The example in the introduction "kids arts dough oven" was an illustration hereof. Jurafsky & Martin (2000) gives with the sentence *"I made her duck"* another example. This sentence could at least have the following five different meanings:

1. I cooked waterfowl for her
2. I cooked waterfowl belonging to her
3. I created the (plaster?) duck she owns
4. I caused her to quickly lower her head or body
5. I waved my magic wand and turned her into undifferentiated waterfowl

The different meanings are caused by different sources of ambiguity of natural language. First *duck* and *her* are syntactically ambiguous. *Duck* can be both a noun and a *verb* and *her* can be a dative pronoun or a possessive pronoun. *Make* is also syntactically ambiguous. It can be transitive, i.e., taking a single direct object as in (2); or it can be ditransitive taking two objects (5), meaning that the first object *her* got made into the second object i.e. the *duck*. Also *make* can take a direct object and a verb as in (4), meaning that the object *her* got caused to perform the action *duck*. Finally *make* is also semantically ambiguous, that is, it can mean *to cook* something or *to create* something. In short, natural language is highly ambiguous.

Research in ontology-based information retrieval can be motivated by continuing on Jurafsky & Martin's example. Try to compare the sentences "I made her duck" and "she cooked me drake". From a lexical point of view there is no match between the sentences: they do not have a single word in common. However, from a semantic point of view the two sentences are highly related. Ontologies offers here the possibility of moving from lexical matching to conceptual matching of queries and documents. What an ontology exactly is will be treated in the next chapter, so let us for now suffice with a general notion of an ontology as some sort of structure that relates different concepts. For instance, if we know that *drake* and *duck* are related with an "is a" relation or "is a kind of" relation, then we can chose to expand the query "I made her duck" with *drake* or vice versa. If our information retrieval model allows us to use weights in the index, we can even add *drake* to the index to a degree that reflects how similar we perceive *duck* and *drake* to be. In the same manner we would be able to add *foie gras* and *confit* to the index.

Given the many sources of ambiguity in natural language, and in expressing conceptual knowledge in itself, ontology-based information retrieval system design is of course a difficult task. The focus in this dissertation will be on a subset of the important challenges involved and their possible solutions. Chapter 5 differs here from the other three chapters in the contribution part in the sense that it is the only chapter focused on content analysis of natural language. More precisely it will demonstrate an approach to extract semantic relations between concepts. In all the other chapters we will presume the meaning of the documents has already been extracted through some content analysis. Chapter 6 will look at how to measure the similarity between concepts (e.g. between *duck* and *drake*). Chapter 7 will present a previously unpublished model of how to determine what the weights of the expansion could be in the

vector space model. Finally, chapter 8 will present a way of summarizing the results of a query based on the ontology. Common for all the contributions are though that they can be perceived as "black box" from the users perspective. That is, the user is not required to have any knowledge of ontologies or being supplied with any kind of ontology-based query language or query assistance tools. Neither is the presented work in opposition to such tools or knowledge.

At this point it is important to note that ontology-based information retrieval is not a single framework, and there is no such thing as such thing as the ontology-based information retrieval model. Also there is not a common agreement of what model is the best basis of ontology-based information retrieval. Lee et al. (1993), Mihalcea & Moldovan (2000), Zhou et al. (2006) have for instance adopted the boolean model, while others like Gonzalo et al. (1998), Vallet et al. (2005), Nagypal (2005), Hliaoutakis et al. (2006), Li & Ramani (2007) have adopted the vector space model. This dissertation will focus on how to strengthen ontology-based information retrieval by including corpus statistics.

2.5 Discussion and Summary

The primary purpose of this chapter and the next chapter on ontologies is to establish a framework upon which the succeeding chapters on the various options for using statistics in ontology-based information retrieval can be built. This chapter has described the important processes involved in information retrieval. Different retrieval models have been presented that each differs in the way they represent documents and queries, and how the matching of the two is performed. Central to each of the models as they are presented here, however, is that they all index content at the lexical level of keywords. In other words, we index and match documents and queries on a lexical level by using a "bag-of-words" approach, i.e. words are just considered as a bag with no internal syntactic or semantic ordering.

The motivation for pursuing the path of ontology-based information retrieval is naturally that ignoring semantics or world knowledge in an information retrieval system might lead to suboptimal search results. If the user types the query "accomplishment Hubble telescope" based on an information need of "Identify positive accomplishments of the Hubble telescope since it was launched in 1991", many relevant documents might be ranked very low, or not even be found. For instance, NASA's own website listing "Hubble's Top Achievements" (NASA 2007) only matches on the single keyword *Hubble*. The same goes for documents with *attainment*, *(amazing) results*, etc. In other words, a simple keyword based approach is unable to handle paraphrases. Here, ontology-based information retrieval offers the possibility of indexing and matching on a conceptual level where "top achievements" and "positive results" become highly related as opposed to the lexical level, where they have nothing in common. In general, ontology-based information retrieval is a way of introducing semantics to the search process.

With reference to figure 2.2, shown on page 11, this thesis will present an ontology-based perspective of all the processes: System vocabulary (chapter 3), content extraction (chapter 4 and 5), indexing (chapter 7), matching (chapter 6), and, finally, presentation (chapter 8).

Chapter 3

Ontologies

Ontology in its classical sense refers to a philosophical discipline stretching back to Parmenides and Aristotle that concerns the nature and the organization of reality, i.e. what things are (Gómez-Pérez et al. 2004). In the discipline of knowledge engineering, which is our concern here, an *ontology* refers to an organization of a shared conceptualization, i.e. a knowledge model in the form of concepts and their relations as they are shared by a community.[1] A concept is a general notion of something or an idea of something formed by a combination of its characteristics, e.g. a *car* or *legal action*. Relations or roles are significant semantic associations between concepts, e.g. in the sentence "The chair is in the room", the chair is *located* in the room, or e.g. in the sentence "Five percent of all accidents are due to drunk driving", the accidents are *caused by* drunk driving.

Ontologies are typically visualized as a taxonomic ordering, an ISA hierarchy in the form of a directed graph where nodes resemble concepts and edges between nodes resemble relations. An excerpt from WordNet (Fellbaum 1998*b*), a general linguistic ontology illustrating this hierarchy, is depicted in figure 3.1

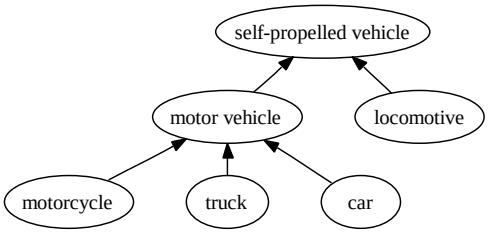


Figure 3.1: An excerpt of WordNet showing the taxonomic ordering of concepts in the form of a directed graph.

Ontologies can vary greatly in their degree of generality. The core or upper level

[1] Following Guarino & Giaretta's 1995 proposal, the philosophical discipline is denoted by a capital *O* versus a lower case *o* for ontology as a model of knowledge.

ontology of SIMPLE used in our work on semantic analysis presented in chapter 5 only includes abstract concepts like *natural substance* and *state*. WordNet, on the other hand, distinguishes, for example, between *bass* as an adult male singer with the lowest voice and *bass* as the lowest adult male singing voice. Ontologies can also vary greatly in their degree of formality. Both SIMPLE and WordNet are lightweight ontologies with a limited or no degree of logic formalism. DOLCE, on the other hand, is described in first order logic and implemented in description logic. Each of these three ontologies will be presented in section 3.2 in greater detail.

The primary purpose of this chapter is to introduce the notion of ontologies, i.e. what an ontology is, what its main components are and how they can be represented. Similar to the previous chapter, this chapter serves as a foundation for the succeeding chapters, which attempt to demonstrate how ontologies can be applied in information retrieval.

3.1 What Is An Ontology?

Studer et al. (1998) provide one of several existing loose definitions of what constitutes an ontology:

> *An ontology is a formal, explicit specification of a shared conceptualization. A "conceptualization" refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. "Explicit" means that the type of concepts used, and the constraints on their use are explicitly defined. For example, in medical domains, the concepts are diseases and symptoms, the relations between them are causal and a constraint is that a disease cannot cause itself. "Formal" refers to the fact that the ontology should be machine readable, which excludes natural language. "Shared" reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group.*

The different definitions of ontology vary mostly regarding how much they emphasize the formal aspects. Lassila & McGuinness (2001) consider a spectrum of interpretations of what an ontology is as depicted in figure 3.2, where the slanted line separates the strictly hierarchical from the non-strictly hierarchical. The direction of the horizontal line marks both an increasing degree of formality and the richness of the internal structure of the ontology. The terms heavyweight and lightweight ontologies are often used to describe formal ontologies with rich internal structures versus less formal and structurally limited ontologies.

The distinction between lightweight and heavyweight ontologies is thus by Lassila & McGuinness (2001) based on the formality of the ISA relation. Let us assume to concepts A and B where A subsumes B. In informal ISA hierarchies an instance

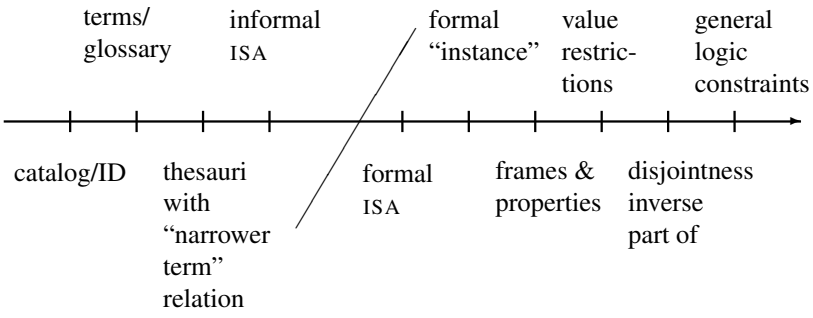


Figure 3.2: A visualization of the spectrum of formality in Lassila & McGuinness' (2001) interpretation of the word ontology. Ontologies on the left side of the spectrum are typically labeled lightweight and the ontologies on the right side of the spectrum are typically labeled heavyweight.

of the concept B is not necessarily a member of A. In formal ISA hierarchies if an instance of B is necessarily an instance of A as well.

3.1.1 Types of ontologies

Besides having a rich internal structure, ontologies can also be categorized according to their subject of conceptualization (Gómez-Pérez et al. 2004). Some ontologies model highly specific tasks, while others are abstract philosophical models.

General or common ontologies model common sense and are reusable across domains. *WordNet* and *Cyc* (Lenat 1995) are probably the most well known ontologies within this category.

Application ontologies model knowledge at the application level, e.g. the development and maintenance of a "skill management" ontology for Swiss Life, an insurance company (Lau & Sure 2002). The upper parts of an application ontology are usually connected to a more general domain ontology.

Domain ontologies model the knowledge within a given domain such as law or medicine. Like application ontologies, they are generally not reusable outside their scope, though they can overlap with other domain ontologies. The Unified Medical Language System (U.S. National Library of Medicine 2007), *UMLS*, and the *Gene Ontology* (Consortium 2000) are good examples of domain ontologies. The upper parts of a domain ontology are usually connected to a top or upper-level ontology.

Top or upper-level ontologies model the most generic concepts, such as how abstract entities are connected to the concept "Thing" (Sowa 2000). Despite their

very general nature, several different proposals for top ontologies exist, e.g. the SIMPLE core ontology described later in this chapter.

A final and important group of ontologies is *linguistic ontologies*. They differ from conceptual ontologies like Cyc in that they model word or lexical knowledge rather than world knowledge or encyclopedic knowledge. Thus, linguistic ontologies only contain concepts that are used in natural language, and they usually do not exhibit the same richness in relations generally found in non-linguistic ontologies. However, since natural language is closely connected to our conceptual model or the encyclopedic knowledge of a given phenomenon, linguistic ontologies are often used as a surrogate for other ontologies.

3.1.2 Lexical Appearance

The distinction between linguistic and more formal ontologies is tied to the fact that there is a difference between word forms and meaning. Different authors may use different terminology, but the distinction between words and meanings remains the same; words are used to denote a deeper semantic or conceptual meaning. Words in the sense of references to a concept are sometimes also termed *signs for concepts* or *the surface form of a word*. Table 3.1 illustrates the different levels of meaning using the noun phrase (NP) *thrombosis in the heart*.

Surface Form	thrombosis	in	the heart
Syntactic Structure	head of first NP	preposition	head of second NP
Ontological Level	disease	location	body part

Table 3.1: The different layers of meaning in the noun phrase *"thrombosis in the heart"*.

In ontology-based information retrieval noun phrases are vital because nouns denote the concepts that form the backbone of the ontology. To formally characterize the relationship between the ontological level and the word level, let us first define a minimal ontology, O_{min}, as a partially ordered set (poset) consisting of a set of concepts, C, and the conceptual inclusion relationship, ISA, as the partial ordering, $\leq$, of these concepts:

$$O_{min} = \langle C, \leq \rangle \tag{3.1}$$

In other words, a minimal ontology constitutes a hierarchy of concepts ordered by inclusion. A prerequisite for doing ontology-based information retrieval is the ability to establish a relation between the word level and the concepts in the ontology, O_{min}. To meet this end, we define the relation as $lex \subset W \times C$, where W is the set of surface forms of the concepts in C:

$$lex = (W, C) \tag{3.2}$$

For example, in the following list:

$$lex(pupil, \text{center of the eye})$$
$$lex(pupil, \text{student})$$
$$lex(car, \text{car})$$
$$lex(automobile, \text{car})$$

As Jensen & Nilsson (2003) notes, lex is not a "true" lexical relation since it does not relate lexical units but rather relates a lexical unit and a non linguistic object, namely a concept. In order to establish the relation between the surface form of a word and the underlying concept, identifying the *sense* of the word given the context of the word must be possible. To disambiguate between the various possible senses is by no means simple. WordNet, for instance, has more than 15,000 polysemous nouns (WordNet 3.0) while more than 250 of these nouns have at least five different senses, some of which are also high frequency nouns like, e.g. *break*, *pass* and *counter*.

As a result, though the relation between word and concepts, in many cases, can be solved by simply using a lexicon, for a large fraction of the words, some kind of automatic word sense disambiguation is required. There are different approaches to handling word sense disambiguation and Ide & Véronis (1998) provide a solid overview of the topic.

3.2 Representing Ontologies

Different languages can be used to represent ontological knowledge, e.g. conceptual graphs (Sowa 2000), first order logic, description logic (Masolo et al. 2003, Nardi & Brachman 2002) and lattice algebra (Nilsson 2001). The lattice algebraic language ONTOLOG is presented here because it includes a generative constructor that is useful in the indexing process described in chapter 4, and it also supports a view on the ontology as a directed graph, which is natural when focusing on semantic similarity measures, as is the case in chapter 6. Description logic will also be presented in a very condensed form since, in different variations, it constitutes the language backbone within the semantic web community.

3.2.1 ONTOLOG

ONTOLOG (Nilsson 2001) is a lattice algebraic ontology language used in the ONTO-QUERY project which extends the simple definition of an ontology given in section 3.1.2. The presentation of ONTOLOG here is based on Nilsson (2001). However, Partee et al. (1990) and Brink et al. (1994) are used to explain details of lattice algebra and the Pierce product not made explicit in Nilsson (2001). O denotes the actual ontology and $\mathcal{O}$ denotes the algebra for expressing and manipulating the ontology.

In equation 3.1, a minimal ontology was defined as the partially ordered set $O_{min} = \langle C, \leq \rangle$. In an arbitrary poset, C, c is an upper bound of elements in A,

$A \subseteq C$, if for all $a \in A, a \leq c$. If for all the upper bounds b of A, c is the smallest element and $c \leq b$, then c is the *supremum* or *least upper bound*, $lub(A) = c$ (Partee et al. 1990). Let the upper bounds U_C of a set A, with $A \subseteq C$, be defined as:

$$U_C(A) = \{C | c \in C, \forall a \in A : a \leq c\} \tag{3.3}$$

The least upper bound, lub_C, of A can then be defined as:

$$lub_C(A) = \{c | c \in U_C(A), \neg \exists b \in C : b \leq c\} \tag{3.4}$$

The *infimum* or *greatest lower bound*, glb, can be defined in the same manner. Least upper bounds are especially interesting from an ontological perspective because they are the most specific concepts that subsume all the concepts for which they are the upper bound. Least upper bounds play a central part in later chapters, especially in the last chapter on conceptual summaries where a fuzzyfied notion of least upper bounds is also introduced.

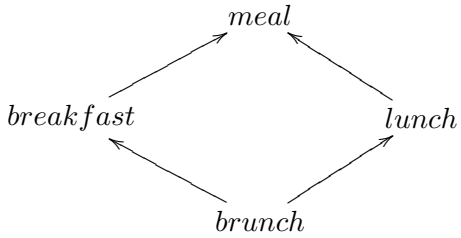
In ONTOLOG, the *conceptual sum* of two concepts, $a + b$, is defined as their least upper bound, $a + b = lub(\{a, b\})$, which corresponds to the lattice algebraic meet operator, $a \vee b$. In addition, the *conceptual product* of two concepts, $a \times b$, is defined as their greatest lower bound, $a \times b = glb(\{a, b\})$, corresponding to the lattice algebraic join operator, $a \wedge b$. A lattice is a special kind of poset characterized by a unique least upper bound and greatest lower bound for all element pairs, $a, b \in C$. With the adoption of the empty ***null*** concept or bottom concept, $\perp$, and the ***top*** concept, $\top$, as the identity elements, we can express and manipulate an ontology with the lattice algebra $\mathcal{O}_{la}$:

$$\mathcal{O}_{la} = \langle C, \leq, +, \times, \perp, \top \rangle \tag{3.5}$$

The basic reasoning in the ontology includes that if the product of two concepts is not explicitly stated, then it is considered to be ***null***. The lattice algebraic definition enables us to represent and construct new concepts in a compact form as, for example, in the following:

$$breakfast + lunch = meal$$
$$breakfast \times lunch = brunch$$
$$cat \times dog = \textbf{null}$$

Another advantage regarding lattices is that they can naturally be interpreted and visualized in a graphical form. This supports an intuitive notion of what is general and "upper" in the ontology and what is specific and "lower" in the ontology.



The lattice algebra, $\mathcal{O}_{la}$, defined above constitutes the language for expressing a *skeleton* ontology that positions all the atomic or primitive concepts in the lattice. This lattice algebraic framework is in ONTOLOG enriched by adding a supplementary set of semantic relations, R, and an algebra for relations interacting with sets, a so called Boolean module (Brink et al. 1994). A closer look at the set of semantic relations will be presented in the next two chapters, but an open set of possible semantic relations is listed in table 3.2.

Abbreviation	Description
TMP	Temporal anchoring, duration, inception etc.
LOC	Place, position
PRP	Purpose, function
WRT	With respect to
CHR	Characteristic (Property ascription)
CUM	Cum (i.e. with accompanying)
BMO	Means to end, instrument
CAU	Inanimate force/actor
CBY	Inverse CAU
POF	Part of whole, member of set
CMP	Inverse POF, whole constituted of parts
AGT	Animate being acting intentionally
PNT	Affected entity, effected entity
SRC	Source, origin, point of departure
RST	Result of act or process
DST	Destination of a moving process
?	?

Table 3.2: The open set of relations proposed by Nilsson (2001).

A Boolean module is a two-sorted algebra, which with the so called *Pierce product*, ":", can be used to combine an ontology, O, with the set of relations, R, by the mapping $O \times R \rightarrow O$ written as $r : c$ (Brink et al. 1994). The Pierce product is used to form compound concepts using the conceptual product:

$$c_1 \times (r : c_2) \tag{3.6}$$

As, e.g. "black dog":

$$dog \times (characterized\ by : black), \tag{3.7}$$

which should be interpreted as the product of the concept *dog* and the concept *characterized by black*. This ability to construct, in principle, an infinite amount of compound concepts from the initial set of concepts and the set of relations makes the ontology generative. The definition of our conceptual algebra thus becomes the Boolean module $\mathcal{O}$:

$$\mathcal{O} = \langle C, \leq, +, \times, \bot, \top, R, : \rangle \tag{3.8}$$

with C being the set of concepts, $\leq$ the ordering relation, $+$ the conceptual sum, $\times$ the conceptual product, $\perp$ the bottom concept, $\top$ the top concept, R the set of relations, and finally ":" as the Pierce operator.

With ONTOLOG as a conceptual algebra, the constructed compound concept can be interpreted as, and are conventionally written in ONTOLOG, the feature structure of the general form:

$$a[r_1 : b_1, ..., r_n : b_n] \tag{3.9}$$

Both nesting and multiple attributions can thus be described. For instance, the concept "dark blue screen" can be described as *screen*[CHR : *blue*[CHR : *dark*]], and the concept "big blue screen" can be described as *screen*[CHR : *blue*[CHR : *big*]].

3.2.2 Description logics

It can be argued that all formalisms for knowledge representation can be seen as a fragment of first order logic, and in this naive sense, all knowledge might as well be modeled in first order logic (Davis et al. 1993). Central to the research on description logic, however, is that only fragments of first order logic are needed for most knowledge-based systems, and that there is a tradeoff between the expressiveness of the language and the tractability of reasoning. Research has focused on how different concept-forming constructs or *constructors* influence tractability, i.e. to receive an answer in finite time does not necessarily imply that the answer was received in reasonable time (Baader & Nutt 2003). More specifically, inference with respect to subsumption has played a vital role, because ISA relationships are not specified by a knowledge engineer as in, e.g. semantic networks, but rather they are inferred from the definition of concepts.

In keeping with this fundamental aspect of description logic, the various language variants are defined and named by the set of constructors that they offer. Table 3.3 gives an overview of the different languages. Based on the application, and a thereby implied need for constructors, a sufficiently expressive language with the smallest possible complexity can be chosen.

In description logic the fundamental building blocks are atomic concepts, roles (relations), and instances. Atomic concepts are defined using symbolic names for more complex description. For instance, using the constructors of the most simple language, FL_0, in table 3.3 as an example the concept *Mother* can be defined as:

$$Mother \equiv Woman \sqcap \forall hasChild.Person$$

which includes intersection, the concepts *Woman* and *Person*, the role *hasChild* and the value restriction $\forall$ on the role *hasChild*. Thus mother is a person having a child and a woman. More complex concept expressions can also be constructed, e.g. "a woman having at most two daughters":

$$WomanMax2Daughters = Woman \sqcap (\leqslant 2(hasChild \sqcap hasFemaleRelative))$$

Construct	Syntax	Language			
Concept	A	FL_0	FL^-	AL	S
Role name	R				
Intersection	C⊓D				
Value restriction	∀R.C				
Limited existential quantification	∃R				
Top or universal	⊤				
Bottom	⊥				
Atomic negation	¬ A				
Negation	¬C	C			
Union	C⊔D	U			
Existential restriction	∃R.C	E			
Number restrictions	(≥ n R) (≤ n R)	N			
Nominals	$\{a_1 \ldots a_n\}$	O			
Role hierarchy	R⊆S	H			
Inverse role	R^-	H			
Qualified number restriction	(≥ n R.C) (≤ n R.C)	Q			

Table 3.3: The various description logic languages (Gómez-Pérez et al. 2004).

which illustrates how the number restrictions on roles can be used. A set of definitions like the one above *Mother* and *WomanMax2Daughters* constitutes the Tbox (terminological box) of the knowledge base. The Tbox introduces the terminological knowledge, i.e. the vocabulary that describes the domain. In addition to the Tbox, a description logic knowledge base also contains an Abox (assertional box), which contains assertions about the world, e.g.:

$$Mother(Jane)$$
$$WomanMax2Daughters(Mary)$$

Based on this Abox, one could infer that *Mary* is also a *Mother* if the TBox contains the necessary definitions, and more generally, that *Mother* is the subsumer of *WomanMax2Daughters*.

3.2.3 On the choice of formalism

The expressive power of ONTOLOG as a lattice algebra extended with the Pierce product is limited compared to many of the description logic languages in that it does not include, e.g. quantifiers or negation. However, as noted, given the trade-off between expressiveness and tractability, the least expressive language, given the application at hand, should be chosen.

The basic assumption behind the work presented in this dissertation is that a vaguer notion of conceptual nearness or conceptual similarity, rather than inference of conceptual subsumption, can be successfully applied in information retrieval. This notion of conceptual nearness centers around viewing the ontology as a graph. This graph structure can naturally be inferred in description logic from the definitions of

concepts but is inherent to the lattice structure of the conceptual algebra defined by ONTOLOG. As it shall become evident, especially in chapter 6 on semantic similarity, the logical reasoning power behind the formalism used to represent the ontology is to a certain extent ignored in information retrieval because we are not interested solely in strict subsumption. For instance, a sibling concept can be of higher relevance than a subsumed concept far down a chain of subsumption relations. In general, if we perceive the ontology as a graph structure, we can chose to interpret nearness in the graph as an indicator of semantic relatedness. This perception of relatedness or similarity is in fact adopted in all the semantic similarity measures described in chapter 6. Thus, the focus in information retrieval can be shifted towards viewing the ontology as a graph structure which renders the reasoning power of the formalism of little importance. The graph structure is more inherent to the lattice algebraic framework used in ONTOLOG than in description logic.

Another difference between the description logics and the lattice algebra is the intended use or the *spirit* of the representation (Davis et al. 1993). In description logics the knowledge base is divided in a Tbox and an Abox. The Tbox contains descriptions of concepts and relations ranging from simple ones like *Mother* to more complex concept definitions like *WomanMax2Daughters*. Posing a query that in some form specified the notion of *WomanMax2Daughters* to a system, one would expect the result set to be all women having at most two daughters and nothing else. In chapter 2, this kind of retrieval is characterized as data retrieval as performed in, e.g. relation database systems. The spirit of description logic can thus be argued to be more akin to the ideas behind data retrieval than to the ideas and assumptions of uncertainty of information made in information retrieval, for instance, regarding partial matching. In other words, the spirit of description logics revolves more around building a knowledge base of precise facts than a knowledge base of vague and uncertain information. Because of this, a lattice algebraic framework seems more appropriate for the task at hand than description logic, though keeping in mind that a description logic language can also be used.

3.3 Resources

This section presents three different ontologies: WordNet, SIMPLE, and DOLCE. The experiments presented in later chapters involved WordNet and SIMPLE. WordNet was used to illustrate the suggested methodology on how to perform conceptual summaries as presented in chapter 8. SIMPLE was used in the experiments on the ontology-based disambiguation of semantic relations presented in chapter 5.

The three ontologies are presented in order of increasing level of formality. SIMPLE includes a much wider set of relations and a more strict notion of what constitutes word meaning than WordNet does, but as it is the case with WordNet it is a lexical ontology, and thus concepts and relations are modeled with a lexical offset. DOLCE is on the other hand a formal or heavy weight ontology which makes

formal distinctions that SIMPLE and WordNet does not. For instance, DOLCE distinct between endurants and perdurants and has also a much more formal notion of the instance-of relation. The purpose of presenting DOLCE here in conjunction with WordNet and SIMPLE is thus to give the reader examples of ontologies on the spectrum of level of formality presented by Lassila & McGuinness (2001) shown in figure 3.2 on page 31.

3.3.1 WordNet

WordNet is a large lexical database for English created at Princeton University by Miller and colleges (Fellbaum 1998*b*). The basic unit in WordNet is words, though it does, to a certain extent, contain idiomatic phrases, collocations, phrasal verbs and compounds. The four open word classes (nouns, verbs, adjectives and adverbs) are organized into four large separate semantic nets. In linguistics, an open word class is a word class where new items are added on a regular basis, through e.g. compounding, derivation, coining etc. A closed word class is a word class to which no new items are normally added e.g. determiners, conjunctions, and pronouns. In WordNet nouns are by far the most numerous with 117,798 noun word forms organized in 82,115 synsets (version 3.0). The synsets are semantically related to each other by e.g. hypernymy and meronymy (see below) thereby forming a semantic net of nouns.

A *synset* is a set of synonyms containing words which are interchangeable in some contexts. As noted previously, there is a difference between modeling concepts or world knowledge in an ontology and modeling word knowledge in a linguistic ontology. However, it is convenient to think of synsets as a form of lexicalized concepts, i.e. words that are used to denote a given concept. Many synsets are accompanied by a definition, *gloss*, explaining the notion behind the synset. The definitions are intended as an aid in distinguishing closely related synsets and as an aid for explaining uncommon concepts.

There are seven different kinds of relations between objects in WordNet, and the most fundamental from a linguistic point of view is *synonymy*, which glues together the synsets as the basic building blocks of WordNet. Synonymy is a symmetric lexical relation relating a single lexical unit to another lexical unit that can then be combined in synsets, e.g. {*dog*,*domestic dog*,*Canis familiaris*}. However, it is the other relations that link the synsets together that make WordNet interesting for many applications:

Hypernymy and its inverse, hyponymy, are the semantic generalization/ specialization relation that connects noun synsets in a tree-like structure that enables WordNet to be used, in some contexts, as an ontology, e.g. *mustang* ISA *pony* ISA *horse*. Hypernymy, also sometimes denoted as inclusion or subsumption, is a transitive asymmetric relation.

Meronymy and its inverse, holonymy, are a semantic part-whole/whole-part relation

connecting noun synsets. In principle, the meronymy relation can be used to form tree-like structures similar to hyponymy, though the network tends to be more tangled. Like hyponymy, meronymy is a transitive asymmetric relation, though its transitivity from a linguistic point of view is somewhat limited. For instance, it is reasonable to say *the handle is part of the door* and the *door is part of the house*, but saying **the handle is part of the house* sounds odd.

Entailment is when one verb entails another, e.g. *snoring* entails *sleeping*. Entailment is a transitive asymmetric semantic relation, e.g. *sleeping* does not necessarily entail *snoring*.

Troponymy is the verb synset equivalent of hypernymy and is a special kind of entailment where every troponym, v_1, of a more general verb, v_2, also entails v_2 ((Fellbaum 1998*a*). For instance, *march* is a troponym of *walk*, but *marching* also entails *walking*.

Antonymy is a symmetric lexical relation that denotes the opposition of meaning, for example, the nouns $victory <> defeat$ or the adjectives $fast <> slow$. Antonymy is clearly a relation between word forms and not concepts, since even though *fast* and *prompt* are similar, the latter is not antonymous to *slow*. Since antonymy is an important feature of adjectives, adjectives in WordNet are organized by a similarity relation forming related sets of adjectives with similar meanings. Adjectives in the different sets can then be related by the antonymy relation, so *slow* can be reached from *prompt* via *fast*.

Similarity is used in WordNet to link adjectives with similar meanings as noted under antonymy. In WordNet, similarity is a symmetric lexical relation, but in research on semantic similarity measures, the term similarity is most often used to denote a semantic relation between concepts connected by hypernymy and hyponymy. Here, similarity is a transitive relation that can be viewed as either symmetric or asymmetric. Chapter 6 covers this kind of similarity.

The relations used in WordNet are generalizations of a more fine-grained set of relations. The hypernymy relation in WordNet covers, e.g. both a formal and a telic hypernymy relation. For a fuller discussion of the different meronymy and hypernymy relations at work, see e.g. Miller (1998).

3.3.2 SIMPLE

The SIMPLE project was a research effort intended to create a semantic lexicon for twelve European languages, including Danish (Lenci, Bel, Busa, Calzolari, Gola, Monachini, Ogonowski, Peters, Peters, Ruimy, Villegas & Zampolli 2000, Lenci, Busam, Ruimy, Gola, Monachini, Calzolari & Zampolli 2000, Pedersen 1999). An important research objective of the project was to make the lexicons corpus based.

The SIMPLE project can be seen as an extension of the PAROLE project since the PAROLE corpora was used in the creation of the SIMPLE lexicons (Pedersen 1999, Braasch et al. 1998).

The project builds on Pustejovsky's (1991, 1995) work on generative lexicons where he introduces the notion of *generative* lexicons as opposed to *enumerative* lexicons. The central idea is that rather than enumerating all possible lexemes and their meaning, the meaning of lexemes is represented by their relations to other lexemes, i.e. by their *inheritance structure*. The perspective is thus one of compositionality in the sense that large structures can be described by their smaller components. The relational structure of a word is denoted as its *qualia* or qualia structure, and qualia can be thought of as a set of properties or events that explains what a word means (Pustejovsky 1995). The motivation for introducing a generative lexicon is, from Pustejovsky's perspective, to be able to account for the creative use of words and the permeability of word senses. Permeability meaning that word senses are not atomic but tend to overlap.

Based on Pustejovsky's ideas 1991, a word sense in SIMPLE corresponds to a semantic type that bears a cluster of semantic information (Pedersen 1999). This information can be in the form of a simple type or a unified type that is related to other semantic types. In order to describe the complex types, Pustejovsky's qualia structure is used, thereby achieving a richer structure than the one obtained using a one-dimensional subsumption hierarchy. Using the example of a *puzzle*, the qualia structure is composed of (Pedersen 1999):

- The *formal role*, which provides information about the positioning of the semantic type by its hypernymy relations to other semantic types, e.g. *a puzzle is a kind of game*.

- The *constitutive role*, which expresses a wide array of semantic relations, all describing the internal structure of the semantic types, e.g. *wooden or cardboard pieces are part of a puzzle*.

- The *telic role*, which describes the typical function of a semantic type, e.g. *a puzzle is used for assembling.*

- The *agentive role*, which concerns the origin of the semantic type primarily concerned with the origin of an entity, e.g. *a puzzle is produced*.

The top level above the different language specific lexicons is the SIMPLE Core Ontology. There are 151 different semantic types contained in the top ontology and formal, constitutive, telic and agentive are immediately under the top. Only the Core Ontology has been used for the experiments described in chapter 5.

3.3.3 DOLCE

The Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) ontology was created in the WonderWeb project that came to a close in 2004 (Masolo et al. 2003). One of the main goals of the WonderWeb project was to explicate the ontological choices being made when constructing ontologies and not to create a single monolithic top-level ontology. The reason for including this field of work in this chapter is to illustrate some of the challenges when moving from a lightweight to a heavyweight ontology. In addition, DOLCE has been used in Gangemi et al. (2003) to examine the ontological choices made in the creation of WordNet. Several of the issues arising from Gangemi et al.'s critique have, however, later been corrected in WordNet.

An important distinction in DOLCE is between universals and particulars. A universal is an entity or concept that has instances, e.g. *car*, whereas particulars are entities that cannot have instances, e.g. *my car*. In other words, particulars are instances of universals but universals can have other universals as instances, e.g. *car* is an instance of *type* in DOLCE. In the construction of WordNet, this distinction has not been drawn so the synsets contain both universals and particulars and there are several inconsistencies from a formal ontological viewpoint (Gangemi et al. 2003).

Another distinction drawn is between endurants and perdurants, which are sometimes also referred to as continuants and occurents. Endurants and perdurants can be distinguished by their behavior in time, where endurants are wholly present at any point in time and perdurants are only partially present at any point in time. To simplify, endurants are objects, e.g. President Bush, a piece of wood and Copenhagen, whereas perdurants are events, e.g. life, running and youth. Endurants and perdurants have a different unfolding in time and space since endurants are easy to place in space but are placed in time according to the perdurants they participate in. On the other hand, perdurants are easy to place in time but they can only be placed in space according to their participating endurants. For a discussion of endurants and perdurants, see e.g. Bittner et al. (2004) and Grenon & Smith (2004).

DOLCE is *an ontology of particulars* in the sense that it consists of related universals modeling a domain of particulars. As the words linguistic and cognitive in the name DOLCE indicate, it is not an attempt to model the intrinsic nature of the world, but rather as the world is perceived by human beings. As a result, no claim is made regarding its robustness in relation to cutting edge scientific research. Guarino & Welty (2000) have earlier developed *an ontology of universals* that formally models the domain of properties.

Figure 3.3 shows an example of how concepts in a given domain can be linked to DOLCE. As the concepts/universals of the domain, *car* and *traveler* are linked by a subclass relation to the middle triangle labeled "top level of particulars". The instance-of links illustrate the ontological ambiguity of the instance-of relation (Gangemi et al. 2001). *Car* and *traveler* are instance-of entities in the top-level ontology of uni-

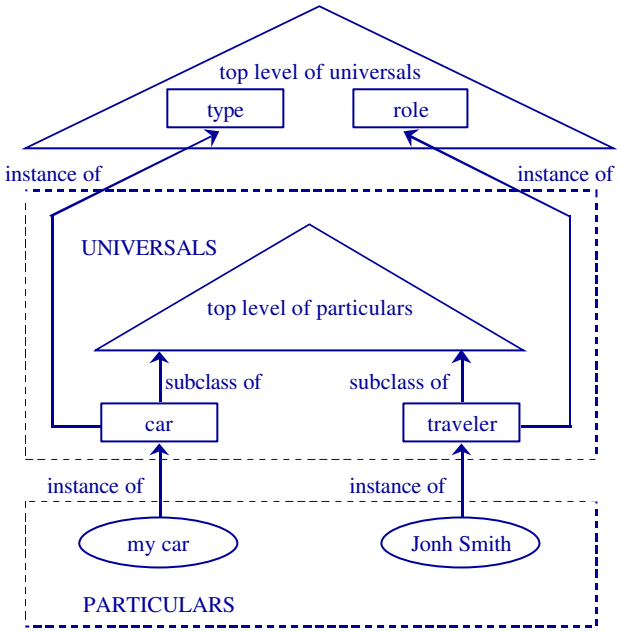


Figure 3.3: An illustration of DOLCE from López & Pérez (2002).

versals, whereas *my car* and *John Smith* are instances of car and traveler, respectively. In other words, they are all instances of something but at different ontological levels.

3.4 Discussion and Summary

This chapter has presented the notion of ontologies. Initially, different perspectives on what constitutes an ontology were illustrated by a spectrum of possible interpretations leading from informal structures of world knowledge to rigid formal specifications of conceptual knowledge. Different types of ontologies were also presented along with a more detailed description of the following three different ontologies: WordNet, SIMPLE and DOLCE. The first two of the three ontologies listed were presented because they were used in the experiments presented in chapters 5 and 8; WordNet will also be used in examples in the succeeding chapters. DOLCE was mainly presented because it has a much more formal basis, and because the research around DOLCE focuses on explicating the ontological choices made in the knowledge engineering process. The chapter also presented the lattice algebraic language ONTOLOG as an algebra for expressing and manipulating ontologies followed by a short presentation of the fundamentals of description logics. Most of what is presented in the subsequent chapters relies on viewing the ontology as a hierarchical graph that is inherent to the lattice algebra but not to the various description logics. More specifically, as will become clearer later on, it is not the logical reasoning sup-

plied by the different formalism that is in focus here, but the ability to use a graph in order to derive the conceptual similarity of a query rather than strict satisfiability. Therefore, ontologies expressed in ONTOLOG are well suited for forming the basis of an ontology-based information retrieval system as it is presented here.

Moreover, it was argued that a vaguer notion of conceptual nearness or conceptual similarity, rather than inference of conceptual subsumption, can be successfully applied in information retrieval. Much of the research on ontology-based information retrieval is thus based on the assumption that we can use the shortest path between two concepts as a measure of their similarity, and then use this measure in the matching of documents and queries. This assumption will also be a main thread throughout the remainder of this dissertation.

Chapter 4

The Ontoquery Project

Chapter one presented a schematic view of term-based information retrieval by describing the processes involved and how they are connected. In the indexing process, content is extracted in the form of descriptive terms, and later the description can be transformed into a representation suitable to the choice of information retrieval model. For instance, in the Vector Space model a weighted term-based description is represented as a vector where matching can be performed by vector-based similarity measures.

In ontology-based information retrieval, the fundamental building blocks of the system vocabulary are concepts. In this sense, the previous chapter on ontologies served as an introduction to the formal language or notation needed to express the system vocabulary of an ontology-based information retrieval system. This chapter will present some of the research performed within the ONTOQUERY project. Already at this stage it is important to note that the research within the ONTOQUERY project should not be perceived as having the goal of presenting a unified approach to ontology-based information retrieval, but rather the research is characterized by different contributions within a common conceptual framework. A special emphasis will thus be put on the areas of particular importance to the contributions presented in this dissertation. First the previous work done within content analysis will presented based on Andreasen et al. (2001, 2002), Andreasen & Nilsson (2004), and Andreasen et al. (2004), and subsequently the previous work measures of semantic analysis will be presented based on Knappe et al. (2007), Andreasen et al. (2005*a*, 2003), Bulskov et al. (2004), and Bulskov et al. (2002).

4.1 Content Analysis

Content analysis, semantic analysis, or semantic parsing as it is also labeled is a process in which natural language is related to a formal representation of meaning. In content analysis a coarse distinction can be made between deep and shallow ap-

proaches. In deep semantic parsing, the goal is a close to full understanding of the semantics of the document. A prerequisite for achieving a high level of understanding in deep semantic parsing approaches is the ready access to a vast body of linguistic and world knowledge. We need to know, for instance, how sentences are structured, what the concepts are and how they are semantically related. However, bodies of linguistic and world knowledge are scarce resources and the knowledge embedded within them is sometimes difficult and complex to apply on a larger scale. Hence, deep parsing has almost exclusively been applied in small, well-structured domains. Shallow semantic parsing, on the other hand, presumes less linguistic and world knowledge and aims at a level of understanding where known concepts are identified and potentially important relations are recognized. To a certain extent, the following simple example illustrates the difference between deep and shallow approaches: *Peter is in the restaurant where he is finishing his pizza*. A deep semantic understanding of this sentence would allow questions like who is eating the pizza to be answered. Solving the anaphoric reference shows that *Peter* is eating the pizza and not just *he*. A deep semantic understanding also makes deducing what they serve at the restaurant possible. Pizza is edible, and since it is being consumed at the restaurant, we presume the restaurant is serving the pizza. On the other hand, a shallow understanding of the sentence would simply be that Peter is in the restaurant, and someone, anyone, is eating pizza. As stated, the distinction is coarse, but as the example demonstrates, there is a distinction to be made.

In the ONTOQUERY project, content analysis is performed through both lexical and conceptual analysis in that it incorporates both simple linguistic heuristics and ontological knowledge in the process. With respect to indexing, the aim has been to recognize possibly compound concepts expressed as noun phrases, and to ensure that noun phrases with almost identical conceptual content, but with potential different lexicalizations, are described identically. The aim of the analysis has been to facilitate an ontology-based information retrieval that utilizes a lattice structure of the ontology for matching rather than strict logical inference described in chapter 3. Therefore, the perspective on noun phrase analysis presented here is limited to information retrieval rather than being an elaborate account of noun phrase analysis in general.

Noun phrases in OntoQuery are represented in ONTOLOG, which is naturally able to represent compound concepts. For instance, the phrases "lack of vitamin D", "deficiency with respect to vitamin D" and "vitamin D deficiency" can all be represented by the descriptor *lack*[WITH RESPECT TO: *vitamin D*]. The general form of a descriptor is shown in equation 4.1 where c is a concept name, r_i is a relation, and d_i is a descriptor that can be a compound descriptor or a simple concept name:

$$c \left[\begin{array}{ccc} r_1 & : & d_1 \\ \vdots & & \vdots \\ r_n & : & d_n \end{array} \right] \tag{4.1}$$

Hence, both nested and non-nested multiple prepositional phrases can be described

as presented in detail in the section on ONTOLOG in chapter 3. In short, a descriptor is the result of a content analysis, in this case, a noun phrase represented as an ONTOLOG expression.

4.1.1 Analyzing noun phrases

In order to create descriptors for compound an anterior analysis of the semantic relations connecting the constituents is performed. This conceptual analysis of noun phrases is facilitated by acquiring syntactic knowledge about the structure of the noun phrases, since the syntactic structure, to a certain extent, exposes the semantic structure of the noun phrase (for a more general discussion of this assumption, see e.g. Nirenburg & Raskin (2004)). Leaving out determiners, the head of the noun phrase can be modified by pre-modifiers and post-modifiers, which roughly corresponds to adjective phrases and prepositional phrases, respectively. Since the constituents of adjective phrases and prepositional phrases typically relate in different ways, the syntactic structure of noun phrases thereby serves as a clue as to how the constituents are related.

With respect to adjective phrases, WordNet, for example, divides adjectives into two major classes, descriptive and relational, which is roughly equivalent to the distinction drawn in SIMPLE (Mendes 2006, Peters & Peters 2000). In WordNet, descriptive adjectives are organized, like nouns, into synsets, and they are related to other adjectives at the lexical level by the antonymy relation. Descriptive adjectives like "beautiful", "fast", and "tall" characterize the nouns they modify, e.g. "a beautiful house", "a fast car", and "a tall man". Ontologically, adjective phrases like these are therefore, in the ONTOQUERY approach, chosen to be described by a *characterized by* relation, i.e. *man*[CHR : *tall*].

The relational adjectives, which are much fewer in number, are also organized in synsets, but there is usually only a single adjective in each synset, and they are not related to other adjectives by the antonymy relation. Instead, there is a pointer to the noun they pertain to, e.g. "environmental" points to "environment" and "chemical" points to "chemistry". The relation "pertains to" is modeled by the relation "with respect to" (WRT) and ontologically, a phrase like "a chemical engineer" is therefore described by the descriptor *engineer*[WITH RESPECT TO : *chemistry*]. Although the distinction between descriptive and relational adjectives is not always clear, the approach sketched here solves the ambiguity to a certain extent, and functions in ONTOQUERY as a heuristic.

Compared to adjective phrases, prepositional phrases are much more difficult to analyze because of the wide range of relations between the constituents that can be denoted by prepositions. Table 4.1 contains some common Danish prepositions and some of the relations they can denote as well as an example text excerpt (Jensen & Nilsson 2003).

The ambiguity of the denoted relation can be handled in different ways. A sim-

Preposition	Role set	Example	Gloss
af	AGT	Behandling af læge	Treatment by a physician
	PNT	Behandling af børn	Treatment of children
	POF	Siden af hovedet	The side of the head
	MAT	Pude af læder	Leather cushion
i	LOC	Betændelse i øjnene	Inflammation of the eyes
	TMP	I to dage	For two days
	POF	Celler i øjet	Cells in the eye
med	BMO	Behandling med medicin	Treatment with medicine
	CHR	Børn med diabetes	Children with diabetes
fra	SRC	Blødning fra tarmen	Intestinal haemorrhaging
	TMP	Fra sidste år	From last year
	POF	En agent fra CIA	An agent from the CIA

Table 4.1: Examples of some of the relations four common Danish prepositions can denote. The complete set of relations proposed by Nilsson (2001) can be seen in the previous chapter on page 35.

ple solution is to express a generic relation, rel, and generate descriptors of the form, $c[\text{REL}: d]$, e.g. "treatment of children" can be described by *treatment*[REL: *children*], indicating that the nature of the relation is unknown. This preserves the information from the syntactic analysis, i.e. that the preposition expresses some kind of relation between c and d. A second alternative is to describe the noun phrase using the different possible interpretations of the relation denoted by the particular preposition:

$$\{\textit{treatment}[\text{PNT}: \textit{children}], \textit{treatment}[\text{POF}: \textit{children}], ...\}$$

A third possibility, which is an extension of the previous alternative, is to reduce the ambiguity by ruling out ontologically inadmissible readings or by indicating likely readings. The ability to rule out ontologically inadmissible readings could be achieved, e.g. by some form of selectional restrictions or the specification of an ontological grammar (Jensen & Nilsson 2003, Jensen et al. 2001). For instance, the Danish preposition *af* (of) sometimes expresses a *part of* relation, e.g. "The side *of* the head". However, in the phrase "decomposition *of* material" we can rule out the reading **decomposition*[PART OF: *material*] because a *material* cannot be part of a *process*. Chapter 5 explores the possibility of indicating likely ontological readings of prepositional phrases by using machine learning and an annotated corpus.

4.1.2 Description and descriptors

In OntoQuery, descriptions are initially derived at the sentence level, since the syntactic analysis required to perform noun phrase recognition presupposes a sentence structure. This is somewhat different from the approaches presented in the chapter on information retrieval where descriptions were derived at document level. However, since the nesting of descriptions can be infinite, the question concerning the level of description is a purely practical matter (see e.g. Andreasen & Bulskov (2007*b*)).

In the ONTOQUERY project, descriptions, D, are sets of sets of descriptors, d, and since descriptions can be nested, the description of a sentence takes the following form:

$$D = \{D_1, \ldots, D_2\} = \{\{d_{11}, \ldots, d_{1m_1}\}, \ldots, \{d_{n1}, \ldots, d_{nm_n}\}\} \tag{4.2}$$

A set of descriptors, D_i, corresponds to a noun phrase, NP_i, in the sentence. Depending on our system's ability to create accurate descriptions through the noun phrase analysis, the description can exhibit varying levels of accuracy. The most accurate description is a single descriptor for each noun phrase, i.e. $D_i = \{d_i\}$ rather than $D_i = \{d_{i1}, ..., d_{in}\}$.

Consider the phrase, "dietary treatment and disorder due to lack of vitamin D". This phrase can be described in several possible ways where the most general is a description which consists of all the concepts in the noun phrase:

$$D = \{\{dietary\}, \{treatment\}, \{disorder\}, \{lack\}, \{vitaminD\}\} \tag{4.3}$$

For a more accurate description, concepts can be grouped according to the noun phrase in which they occur. This grouping corresponds to expressing the existence of a generic relation between the constituents as described in the previous section:

$$D = \{\{dietary, treatment\}, \{disorder\}, \{lack, vitaminD\}\} \tag{4.4}$$

If it is possible to determine the nature of the relation between the constituents, an even more accurate description can be created:

$$D = \{\{\textit{dietary}[\text{CHR}: \textit{treatment}], \{disorder[\text{CBY} : lack[\text{WRT} : vitaminD]]\}\} \tag{4.5}$$

The generative aspect of the ontology makes the creation of a single descriptor of a compound concept possible. Although emphasis has been put on the analysis of noun phrases, the formalism puts no constraints on the words included in the initial description. In a practical setting, the description, D, would most likely also include, e.g. verbs.

4.2 Similarity measures

Based on the need for matching descriptions of document and query the question of similarity emerges. How does one measure similarity between two descriptors? A wide range of similarity measures and their basis are considered in chapter 6 but here we will present the research on similarity measures within the ONTOQUERY project.

The two main contributions within similarity measures has been with weighted shortest path presented e.g. in Bulskov et al. (2002) and (weighted) shared nodes presented in e.g. Bulskov et al. (2004), and Andreasen et al. (2003).

4.2.1 Weighted Shortest Path

In information retrieval, the specializations of concepts appearing in a query can be argued to be more relevant than generalizations (Knappe et al. 2007). For instance, in a search for *motorcycle*, documents where specializations of motorcycles such as *offroader* or *cruiser* appear are interesting because they are subsumed by *motorcycle*. On the other hand, documents where *motor vehicles* appear are less interesting, because they can cover topics other than *motorcycles*, such as *cars*, *trucks* or even *tanks*. Following this assumption, a similarity measure must therefore be asymmetric, i.e. resulting in *offroader* being more similar to *motorcycle* than *motorcycle* to *offroader*. A measure that adheres to this idea is the weighted shortest path measure where a smaller similarity is given to steps leading upwards in the ontology (along the hypernymy relation) than to steps leading downwards (along the hyponymy relation) (Bulskov et al. 2002). The principle of weighted shortest path is illustrated in figure

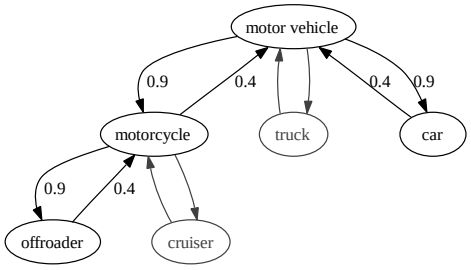


Figure 4.1: An illustration of the weighted shortest path measure of offroader and car with $sim(offroader, car) = 0.4 \times 0.4 \times 0.9 = 0.14$ and $sim(car, offroader) = 0.4 \times 0.9 \times 0.9 = 0.32$.

4.1, where direct hypernymy weights are set at 0.4, direct hyponymy weights are set at 0.9 and weights are multiplied rather than summed up.

4.2.2 Shared nodes

The weighted shortest path approach described above is straightforward, but only the shortest path contributes to the similarity and all other paths are ignored. In e.g. Knappe et al. (2007) it is argued that it must be assumed that all the paths contribute to the similarity. Consider for instance the three following text excepts and their descriptors in ONTOLOG :

1. The black cat - *cat*[CHR : *black*]
2. The brown poodle - *poodle*[CHR : *black*]
3. The poodle in the garden - *poodle*[LOC : *garden*]

The intuition is that text except (1) and (2) has more in common than text except (1) and (3) because they are connected by their color attribution. This naive example shows that ideally a similarity measure should take into account all possible paths connecting two descriptors. However, in large ontologies, computing all possible paths between any combination of two concepts is not a realistic approach. Instead Bulskov et al. (2004), and Andreasen et al. (2003) suggest a similarity measure that reflects all possible paths by comparing so-called shared nodes without actually deriving paths. The set of shared nodes is the set of all nodes in the ontology graph that are hypernyms of both concepts. Fig. 4.2 taken from Andreasen et al. (2003) shows an example of the set hypernyms {*anything*,*animal*,*color*} that are both the hypernyms of *cat*[CHR : *black*] and *poodle*[CHR : *black*].

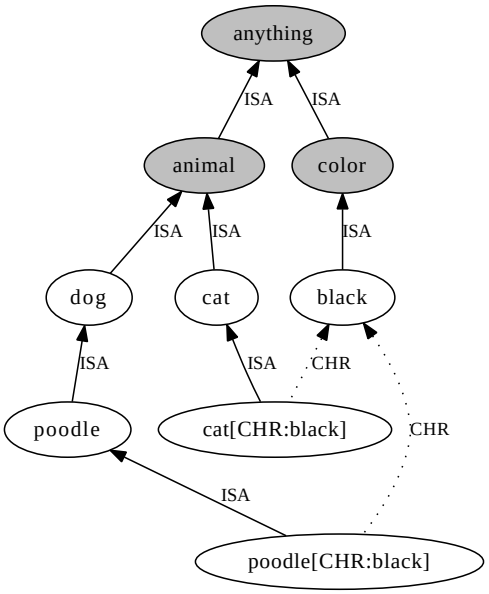


Figure 4.2: An example of the shared nodes for the descriptors *cat*[CHR : *black*] and *poodle*[CHR : *black*] from (Andreasen et al. 2003). Shared nodes are shaded.

With $\alpha(x)$ as the set of hypernyms of x, Andreasen et al. (2003) presents the different variations of a normalized similarity measure based on the shared nodes principle with one of them being:

$$sim(x,y) = \frac{|\alpha(x) \cap \alpha(y)|}{|\alpha(x) \cup \alpha(y)|} \tag{4.6}$$

$$\tag{4.7}$$

In Andreasen et al. (2005*a*) a fuzzy generalization termed *weighted* shared nodes is introduced. In weighted shared nodes the concepts are weighted based on how they are related to the original descriptor, e.g. ISA can be higher weighted than CHR.

4.3 Related Work

The content analysis presented by the ONTOQUERY project is related to both biomedical term identification (Krauthammer & Nenadic 2004, Zhou et al. 2006), and key phrase indexing (Jacquemin & Tzoukermann 1999, Jacquemin & Bourigault 2003). Jacquemin & Tzoukermann (1999) presents an approach to conflating multi word terms like *signal frequency controllers*, *frequency controllers for signals*, and *control of frequency for signals* into a single descriptor. The presented approach is based an inflectional analyzer for single word conflation, i.e. a form of stemming, and syntactic rules focused on noun phrase analysis like the heuristics presented in 4.1.1. Compared to Jacquemin & Tzoukermann (1999) the ONTOQUERY approach differs in the central role played by the ontology. Both as a mean of representing the complex concepts expressed by noun phrases and as a mean for analyzing these. The next chapter will further explore one approach to utilize the ontology in the analysis.

Due to the constant creation of new multi word terms in the biomedical literature much research has focused on how to extract and identify these terms Krauthammer & Nenadic (2004). In the context of ONTOQUERY the work by Zhou et al. (2006) is interesting because it is characterized by the use of an ontology as a vessel for identifying concept pairs. Concept pairs are by Zhou et al. defined as pairs of concepts sharing both a semantic relation specified by the ontology and a reoccurring syntactic relation. If these two characteristics are met the concept pair is identified as a single new concept to be used in the indexing. Compared to ONTOQUERY the type of relation is not determined, and neither is the ontology used actively in the analysis. The experimental results presented by Zhou et al. are encouraging, and they show a considerable improvement in precision on the TREC 2004 Genomics Track.

Similarity measures will be treated in a detail in chapter 6 so related work on this topic will be described there.

4.4 Discussion and Summary

This chapter presented the approach to ontology-based indexing forged within the ONTOQUERY project. The focus of this approach is an analysis which ensures that noun phrases with almost identical conceptual content, but with possibly different lexicalizations, are described identically. For instance, the text excerpt "lack of vitamin D", "deficiency with respect to vitamin D" and "vitamin D deficiency" can all be represented as *lack*[WRT: *vitamin D*]. Also it was described that a sentence can be analyzed and represented at various levels of specificity depending on the system's ability to perform the appropriate natural language processing needed, e.g. the ability to identify the relations between the concepts. However, in order to determine that "lack of vitamin D" can be represented as *lack*[WRT: *vitamin D*] a deeper content analysis than the one presented in this chapter is needed. This content analysis must be is able to recognize that *of*, in this case, denotes the relation WRT and not

for instance a *comprised of*, CMP, relation as it does in the text excerpt "depot of Vitamin D" (*depot*[CMP: *vitamin D*]). The need for a deeper content analysis is the motivation for the work presented in the next chapter. With an offset in prepositional phrases an approach is presented which, with a high degree of precision, is able to correctly recognize the type of relation being expressed. This greatly improves the quality of the ontological indexing.

The second part of the ONTOQUERY research which were presented in this chapter was the work on ontology-based similarity measures. Both the weighted shortest path and the shared nodes approach was described. All of chapter 6 is devoted to similarity measures, and thus there is no need for a lengthy discussion of the topic here. There are though a commonality and a difference between the previous contributions within ONTOQUERY and the contribution presented in chapter 6. Both weighted shortest path, weighted shared nodes presented here, and the distributional density measure presented in chapter 6 are based on the acknowledgement that some relations are of higher importance than others. As chapter 6 will show, the principle that some relations are more important than others in measures of similarity, is in fact acknowledged in most of the previous research on the topic. What differs is how to value the different relations in the ontology. Here are the measures introduced in the final part of chapter 6 novel because they combine patterns of co-occurrence of concepts with ontology-based similarity measures.

This chapter concludes the foundation part of this dissertation. Chapter 2 presented the different components in a prototypical information retrieval system, the different information retrieval models, and an introduction to ontology-based information retrieval. Here two of the main reasons for research on ontology-based information retrieval was described. First reason was that pure lexical-based information retrieval is inept to deal with the ambiguity of natural language. Second that ontologies enables the retrieval of semantically related information. As mentioned above the next chapter will describe an approach to semantic analysis dealing with the ambiguity of natural language. The chapter hereafter, chapter 6, presents three different but highly related similarity measures that continues on the same thread as the ones presented in this chapter, but are novel in their combination of ontology-based similarity measures and co-occurrence based measures of similarity. These measures can be used in the previously unpublished model for index expansion presented in chapter 7. This model enables the retrieval of documents that are semantically related to the query rather than only lexically related. Finally the contribution part will end with chapter chapter 8 which will present a way of summarizing the results of a query in the form the of a modified ontology in itself rather than a set of documents or a summary in natural language.

Part II

Contributions

Chapter 5

Finding Semantic Relations Expressed in Natural Language

Semantic analysis is the study of meaning. For example, what meaning do the linguistic utterances "lack of vitamin D" and "deficiency with respect to vitamin D" share? *Lack* and *deficiency* clearly convey the same meaning, as do *of* and *with respect to*. From an ontological perspective, *lack* and *deficiency* denote the same concept and *of* and *with respect to* denote the same *semantic relation*. *Of* and *with respect to*, however, do not always denote the same relation, e.g. *of* denotes an entirely different relation in "soup of the day". The identification of the different semantic relations at play in a text is an important part of the extraction of conceptual content in ontology-based information retrieval.

As described in the previous chapter, the intent of the ONTOQUERY project has been to devise a semantic analysis of noun phrases that ensures that phrases with identical meaning or conceptual content, but with different lexicalizations, are described identically. The motivation for doing so is the objective of ontology-based information retrieval to retrieve documents that convey the same meaning but are lexicalized differently. This chapter describes a preliminary approach to the conceptual extraction of the relations denoted by prepositions in constructs like "lack of vitamin D". The presented approach enables a much more accurate analysis of such constructs than previously has been attempted. By this approach, more specific descriptors of the document and query content can be created, thus enabling more precise ontology-based information retrieval. It is shown that knowledge at a general level about the participating concepts is sufficient in order to determine the relation type with high precision. For instance it is sufficient to know that "vitamin D" is a *natural substance*. A less elaborate account of this research has already been presented in Lassen & Terney (2006*a*,*b*).

Prepositions have been characterized as "semantically vacuous and distributionally highly promiscuous" (Baldwin 2006), and, indeed, in most keyword-based search engines, prepositions are considered to bear little meaning. Admittedly, prepositions

are some of the most frequent words and their use and meaning are not confined to a limited number of domains or text types. Therefore, in a classic keyword-based approach, it makes sense to exclude prepositions either before the indexing via a list of stop words, or in the indexing via a threshold leaving out all words with low discrimination value. However, in ontology-based information retrieval, we are moving away from the keyword-based approach and have begun looking at the relations between concepts. As the vitamin example above demonstrates, a preposition can denote different kind of relations, and identifying the current relation in a text might be valuable in information retrieval.

Consider, for instance, the following example. In the query "treatment of cancer", there are two nouns and a preposition denoting a relation between them. In this case, the preposition *of* expresses a patient relation between treatment and cancer (more on relations in the following section). Texts dealing with, for example, what effects the development of cancer could be considered relevant to this query. Consider the following two text excerpts: "...nutrition has a high effect on cancer..." and "...weight loss can occur as a result of cancer...". In the first case, *on* expresses a patient relation between *effect* and *cancer* and should therefore be ranked high in the result set. In the second case, because *of* denotes a source relation between *result* and *cancer*, this text excerpt is of less relevance than the first. The question this chapter tries to answer is the extent to which the type of concepts expressed in noun phrase-preposition-noun phrase (NP-P-NP) constructs can serve as a clue as to what kind of relation the preposition denotes. If this in fact is the case, ontologies can be used in the semantic analysis of these constructs despite the "promiscuous" nature of prepositions.

Our assumption behind the work presented is naturally that there is an affinity between the concepts denoted by the heads of the noun phrases and the relation denoted by the preposition. For instance, if the first noun is a kind of *disease* and the second noun is a *body part*, then *in* very likely expresses a locative relation, e.g. "thrombosis in the heart", "oedema in the legs", "cancer in the uterus", etc. If the second noun, on the other hand, has something to do with *time*, *in* expresses a temporal relation, e.g. "plague in the Middle Ages", "vitamin deficiency in childhood", etc.

The following steps were taken to examine this assumption. First, NP-P-NP excerpts in Danish were found. Second, the head of the noun phrases was mapped into SIMPLE and the semantic relation denoted by the preposition with a relation type from a finite set of relations was annotated. Last, a machine learning approach was used to explore the affinity between the preposition and the concepts denoted by the heads of the surrounding nouns, i.e. the relation was classified based on the surrounding context. In a sense, these experiments are a type of word sense disambiguation; we want to discover on a semantic level what kind of relation a preposition denotes given its context. The following section presents each of the three steps taken in the study and the results of the experiments.

5.1 Semantic relations

In general, relations can exist between all entities referred to in discourse and at different syntactical levels across sentence boundaries, or within a sentence, a phrase or a word. The relations can be denoted by different word classes, such as a verb, a preposition or an adjective, or they can be implicitly present in compounds and genitive constructions. In the experiments, only binary relations denoted by prepositions are considered. A preposition can be ambiguous in regard to which relation it denotes. Consider, for example, the Danish preposition *i* (Eng: in): The surface form *i* in "A i B" can denote at least five different relations between A and B:

1. A patient relation (PNT): A relation where one of the arguments' case roles is a patient, e.g. *"ændringer i stofskiftet"* (changes in the metabolism).

2. A locational relation (LOC): A relation that denotes the location/position of one argument compared to another argument, e.g. *"skader i hjertemuskulaturen"* (injuries in the heart muscle).

3. A temporal relation (TMP): A relation that denotes the placement in time of one argument compared to another, e.g. *"mikrobiologien i 1800-tallet"* (microbiology in the nineteenth century).

4. A property ascription relation (CHR): A relation that denotes a characterization relation between one of the arguments and a property, e.g. *"antioxidanter i renfremstillet form"* (antioxidants in a pure form).

5. A "with respect to" relation (WRT): an underspecified relation that denotes an "aboutness" relation between the arguments, e.g. *"forskelle i saltindtagelsen"* (differences in the salt intake).

The set of possible semantic relations is in principle infinite, ranging from more general relations denoting relationships between general concepts or sets of concepts to very finely grained relations between very specific concepts. An attempt to arrive at a general complete list of relations is, therefore, futile. In the research presented here, semantic relations are to be perceived as general thematic roles linking concepts in the world. For instance, with an example taken from Dowty (1991), consider three situations where a murders b, a accuses b and a interrogates b. Even though the three verbs have very different meanings, they all share that a commits a volitional act, that committing the act is a's intention, and that a causes some event to take place involving b. Expressed using thematic roles, a can be said to be the *agent* in that he deliberately performs the act, while b can be said to be the *patient* in that he undergoes the action and his state is changed.

As a basis for these experiments, we selected the set of relations proposed by Jensen & Nilsson (2003) in their work on ontology-based semantics for prepositions.

This set of relations is summarized in table 5.1 and, for the most part, is exemplified in table 5.2, which is also shown in chapter 3.

Role Relation	Abbreviation	Description
TEMPORALITY	TMP	Temporal anchoring, duration, inception etc.
LOCATION	LOC	Place, position
WITH RESPECT TO	WRT	With respect to
CHARACTERIZE	CHR	Characteristic (Property ascription)
BY MEANS OF	BMO	Means to end, instrument
CAUSED BY	CBY	Inverse CAU
COMPRISE	CMP	Inverse POF, whole constituted of parts
AGENT	AGT	Animate being acting intentionally
PATIENT	PNT	Affected entity, effected entity
SOURCE	SRC	Source, origin, point of departure

Table 5.1: The set of relations found expressed in the data set used in the experiments presented in this chapter.

Besides the relations in table 5.1, a WRT relation was introduced. The annotation with WRT was done in the appropriate cases and when none of the other relations denoted the relation being expressed.

5.2 Corpus and Annotation

Most natural language processing experiments are performed on English texts where resources like ontologies are much more abundant and easily accessible. One important goal of this research is to contribute, in a small way, to the advancement of research on the Danish language, hence conducting these studies in Danish has been an essential aspect of this project.

A small tagged corpus was obtained from Hansen (2005) that was compiled from

Preposition	Role Set	Example	Gloss
af	AGT	Behandling af læge	Treatment by a physician
	PNT	Behandling af børn	Treatment of children
	POF	Siden af hovedet	The side of the head
	MAT	Pude af læder	Leather cushion
i	LOC	Betændelse i øjnene	Inflammation of the eyes
	TMP	I to dage	For two days
	POF	Celler i øjet	Cells in the eye
med	BMO	Behandling med medicin	Treatment with medicine
	CHR	Børn med diabetes	Children with diabetes
fra	SRC	Blødning fra tarmen	Intestinal hemorrhaging
	TMP	Fra sidste år	From last year
	POF	En agent fra CIA	An agent from the CIA

Table 5.2: Examples of some of the relations four common Danish prepositions can denote.

The Danish National Encyclopedia on nutrition (Lund 1994). From this corpus, NP-P-NP excerpts were extracted based on the original part of speech tagging. Subsequently, the head of the nouns was mapped automatically (ontology look-up) and manually (for heads not present in the ontology) to the concepts in SIMPLE. However, it quickly became evident that SIMPLE's coverage was a problem since only a fraction of the heads were to be found in the ontology. As a result, the heads were mapped to the most specific concepts in the core ontology of SIMPLE, primarily to avoid the ontology engineering process involved in finding out exactly where to position the missing concepts in the ontology. Figure 5.1 illustrates, e.g. how *thrombosis* maps to *disease* by the transitivity of hypernymy, since *disease* is the most specific concept that thrombosis can be mapped to in the top ontology of SIMPLE.

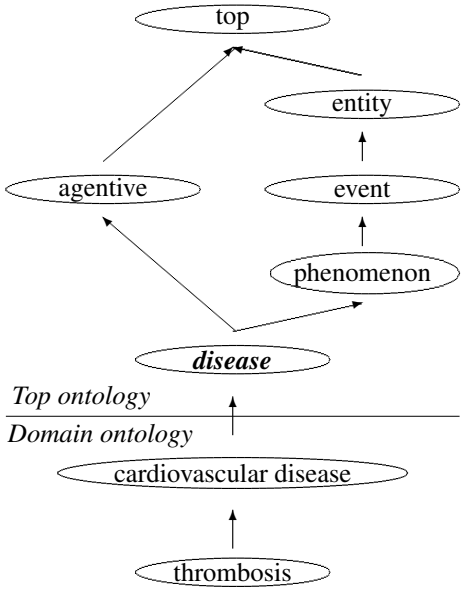


Figure 5.1: An illustration of the path from *thrombosis* to the top level of the SIMPLE ontology. If *thrombosis* had not been in SIMPLE, it would have been mapped directly to *disease* because *disease* is the most specific concept in the core of SIMPLE that subsumes *thrombosis*.

Table 5.3 shows the annotation of the corpus by listing the text excerpt and the conceptual content of the text excerpt as an ONTOLOG expression. For instance, *sammensætningen* (composition) maps to *constitutive state*, *kosten* (food) to *food*, *påvirkning* (effect) to *cause change*, etc.

5.2.1 Descriptive statistics of the corpus

The compiled corpus consists of 952 text excerpts or *instances*, giving a total of about 18,500 running words. In general, the distribution of the data is highly skewed and

Text excerpt	Annotation
sammensætningen af kosten (the composition of the food)	*constitutive state*[WRT : *food*]
påvirking af huden effect on the skin	*cause change*[PATIENT : *body part*]
undersøgelser af kvælstofbalancen examination of the nitrogen balance	*act*[WRT : *state*]
person på 80 kg person at 80 kg	*human*[CHR : *unit of measurement*]
virkning på hjernen effect on the brain	*cause change*[PNT : *body part*]

Table 5.3: Examples of text excerpts from the corpus with their annotation.

data sparseness is a serious problem. The data consists of the concepts denoted by the heads of the phrases, the lemmatized heads, the prepositions and, finally, the relation denoted by the preposition.

The following account gives an idea of the distribution of the data: Of the 74 distinct concepts of the first head of the phrases, 23 are unique, and six concepts account for slightly fewer than half the instances (act, change, state, natural substance, physical property, and creation in that order). Of the 64 distinct concepts of the second head of the phrases, 20 are unique and only four concepts account for approximately half of the instances (natural substance, body part, disease, human). Looking at the combinations of the concepts, there are 332 different combinations and 197 of them are unique. In short, most types of combinations only appear once or just a few times, which makes the data set difficult to visualize.

At the word level, there are 443 distinct lemmas as first heads, with 324 occurring only once as the first head, and 482 distinct lemmas as the second head, with 343 occurring only once as a second head. Compared to the ontological level, the mass of unique lemmas constitutes about 37 percent, where only about two percent of the concepts were unique. Compared to the conceptual level, the data set is obviously even sparser at the word level.

Slightly more than half of the instances are of the relation type WRT or PNT, and the rest of the instances are distributed among the remaining ten relations with only 14 instances scattered across the three smallest classes. The distribution is shown in figure 5.2.

The same skewness in distribution is true for the distribution of the prepositions. Here, *af* (of) and *i* (in) also account for more than half the instances. The distribution of each of the 15 relations found in the corpus is shown in figure 5.3.

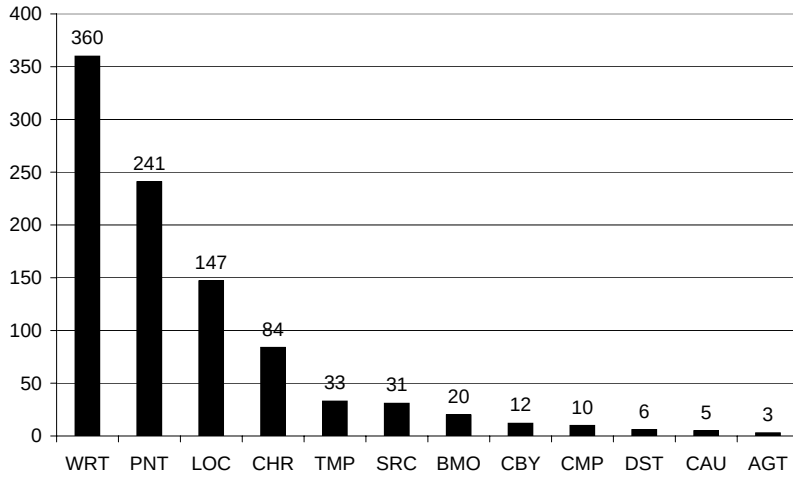


Figure 5.2: The distribution of the 12 relations expressed in the corpus.

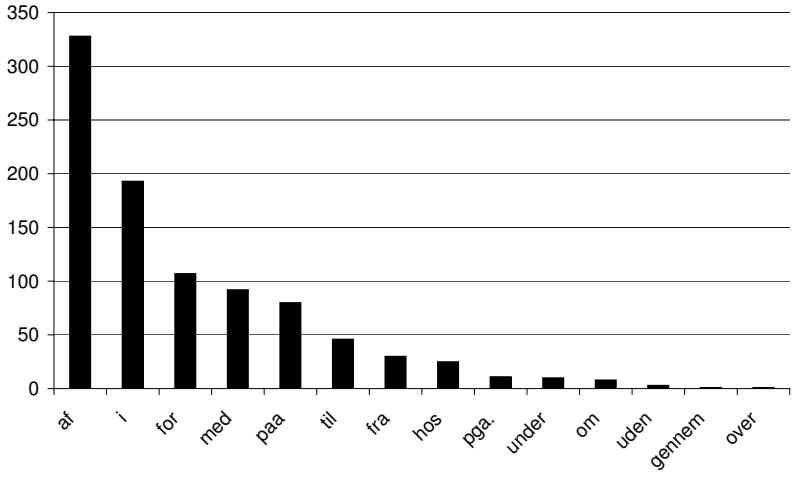


Figure 5.3: The distribution of the 15 prepositions in the corpus.

5.3 Machine Learning

Machine learning is a broad subfield within artificial intelligence where the purpose is to design a learner, i.e. a system or an algorithm that improves with experience concerning a given task with respect to some performance measure (Mitchell 1997). The task can be anything from driving an autonomous vehicle based on sensory input, to identifying patients likely to catch a specific illness based on patient records or finding trends in consumer purchases. The motivation for applying machine learning to all these tasks is that they are either impossible or too cumbersome to solve manually – or that computers simply perform better.

Machine learning can be subdivided into *supervised*, *unsupervised*, and *reinforcement learning*. Supervised learning is the situation where the data set consists of a number of *instances*, where each instance is *classified* according to some function. The function can, for instance, be to determine the kind of customer based on the content of their shopping basket. If the basket contains *diapers*, *plasticine*, and *organic milk*, the consumer is probably a parent with small children. On the other hand, if the basket contains *Eurowoman*, a bottle of *Evian*, and *rice biscuits*, the customer is probably a young female. In supervised learning, training the learner would consist of presenting a number of customers with their shopping baskets to make the learner capable of recognizing the different characteristics of the different customers. Because the learner is presented initially with the correct label or classification of each instance, in this the items in the shopping basket and the customer who bought the items, the learning is termed supervised learning. Some kind of supervisor is necessary to create the data set of labeled instances by identifying each instance as a *parent*, *young female*, etc. The classification of the supervisor is often denoted the *true function*. The precision of the result produced by a learner is measured by the difference between the classification by the true function and the classification by the learner.

In unsupervised learning, there is no supervisor present in the form of labeled instances, and the causal connections explored between input and output in supervised learning are focused upon less in unsupervised learning. The objective is typically to explore the data set using various statistical measures; for example, in the classical market analysis where supermarkets try to identify the correlation between sales of different goods to optimize the placement of the goods. *Beer* and *diapers* are often mentioned as a surprising correlation between everyday goods often bought at the same time. *Clustering*, which will be a main component of chapter 8, is another example of an unsupervised task where instances are grouped based on their similarity. Again, using the shopping basket example, a clustering task could be to group customers based on the content of their shopping baskets. The result of such a clustering process could then be presented to the market analysts in order to verify and label the groups found.

Finally, reinforcement learning is a dynamic setting where the output of the learning system serves as an input to the next iteration of learning, and where the desired outcome is the result of a series of successful steps instead of a single step. Also, the value of taking each step is not known. Examples of reinforcement settings are, e.g. autonomous agents moving around in a specific environment and in games like chess.

Since the aim here is to explore to what extent semantic relations denoted by prepositions in NP-P-NP constructs can be learned, the setup of the experiments took place within the supervised framework.

5.3.1 Symbolic and non-symbolic learners

Supervised learning algorithms can be divided into symbolic and non-symbolic learners. The difference between the two groups lies in the kind of model or hypothesis they produce. Symbolic learners produce models in logic like formalisms that easily can be interpreted by humans, e.g. in the form of rules such as "if first concept=human then relation=patient". Non-symbolic learners, on the other hand, produce models that are not easily interpreted by humans. A good example of a non-symbolic learner is the instance-based learning algorithm *KNN*, where the instance is classified based on the classification of its *K* *N*earest *N*eighbors measured by some similarity measure. In this case, the relation expressed in a text excerpt would be classified as the same as its *K* most similar text excerpts. Another example of a non-symbolic learner is support vector machines (SVM), which is among the state-of-the-art approaches with respect to precision in classification. Support vector machines are a family of rather complex algorithms that produce a model by mapping the original feature space into a new space by a mapping function, θ, where the output can be linearly separated. The idea behind support vector machines is that data sets that are difficult to separate into the right classes can more easily be separated in a transformed vector space (Cristianini & Shawe-Taylor 2000).

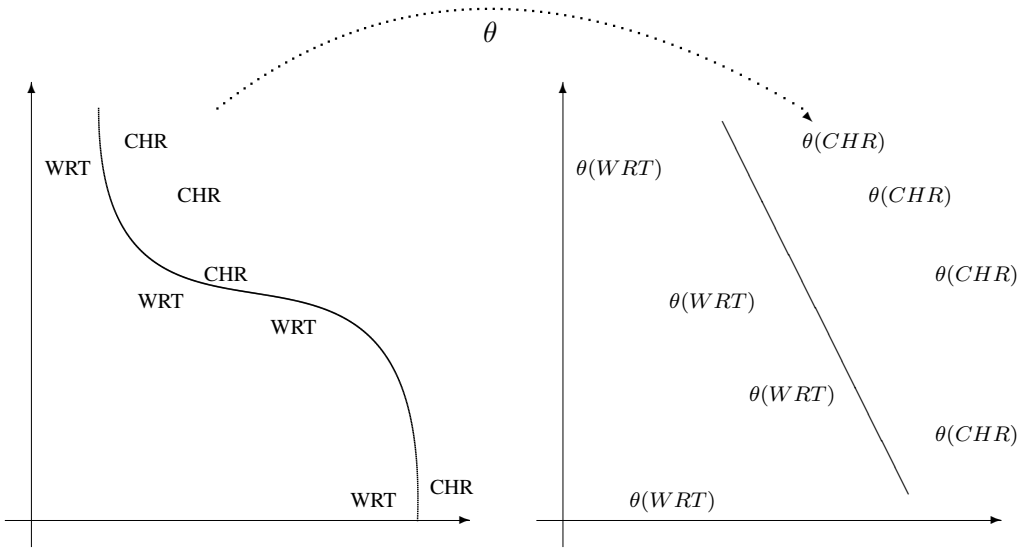


Figure 5.4: An illustration of the mapping θ of the feature space to a new space where the instances can be separated linearly.

From a performance perspective with respect to accuracy, non-symbolic learners usually produce better models than symbolic learners. However, given their non-symbolic nature, they provide less insight into the domain than symbolic learners – at least from an end user perspective.

5.3.2 Our experiments

In the example with the shopping basket in supervised machine learning, the possible content of the basket constituted the input features available to the learner. The set of features available to the learner is often termed the *feature space*. With respect to our corpus and experiments, the feature space is a six dimensional space that includes (1) the first concept, (2) first head, (3) relation, (4) preposition, (5) second concept, and (6) second head.

The function we are trying to learn is which relation a preposition denotes in a given context. To meet this end both a support vector machine algorithm (Keerthi et al. 2001) and the rule producing algorithm JRip (Cohen 1995) has been applied. Support vector machines were applied to the task in order to estimate the maximum precision one could expect from a content extraction module in an information retrieval system. JRIP was applied to the task in order to explore the affinities in the data set, and to get a rough idea of whether a simpler learner could solve the task adequately.

The experiment was performed using ten-fold cross validation (Mitchell 1997, Manning & Schütze 2003). The approach is to partition the data set in ten subsets. Out of these ten subsets, the first nine are used for training the learner and the final set is used for testing what the learner has learned. Then, the first eight and the tenth subsets are used for training, while the ninth subset is used for testing and so on. The overall accuracy of the learner is the average performance on all ten runs of the experiments. This experimental approach was chosen in order to avoid the risk of partitioning the data set into a training set and a test set with a dissimilar distribution. Moreover, all the instances in the data set can be used in the assessment of the precision of the learner.

5.4 Results

Seven experiments were run on different combinations of the feature space, ranging from using only the heads to using heads, the preposition and the concepts denoted by the heads. This was done to gain insight into the importance of using concepts in the learning. The results of these experiments are shown in table 5.4. The last column shows the precision for a projected classifier (PC) in the cases where it outperforms the trivial rejector. The projected classifier, in this case, assigns the relation that is most common for the corresponding input pair of concepts, e.g. if the concepts are *disease* or *human*, then the most common relation is PNT. The trivial rejector that assigns the most common relation to all the instances, in this case PNT, achieves a precision of 37.8%. Precision was measured as the percentage of instances where the relation were correctly classified by the learner. Statistical significance was measured using a two-tailed test at a .05 significance level.

The support vector machine algorithm produces a result which, in all cases, is

Feature Space		JRip	SVM	PC
1	Prepositions	68.4	68.5	67.6
2	Concepts	74.4	77.0	61.8
3	Lemmas	66.8	73.3	–
4	Lemmas and preposition	72.3	83.4	–
5	Concepts and lemmas	74.7	81.7	–
6	Concepts and preposition	82.6	86.6	–
7	Concepts, preposition and lemmas	84.0	88.3	–

Table 5.4: The precision of SVM, JRip and a projected classifier for the seven different combinations of input features. "Lemma" is short for lemmatized NP head.

better than the baseline, i.e. we are able to produce a model that generalizes well over the training instances compared to the projected classifier or the trivial rejector. The fact that both JRip and SVM perform better than the projected classifier when the input is only the preposition is due to statistical variation: Given just the preposition, the only reasonable model that can be built by the learner is an assignment of the most frequent relation denoted by each preposition, which is exactly what defines the projected classifier. The following trends can be identified in table 5.4 based on the performance of the SVM algorithm.

A comparison of experiments 1, 2 and 3 shows that training on concepts seems to be superior to using the lemmatized heads of the noun phrases or prepositions, but this superiority is only statistically significant when the comparison is made to the preposition and not to the lemmatized heads. Based on this reduced feature space, nothing significant is apparently gained from introducing concepts into the classification process.

When comparing experiments 4, 5, 6, and 7, the difference between the results from using the different input features is not statistically significant either. However, when comparing experiments 1, 2 and 3 to experiment 6 or 7, the improvement of using all the features compared to just one of the features is statistically significant. Also, experiment 6 shows that, in comparison to experiments 1 and 2, both the preposition and the concepts contribute significantly to the determination of the relation type.

A simple comparison between the precision scores shows that the concepts from the core of SIMPLE are the most important input feature, followed by the preposition, and, finally, the lemmatized heads of the noun phrases. The hypothesis stated in the beginning of the chapter is thus confirmed. There is an affinity between the concepts denoted by the heads of the noun phrases and the relation denoted by the preposition, as experiment 2 shows. However, the conclusion cannot be made that the concepts are of greater importance in the classification of the relation than the preposition and the lemmas. The results only indicate that this is probably the case.

In general, the results reveal an unexplored opportunity to include the concepts

and the relations that prepositions denote in information retrieval. In the next section, we examine the rules produced by JRip on the data set with only the concepts, since they are the most interesting in this context.

5.4.1 Analyzing the rules

The JRip algorithm produced, on average, 21 rules. The most general rule covering almost half of the instances is the default rule that assigns all instances to the WRT relation if no other rules apply. At the other end of the spectrum, there are ten rules that cover, all in all, no more than 34 instances, but with a precision of 100%. Analyzing these rules is difficult, since they cover the most infrequent relations and, hence, may overfit the data set. However, this does not seem to be the case with a rule like, "IF the concept of the first head is *disease* and the concept of the second head is *human* THEN the relation is *patient* covering an instance, for example, like "iron deficiency in females".

The rule with the second highest coverage, and a fairly low precision of around 66%, is the rule: "IF the concept of the second head is *body part* THEN the relation type is *locative*". The rule covers instances such as "thrombosis in the heart" but also incorrectly classifies all instances as *locative* where the relation type should be *source*. E.g. the phrase "iron absorption from the intestine", which is annotated as a *source* relation, but is classified as *locative* by the rule. However, in this case, *from* expresses a source relation very similar to a locative relation.

One of the least surprising and most precise rules is: "IF the concept of the second head is *time* THEN the relation type is *temporal*" covering an instance such as "diet for many months". We would expect a similar rule to be produced, if we had performed the learning task on a general language corpus.

With respect to speed of classification and ease of implementation, a simple set of rules like the ones produced by JRip is a viable alternative to the much more advanced model produced by a SVM algorithm. Therefore, in the implementation of an information retrieval system, a natural choice might be a set of rules even at the loss of some classification accuracy. Shown below is an example rule set produced by JRip based on the corpus:

1. IF second head is *quality* THEN the relation is CAUSED BY
2. IF second head is *state* and first head is *natural substance* THEN the relation is TEMPORAL ASPECTS
3. IF second head is *event* THEN the relation is TEMPORAL ASPECTS
4. IF second head is *amount* and first head is *food* THEN the relation is CHARACTERIZED BY
5. IF first head is *institution* THEN the relation is CHARACTERIZED BY

6. IF first head is *body depot* and second head is *natural substance* THEN the relation is COMPRISING
7. IF second head is *state* and first head is *change* THEN the relation is TEMPORAL ASPECTS
8. IF first head is *agent of temporary activity* THEN the relation is CHARACTERISTIC
9. IF first head is *disease* and second head is *change* THEN the relation is CHARACTERIZED BY
10. IF first head is *cause change of state* THEN the relation is PATIENT
11. IF first head is *natural substance* and second head is *microorganism* THEN the relation is LOCATIVE
12. IF first head is *disease* and second head is *human* THEN the relation is PATIENT
13. IF second head is *artifact* and first head is *act* THEN the relation is BY MEANS OF
14. IF second head is *time* THEN the relation is TEMPORAL ASPECTS
15. IF first head is *cause change* THEN the relation is PATIENT
16. IF first head is *human* THEN the relation is CHARACTERIZED BY
17. IF second head is *unit of measurement* THEN the relation is CHARACTERIZED BY
18. IF first head is *creation* THEN the relation is PATIENT
19. IF first head is *change* THEN the relation is PATIENT
20. IF first head is *act* THEN the relation is PATIENT
21. IF second head is *body part* THEN the relation is LOCATIVE
22. In all other cases the relation is WITH RESPECT TO

5.5 Related Work

As noted at the beginning of this chapter, with no previous studies of this kind ever having been performed in Danish before, conducting the experiments in Danish was a priority. However, a large amount of research on the behavior and semantics of prepositions exists and ACL-SIGSEM has had frequent workshops on the topic (Toulouse 2003, Colchester 2005, Trento 2006, and Prague 2007). SemEval 2006 also had a task on the word sense disambiguation of prepositions based on the Framenet corpus. The task originated within The Preposition Project, whose goal is to provide a comprehensive characterization of English preposition senses suitable for use in natural

language processing (TPP 2007). There is a difference, though, in word sense disambiguation and ontological relation disambiguation in that the former has a lexical starting point rather than an ontological one. Compared to what has been presented here, the general focus has been on deep linguistic studies of individual prepositions or, e.g. locative or spatial prepositions (Litkowski 2004), rather than learning to identify the semantic relation denoted by prepositions in general through corpus annotation. To the best of our knowledge there has been no previous research to Lassen & Terney (2006*a*,*b*) on what kind of semantic relations there exists between concepts based on general ontological knowledge about the participating concepts.

5.6 Discussion and Summary

The experiments presented in this chapter confirm the hypothesis that there is an affinity between the concepts denoted by the heads of two noun phrases and the relation denoted by the preposition between these two noun phrases. It is shown that general ontological knowledge about the concepts is sufficient to achieve high precision in categorizing the type of relation. Also by applying a relatively small set of rules we are able to gain a high precision. Thus a more detailed and accurate semantic analysis of both documents and queries can be achieved by this relatively simple mean which again results in more specific descriptors.

The manual relation annotation has been done by one annotator. The ideal situation would naturally be to have several annotators annotate the corpus. If two or more people annotate the same corpus, they are almost certain to disagree on some occasions. This disagreement can have two sources. First, it can be due to cognitive differences. Two people exposed to the same utterance are not guaranteed to perceive the same content, or to perceive the content intended by the producer of the utterance. Many factors are at play here: Cultural background, knowledge, memory, etc. Multiple annotators, of course, would have been better. In fact, the experiments can only tell us if learning the relations as interpreted by a single annotator is possible. Since no experiments with multiple annotators have been performed, the conclusion cannot be extended to a more general statement, but a multiple annotator scenario is unlikely to create a result very different from the one presented here, given that the relations are generic relations with relatively little overlap (except perhaps the locative and partitive relation as described in Lassen (2007)).

The data set is too small to conclude that the concepts are of greater importance than the preposition and the lemmatized heads, although this is what the experiments certainly indicate. Currently, a larger general language corpus in the form of excerpts from the Bergenholtz corpus is in the process of being compiled. This work is in an early phase, but the intent is to be able to refine the conclusions already drawn here, and to test their validity in general language.

The fact that a relatively small rule set covers many of the instances and provides a precision in classification comparable to that of a state-of-the-art approach like

support vector machines is encouraging. There is, in other words, a tradeoff between simplicity and precision in classification. That the best solution is high precision and slow classification in an information retrieval system is not at all a given.

There are several aspects of these studies that warrant future work. From a linguistic and ontological perspective, it is interesting that "with respect to" (WRT) is used widely in the annotation. Most likely due to the broadness of WRT, an examination of the need for expanding the set of relations or possibly replacing some relations by more finely grained sub types is called for. In an exploratory study like this, it might be relevant to perform different kinds of clustering on the set of instances labeled with the WRT relation in order to find possible natural groupings of the instances.

Another interesting future task is to base the learning process on a semantic similarity measure, i.e. a similarity measure derived from the ontology. Since approximately 120 of the instances have both nouns mapped into a concept beneath the level of the core of SIMPLE, it could be interesting to perform an experiment with a semantic similarity measure on this subset of the data or with a larger set of data with more nouns being mapped at a more specific level than at the core level of SIMPLE. The aim would be to examine if semantic similarity measures are better at capturing the similarity between the instances than the similarity measures based on the vector space representation used in the experiments presented here. Naturally, one would assume a much more accurate classification when using a semantic similarity measure.

Chapter 6

Combining semantic and distributional similarity

Is the concept of a dog more similar to the concept of a cat than to the concept of a mammal? And are dogs and cats more similar than, say, bikes and automobiles? And what does similarity actually imply? This chapter explores two different notions of similarity, one based on ontologies and one based on corpus statistics, to answer these questions. In addition, this chapter presents three different approaches for combining these two notions of similarity.

We restrict the use of *semantic similarity measures* to denote similarity measures based on some sort of world knowledge typically expressed in the form of an ontology. Thus, *dog* is not just a three letter word comprised of *d-o-g*. It is our conception of a *dog* as a special canine somewhat different from, say, a wolf. The intuition behind semantic similarity measures is that it is possible to derive a measure of similarity which corresponds to a human perception of semantic similarity. *Distributional similarity measure*s, on the other hand, are based solely on corpus statistics and similarity is used to denote the appropriateness of substituting lexical unit a with lexical unit b; thus, *dog* is only a three-letter word made up of *d-o-g*. Distributional similarity measures are intended to measure the appropriateness of substituting *dog* with, e.g. *pet*. However, when we later attempt to combine semantic and distributional similarity measures, we use the term distributional similarity in a more general way, namely to denote the appropriateness of substituting object a with object b, thereby enabling the distributional similarity of concepts rather than mere terms.

The question of how to model similarity or relatedness is central to many natural language processing applications, for example, word sense disambiguation, resolving prepositional phrase attachment ambiguities, etc. (for an overview see, e.g. Weeds (2003)). Since ontologies and taxonomies can be viewed as structural encodings of the semantic similarity between different concepts, various methods have been proposed for using these structures as a basis for measuring semantic similarity. An early method proposed for measuring similarity is simply to use the shortest path be-

tween two concepts as a measure of their semantic similarity (Rada et al. 1989). A problem with this simple approach, however, is that it, "relies on the notion that links in taxonomy represent uniform distances" (Resnik 1999). For instance, in WordNet, *horse* is the immediate subsumer of *pony*, and *biology* is the immediate subsumer of *zoology*. Most would agree, however, that *horse* and *pony* are more closely related than *biology* and *zoology*. Several semantic similarity measures have later been proposed that modify the link distance; some of them are corpus based, and some of them modify the link distance based on the taxonomic structure itself, e.g. the density and depth of concepts in the hierarchy. We will return to the different semantic similarity measures in the next section.

Distributional similarity measures have been suggested as an approximation of semantic similarity measures (Dagan et al. 1999, Weeds & Weir 2005, Mohammad & Hirst 2005) and are based on the co-occurrence of words or concepts with no knowledge of their ontological relatedness. The underlying assumption behind the distributional similarity measure is the distributional hypothesis, which Harris (1968)) describes as, "The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities". Or to put in more simple terms; words that occur in the same context tends to have related meaning.

The previously suggested methods for semantic similarity that incorporate corpus statistics apply information theoretic measures that only use concept or word frequency. Intuitively, as the distributional hypothesis states, we would consider co-occurring concepts as being more related than concepts that are rarely used in the same context. As a result, taking advantage of the knowledge embedded in the ontology while taking into account the distributional patterns of concepts as they appear in a particular corpus seems like an obvious step. This opens up for a tailoring of the similarity measure to a specific corpus for use in, e.g. information retrieval, for instance, to achieve a co-occurrence-based modification of the link distance central to many semantic similarity measures. This chapter introduces three general approaches that combine semantic similarity measures with distributional similarity measures. The presentation of the contribution in this chapter is a further development of the ideas presented in Terney (2007).

Section 6.1 contains a short introduction to some of the semantic similarity measures, while section 6.2 introduces distributional similarity and the importance of context definition. The main concern of both of these sections is to provide an intuitive understanding of the main principles of the different measures using visualization and examples. Section 6.3 follows with the presentation of three general approaches for combining semantic similarity measures with distributional similarity measures. Finally, section 6.4 ends the chapter with a discussion and a summary.

6.1 Semantic Similarity

Resnik (1995) argues that semantic similarity is a special case of the more general notion of semantic relatedness. He exemplifies his claim with *cars*, *bikes* and *gasoline*. *Cars* and *gasoline*, he argues, are normally considered to be more related than *cars* and *bikes*. On the other hand, *cars* and*bikes* are certainly perceived to be much more similar than *cars* and *gasoline*. In ontological terms, this perspective narrows semantic similarity measures down to including only the hypernomic links between the concepts. Relatedness is a broader notion that can be any kind of links between the concepts in the ontology, e.g. meronymy or causality. The use of distance, Resnik notes, is more ambiguous than similarity and relatedness, since it is used as the inverse of both. Resnik's distinction between similarity and relatedness is adopted here.

In their evaluation and overview of various semantic similarity measures, Budanitsky & Hirst (2006) present the following two possible groupings: 1) edge-based methods, and 2) information theoretic measures and combined measures. The first group bases its similarity model solely on the structure of the taxonomy, while the second group uses information theoretic measures derived from corpus statistics.

6.1.1 Edge-based methods

An early method proposed for measuring semantic similarity is simply to use the shortest path between two concepts. Using this method, Rada et al. (1989) show good results for their information retrieval task on *Medline* using *Mesh* as an ontology[1]. With respect to atomic concepts, semantic similarity was calculated here as the minimum number of edges connecting two concepts in the ontology. With respect to sets of concepts, Rada et al. (1989) measure semantic similarity as the average distance between all pairwise combinations of concepts.

In his experiments on word sense disambiguation on the *Time* corpus using WordNet, Sussna (1993) notes that there often is a varying semantic similarity between superclasses and subclasses in different parts of the ontology. Because concepts appearing in the lower parts of the ontology are usually more closely related than concepts appearing in the upper parts, Sussna devises an edge-based weighting scheme as an extension to the shortest path approach that takes into account the depth of the concepts in the tree. This weight is termed *relative depth scaling*. Concepts that are positioned deep in the ontology get a higher similarity rating than concepts positioned close to the top. The principle of relative depth scaling is also implemented in a different form in the similarity measures by Leacock & Chodorow (1998) and Wu & Palmer (1994). In addition to the relative depth scaling, Sussna also adds a *type specific fanout* factor that adjusts the weight based on the number of children of a given concept, which can be thought of as the density of the immediate area of the graph covered by the con-

[1]Note that the ordering relation in Mesh is BROADER-THAN, which besides hypernomy also includes meronomy.

cept. The intuition is that a high number of children indicates a broad concept, thus the similarity between subsumer and subsumed should be low compared to concepts in the ontology with fewer children. The principle of a type specific fanout factor is illustrated in figure 6.1. Overall, Sussna's approach is based on the shortest path with a modification of the edge distance via the relative depth scaling and the type specific fanout factor.

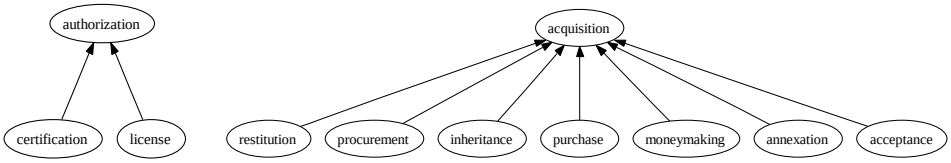


Figure 6.1: An illustration of Sussna's fanout factor principle. Given that the two ontology excerpts are positioned at the same level of depth in the ontology (which is the case in WordNet), *certification* and *authorization* are given a higher similarity than the similarity between *restitution* and *acquisition*.

6.1.2 Information theoretic and combined measures

Resnik (1995, 1999) suggests a different solution to the problem of measuring semantic similarity than Sussna (1993), though his approach is based on the same observation that, "A widely acknowledged problem with this approach [the shortest path], however, is that it relies on the notion that links in the taxonomy represents uniform distance". Using information content, Resnik devises a new similarity measure and applies it to the problems of syntactic and semantic ambiguity. With these tasks, it clearly outperforms the shortest path approach. The measure is based on the information content of concept c given by $-\log(p(c_1))$, where $p(c_1)$ is the probability of encountering c. The probability estimates used by Resnik are undisambiguated word frequencies, i.e. every occurrence of "bank" counts towards the same total regardless of its sense being either river bank or financial institution. If a is the subsumer of b, the information content of a will always be equal to or less than the information content of b, since every occurrence of b counts as an occurrence of a. Therefore, the information content is monotonically decreasing when moving upwards in the hierarchy, which intuitively makes sense since the concepts become more and more abstract. The similarity between two concepts is defined as the information content of the least upper bound with the highest information content. This is illustrated in figure 6.2 with the similarity between *trout* and *salmon*, which share two least upper bounds. Since *salmonid* has the highest information content of the two least upper bounds, $sim(trout, salmon) = 17.8$.

Jiang & Conrath (1997) suggest a combined method based on the proposed methods presented earlier. The offset is the shortest path approach, but each edge in the ontology is scaled by a parameterized factorization of density of the graph (i.e. a

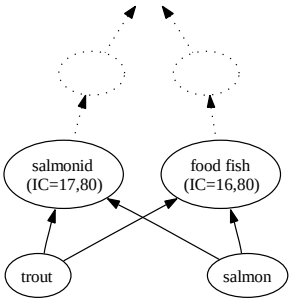


Figure 6.2: An illustration of Resnik's similarity measure showing that $sim(trout, salmon) = 17.8$ given that *salmonid* is their least upper bound with the highest information content.

modified fanout factor), relative depth scaling, a weighting of the relation type (all three factors, similar to Sussna's proposal), and, finally, difference in information content of a subsumer and a subsumed (an idea similar to Resnik's). In empirical studies, this approach has shown good, if not the best, results when compared to human evaluation of semantic similarity and the detection and correction of real-word spelling errors in open-class words, i.e., malapropisms.(Budanitsky & Hirst 2006).

For a more detailed and formal description of the measures presented here and of other measures of semantic similarity, see, e.g. Budanitsky & Hirst (2006) or Andreasen & Bulskov (2007*b*).

6.2 Distributional Similarity

Distributional similarity measures, which have been suggested as an alternative to semantic similarity measures, have been applied to some of the same tasks (Dagan et al. 1999, Weeds & Weir 2005). Formally, the distributional similarity of two events, e_1 and e_2, is their tendency to co-occur in the same context. Several different measurements have been proposed to measure this tendency, e.g. the cosine, pointwise mutual information, the Kullback-Leibler divergence and Jaccard's coefficient, etc. The principles of these measures will be presented shortly, however, the definition of context profoundly influences the distributional similarities found. A presentation of the definition of context should therefore always precede the exact measures chosen.

6.2.1 Context

Next, we analyze how the different possible definitions of context influence the similarities found. Context in natural language processing can be defined along the following three dimensions:

1. Syntactic
2. Affinity
3. Entity

The first dimension is whether the context is derived syntactically or whether it is just a window frame. Syntactically derived context can be, for instance, the surrounding noun phrase, verb phrase, sentence, or perhaps even special syntactic functions like verb-object relations. A window frame around the word can be, e.g. $\pm$ a 5 or 10-word window or even the entire document.

The second dimension is the order of affinity influences. Grefenstette (1994*a*) distinguishes between different orders of affinity. A first order affinity is a word often occurring in immediate vicinity of another word. Examples of affinities found this way are topical affinities like the affinity between pairs such as *doctor*...*nurse* and *save*...*from* found by Church & Hanks (1990) using a non-grammatical context.

Second order affinity is words that share the same environments, i.e. a second order affinity is not between words in the immediate vicinity of each other but words that share first order affinities. An example of a second order affinity is between the words *tumor* and *tumour*, where the two different spelling variations are unlikely to occur in the immediate vicinity of one another; however, they will very likely share the same kind of context. Affinities found in this way are often between words from the same syntactic classes and general semantic classes, similar to the affinities found by Hindle (1990). Hindle, who examines the affinity between nouns based on their distribution with respect to subject-verb-object relations in a corpus of news stories, finds, e.g. that the most similar noun to *boat* is *ship*. When he lists a set of "reciprocally most similar" nouns, i.e. pairs of nouns that are each other's most similar noun, many of them are near synonyms.

The third dimension in defining context is the issue of what kind of entities is analyzed: Word senses, lexical units or perhaps characters. Using a context of characters has proven to be successful, for instance, in spelling error correction. Third order affinity is defined by Grefenstette (1994*a*) as words *senses* sharing the same context. Though it is interesting that Grefenstette includes the notion of senses and thereby concepts in his work on distributional similarity, the question of senses is in principle unrelated to the other two orders of affinity or to the question of syntactically derived context.

Clearly, the distributional similarity is able to capture some kind of relatedness between lexical units. However, their relatedness is influenced by the order of affinity between the two entities and whether the context is syntactical or not. The first order affinity word pairs found by, e.g. Church & Hanks (1990) using a five-word window, were of a topical nature. The word pairs found with second order affinity and related by a specific syntactic relation found by, e.g. Hindle (1990), were near synonyms.

In the following, the principles behind four different kinds of distributional similarity measures are presented. As with the semantic similarity measures, the focus is

1.	bomb	–	device	16.	peace	–	stability
2.	ruling	–	decision	17.	property	–	land
3.	street	–	road	18.	star	–	editor
4.	protest	–	strike	19.	trend	–	pattern
5.	list	–	field	20.	quake	–	earthquake
6.	debt	–	deficit	21.	economist	–	analyst
7.	guerrilla	–	rebel	22.	remark	–	comment
8.	fear	–	concern	23.	data	–	information
9.	higher	–	lower	24.	explosion	–	blast
10.	freedom	–	right	25.	tie	–	relation
11.	battle	–	fight	26.	protester	–	demonstrator
12.	jet	–	plane	27.	college	–	school
13.	shot	–	bullet	28.	radio	–	IRNA
14.	truck	–	car	29.	2	–	3
15.	researcher	–	scientist				

Table 6.1: Reciprocally similar nouns from Associated Press news stories (Hindle 1990).

on an intuitive understanding of the measures rather than their formal aspects. See, e.g. Cover & Thomas (1991), Dagan (2000), Weeds (2003), Terra & Clarke (2003), and Mohammad & Hirst (2005) concerning the formal aspects.

6.2.2 Set theoretic measures

A common trait of measures like *Dice, Jaccard, Tanimoto* (Dagan 2000, Manning & Schütze 2003) is that they measure similarity, or inversely, distance, as the proportion of features or elements shared by the two entities compared to all the features or elements characterizing both the entities. Let the set A, depicted in the Venn diagram in figure 6.3, be all the elements in the context of concept a, and B be the set of elements appearing in the context of b. Elements can be neighboring concepts, words or whatever the definition of context is.

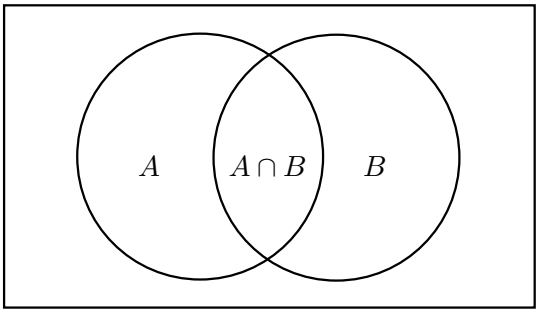


Figure 6.3: An illustration of the sets of context elements A and B included in the majority of set theoretic similarity measures.

The Jaccard measure of similarity is defined as:

$$sim_{Jaccard}(a,b) = \frac{|A \cap B|}{|A \cup B|} \tag{6.1}$$

As a refinement to the Jaccard measure, Grefenstette (1994*b*) introduces a weighted version where each element in A and B are weighted according to the degree of their affinity with a and b, respectively:

$$sim_{wJaccard}(a,b) = \frac{\sum_{x \in A \cap B}(\min(weight(a,x), weight(b,x))}{\sum_{x \in A \cup B}(\max(weight(a,x), weight(b,x))} \tag{6.2}$$

The weighted Jaccard measure is in effect, thus, the fuzzy generalization of the crisp Jaccard measure with union and intersection being the standard fuzzy intersection (the minimum affinity) and union (the maximum affinity).

6.2.3 Geometrical measures

The different geometrical distributional similarity measures build on a vector space model representation of context. Given this vector representation of entities to compare, similarity can be measured by, e.g. the *cosine* as presented in chapter 2 or by the *Minkowski* distance (Weeds & Weir 2005). If, for instance, $sim(c,e)$ is a weight denoting the affinity between concept c and the element e, then the Minkowski distance between concept a and b is defined as:

$$dist_m(a,b) = \sqrt[m]{\sum_{e \in E} |sim(a,e) - sim(b,e)|^m} \tag{6.3}$$

where E is the set of possible elements that can appear in the context of a and b. If $m = 2$, this gives the familiar *Euclidian* distance, and if $m = 1$, it gives the L_1 norm (Weeds & Weir 2005, Manning & Schütze 2003, Dagan 2000). The L_1 norm is also called the *Manhattan* distance or *city block* distance because it measures the distance between two points if you are only able to travel in orthogonal directions. The geometrical interpretation of similarity in a two dimensional space is illustrated in figure 6.4. The second order affinity between *car* and *bus* can be measured by representing each concept in the vector space with the dimension *ride* and *drive*.
The L_1 norm would in this case, thus, be calculated as:

$$\begin{aligned} dist_{m=1}(car, bus) &= |sim(car, drive) - sim(bus, drive)| + \\ &\quad |sim(car, ride) - sim(car, ride)| \\ &= 0.55 \end{aligned}$$

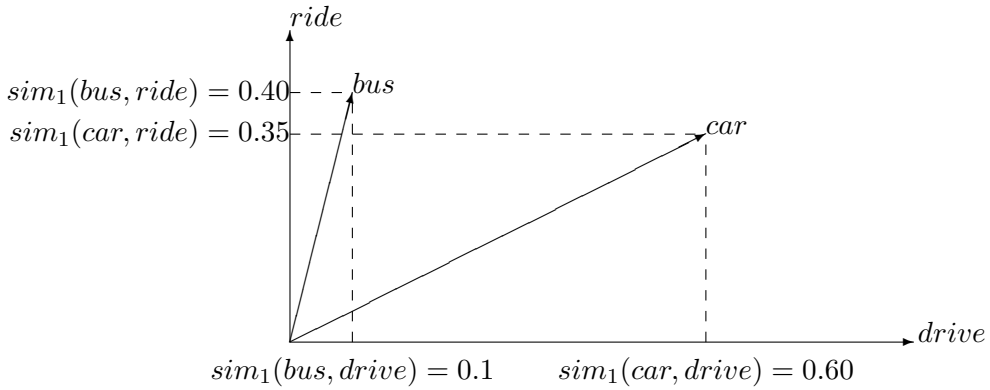


Figure 6.4: An illustration of the geometrical interpretation of similarity leading to distributional similarity measures like, e.g. the cosine.

6.2.4 Information theoretic measures

In contrast to the previously presented distributional measures, the probabilistic *Kullback-Leibler* divergence is an asymmetrical measure of similarity (Cover & Thomas 1991, Manning & Schütze 2003, Dagan 2000). The Kullback-Leibler divergence measures the similarity between two probability distributions as given by the formula[2]:

$$D(p||q) = \sum_{e \in E} p(e) \times \log \frac{p(e)}{q(e)} \tag{6.4}$$

where p and q are the two different probability distributions over some discrete random variable, E. The Kullback-Leibler divergence measures the total divergence of the probability distribution, p, to the probability distribution of q.

For instance, p and q can be the probability distributions of the set of verbs that the concepts *car* and *bus* are the direct objects of. If the cardinality of the set of verbs is 15, then the probability distributions can be depicted as illustrated in figure 6.5. The Kullback-Leibler divergence measures the total similarity of $p(v)$, the white bars, to $q(v)$, the shaded bars, i.e. how much the white bars differ from the shaded bars.

The mathematic convention $p(v) \times \log(p(v)/0) = \infty$ and $0 \times \log(0/q(v)) = 0$ (Cover & Thomas 1991) is required otherwise the Kullback-Leibler divergence is undefined when either p or q have a probability estimate of 0. Probability estimates of 0 can be avoided by applying some form of smoothing prior to applying the Kullback-Leibler divergence, i.e. adjusting the likelihood estimates of low frequency words and especially assigning a non-zero probability to unseen words. The rationale for

[2]Please note, that following Cover & Thomas (1991), Manning & Schütze (2003) we let $p(x)$ denote the probability mass function $p_X(x)$ for convenience.

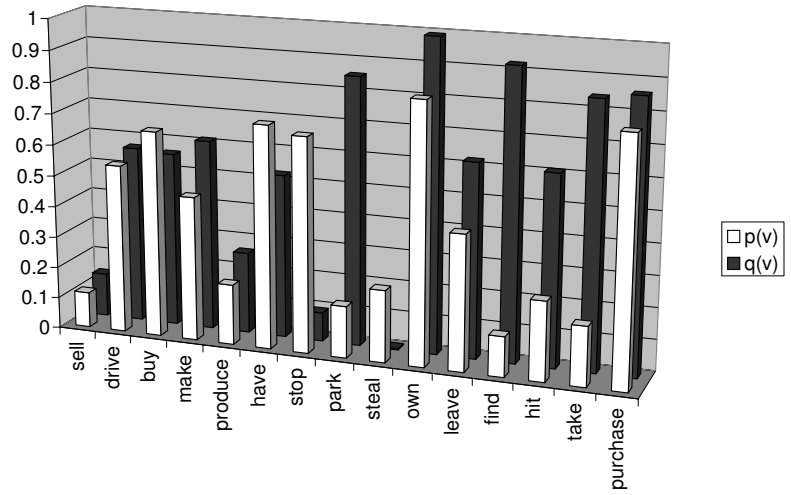


Figure 6.5: An illustration of the probability distributions of the concepts *car* (p) and *bus* (q) over the set of verbs, (v), of which they are the direct object. The Kullback-Leibler divergence measures the total similarity of $p(v)$, the white bars, to $q(v)$, the shaded bars. Note that this is just an illustration, and that the numbers are generated randomly.

applying smoothing is that because *drive* and *car* are never seen in context in our corpus assuming a zero probability of them ever appearing is unreasonable (for more on smoothing see e.g. (Manning & Schütze 2003)).

The *Jensen-Shannon* divergence (Dagan et al. 1999) measures the total divergence to the average of the two distributions, p and q, as given by the formula:

$$D_{js}(p,q) = \frac{1}{2}\left[D\left(p||\frac{p+q}{2}\right) + D\left(q||\frac{p+q}{2}\right)\right] \tag{6.5}$$

Thus the Jensen-Shannon divergence eliminates the problem of smoothing since neither of the probability estimates appears in the denominator. The Jensen-Shannon divergence also differs from the Kullback-Leibler divergence by being symmetric in that it measures the divergence to the average of the two distributions.

Pointwise mutual information is another probability-based measure that has been used by several researchers as a basis for measuring the strength of affinity in natural language processing, e.g. the experiments by Church & Hanks (1990) and Hindle (1990) referred to in the section on context definition. *Mutual information* is the reduction in uncertainty of one random variable due to knowing the value of the other variable. This can be defined as (Cover & Thomas 1991):

$$I(C;E) = \sum_{c,e} p(c,e) \times \log \frac{p(c,e)}{p(c) \times p(e)} \tag{6.6}$$

where E is the set of elements, C is the set of concepts and $p(c,e)$ is their joint

probability distribution. For instance, if the elements are verbs in a direct object-verb construction with the concept c, then *climb*, to a larger extent, reduces the uncertainty of what concept c is than if the verb had been *have*, which is natural because a good deal fewer concepts appear as the direct object of *climb* than as the direct object of *have*.

The mutual information of two variables can be thought of as a measure of their independence, which in the case of the direct object-verb relation is how independent the verb is of the direct object and vice versa. Pointwise mutual information is the independence between two specific variables, c and e:

$$I(c,e) = \log \frac{p(c,e)}{p(e) \times p(c)} = \log \frac{p(c|e)}{p(c)} = \log \frac{p(e|c)}{p(e)} \tag{6.7}$$

Based on pointwise mutual information, Hindle (1990) defines a measure of similarity between two nouns as:

$$sim_{Hindle}(n_1, n_2) = \sum_v f(I(v, n_1), I(v, n_2)) \tag{6.8}$$

where:

$$f(a,b) = \begin{cases} \min(a,b) & if\ a > 0, b > 0 \\ \text{abs}(a,b) & if\ a < 0, b < 0 \\ 0 & otherwise \end{cases} \tag{6.9}$$

One of the problems with using pointwise mutual information is that it tends to overemphasize low frequency co-locations, which is counter intuitive to what we expect of a good similarity measure (Manning & Schütze 2003).

6.3 Combining Semantic and Distributional Measures of Similarity

We now turn to approaches for combining semantic and distributional similarity measures. As described in the introduction, the purpose of combining semantic similarity and distributional similarity is to arrive at a similarity measure that utilizes the knowledge embedded in the taxonomy while taking into account the distributional similarities between concepts as found in a particular corpus.

The motivation for proposing this combined measure of similarity is closely connected to the discipline of ontological engineering. If the semantic similarity measure used to search in a corpus is not application or corpus specific, it is likely that not all relevant relations are modeled in the ontology. This is partly because the ontological coverage of the conceptual structure of a domain can never be guaranteed to be complete, but more importantly, also because there are relevant relations from an information retrieval viewpoint that are not reasonable to model. For instance, is *Iraq* closer to *war* than, e.g. *Sweden*? From an ontological viewpoint this is not the case;

but all things being equal, *Iraq* is currently more closely related to the concept of *war* than *Sweden*. In a corpus on historical events, associating *war* more closely to especially war faring nations compared to largely peaceful nations, as measured by how often they appear in the context of *war*, would make sense from an information retrieval perspective. Another example is the different associations *oil* would have in a corpus on foreign policy compared to a corpus on environmental issues. In the former, *oil* probably has strong associative links to *OPEC*, whereas in the latter, *oil* probably has strong associative links to concepts like *beach* and *tanker*. In summary, the argument put forward here is that from an information retrieval perspective on similarity measurement, dynamically establishing associative links between concepts in the ontology is relevant, i.e. links where grounding them more formally in the ontology is not reasonable.

Another perspective on combining semantic and distributional similarity is that some links are more important than others, and that this importance, to a certain degree, is corpus dependent. Rather than establishing dynamic links, the intent is thus to adjust the similarity based on the strength of the associativity along existing relational links between the concepts. For instance, in the domain of agriculture, *pesticide* and its generalization *chemical* is probably more closely related both conceptually and distributionally than in a corpus on toys where *chemical* is probably more related to its generalization *softener*.

From the perspective of distributional similarity, there is also a reason for trying to combine with more knowledge-based methods of measuring similarity. In his overview of different distributional similarity measures, Dagan (2000) argues that there is still ample room for future research in distributional similarity measures since many of the patterns found can be regarded as noise. The addition of ontological knowledge can guide the process of finding relevant distributional patterns. If a concept, a, has a high distributional similarity with several concepts situated closely to concept b in the ontology, these patterns should not be regarded as noise but rather as expressing special affinities in the corpus. Dagan, for instance, finds that *shipment* has a high distributional similarity with both *sale* and *contract* which on a conceptual level are a type of *agreement*. That *shipment* also has a high distributional similarity with *election* is probably less relevant if it is the only link between the two areas in the ontology.

The proposed methodology here is related to *ontology learning* based on the statistical analysis of corpora as described in, e.g. Maedche & Staab (2004). In ontology learning, the aim is to find permanent and, to a large extent, verified ontological relations that can be used for various information system tasks. However, the main concern focuses on measuring similarity between concepts for use in information retrieval based on the possible dynamic and temporary associativity of concepts, i.e. associativity as expressed through their distributional similarity and their conceptual similarity.

In the following, we suggest three possible approaches for combining semantic

similarity and distributional similarity. First, a direct approach where the semantic similarity of two concepts, a and b, and the distributional similarity are directly combined. Second, an approach where the semantic similarity is combined with what we will refer to as the "distributional density" of all the concepts on the path leading from a to b. Third, an integrated approach where the link strength between parent and child concepts is modified by their distributional similarity.

6.3.1 A direct approach

The similarity of two concepts, a and b, can be measured with both a semantic similarity measure, sim_s, and a distributional similarity measure, sim_d. A straightforward approach is a weighted combination of the two similarity measures, for instance, as a weighted average or possibly the product:

$$sim_{direct}(a, b) = sim_s(a, b)^{\alpha} \cdot sim_d(a, b)^{\beta} \tag{6.10}$$

When the parameter $\alpha = 1$ and $\beta = 0$ sim_{direct} is the semantic similarity, and when the parameter $\alpha = 0$ and $\beta = 1$ sim_{direct} is the distributional similarity. This straightforward kind of combination is illustrated in the taxonomy fragment in figure 6.6, where the taxonomic path leading from a to b, represented by a solid line, is used for calculating the semantic similarity which, in turn, is modified by the distributional similarity represented by the dotted line. The advantage of the direct combination is its simplicity and that any kind of semantic and distributional similarity measures can be used.

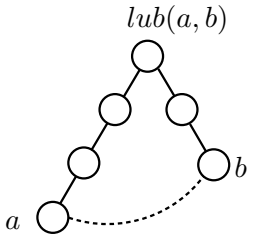


Figure 6.6: An illustration of the connections taken into account in the direct approach. The concept pair for which distributional similarity is measured is shown with a dotted line and taxonomic links are shown with solid lines. $lub(a, b)$ denotes the least upper bound of a and b.

6.3.2 A density-based approach

Even though the direct approach is simple and therefore appealing, we do not fully benefit from the taxonomy when calculating the distributional similarity. High dis-

tributional similarity of any pair of concepts along the path from a to b should have an effect on the combined similarity measure. For instance, in a corpus on environmental issues, a high similarity between $chemical$ and $pollution$ should affect the similarity of *pollution* to any specific kind of *chemical*, e.g. $pesticide$ or $herbicide$.

To capture this aspect, we introduce the notion of distributional density. Distributional density is concerned with the general notion of an aggregated distributional similarity derived from concepts in the structural vicinity of a and b in the ontology. Let V_a and V_b be the concepts in the vicinity of a and b, respectively. The distributional density can then be expressed as:

$$d(a,b) = \frac{1}{|V_a \times V_b|} \sum_{(x,y) \in V_a \times V_b} sim_d(x,y) \tag{6.11}$$

With V_i, for instance, given by:

$$V_i = \{c | len(c_i, c) < x\} \tag{6.12}$$

with $len(c_i, c)$ being the shortest path length from c_i to c. The notion of distributional density is illustrated in the taxonomy fragment in figure 6.7.

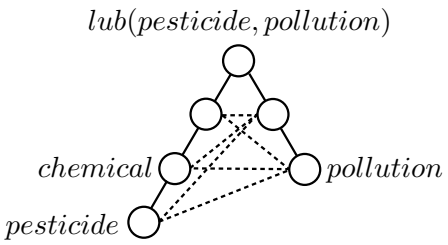


Figure 6.7: An illustration of the connections taken into account in the distributional density approach. Concepts are considered relevant if they are at a distance of 1 from $chemical$ or $pollution$.

Similar to the direct measure, the combined similarity can be calculated as a product of the semantic similarity and distributional density:

$$sim_{density}(a,b) = sim_s(a,b)^{\alpha} \cdot d(a,b)^{\beta}$$

The direct method introduced earlier is in fact a special case of the density based method with V_i as in equation 6.12 where $x = 0$. Similar to the direct method distributional density approach, this method can be a combination of any kind of semantic and distributional similarity measure.

6.3.3 A weighted link approach

While the two other suggested approaches modify the result of the semantic similarity measure, the integrated approach introduced in the following is based on a modification of the basis for calculating semantic similarity. The purpose is to reduce the distance between child and parent concepts if there is a minor difference in their use in the corpus. The inspiration for this approach is taken from the combined approach proposed by Jiang & Conrath (1997) described in 6.1.2. Jiang & Conrath adjust the link strength between a child and a parent node based on, among other things, their difference in information content. A small difference indicates high similarity and a large difference indicate low similarity.

Figure 6.8 illustrates a scenario where *agreement* is the subsumer of *settlement* and *conspiracy*, and where the number of occurrences in the corpus is indicated in parentheses. The concept $agreement$ appears a total of 200 times, 150 of which are

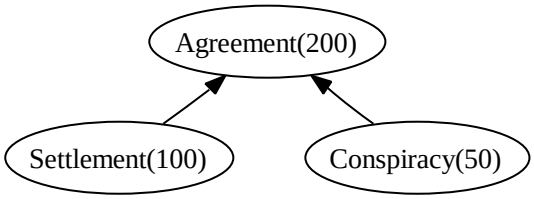


Figure 6.8: An illustration of the number of occurrences of the three concepts *agreement settlement*, and *conspiracy* and their relation. Note that *agreement* occurs 200 times because each time*settlement* and *conspiracy* appear, they count as an occurrence of *agreement*.

because each occurrence of $conspiracy$ and $settlement$ counts as an occurrence of $agreement$ and 50 of which are when *agreement* occurs alone. Let us assume that the 50 times $agreement$ occurs, it actually co-occurs with $conspiracy$. Using information content to adjust the edge distance would result in $agreement$ and $settlement$ having a high degree of similarity compared to $agreement$ and $conspiracy$, despite $agreement$ and $conspiracy$ having a distributional similarity that is much higher ($sim_d(agreement, settlement) << sim_d(agreement, conspiracy)$). In other words, the use of information content can lead to a counter-intuitive weighting of the link distance between two concepts.

An alternative to the information-based link adjustment can be achieved by weighting the link strength between a parent concept and a child concept based on their co-occurrence. Like the information-based weight, a co-occurrence-based weight can be combined with any of the previously suggested edge-based weighting schemes or all of them (as Jiang & Conrath in essence also propose). The result would thus be a weight that incorporates diverse information about density, depth, information content, relation type and, finally, distributional similarity. This method is illustrated in the taxonomy fragment in figure 6.9. As noted, only edge-based semantic similarity

measures are suited for this combination with distributional similarity measures.

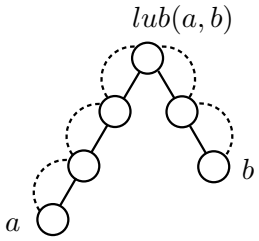


Figure 6.9: An illustration of the integrated method.

If sw_p denotes the sum of the total weights on the links on path p from concept a to b, and P is the set of all paths, then the similarity between the two concepts can be expressed by applying the shortest path principle as:

$$sim_{integrated}(a, b) = \min_{p \in P} [sw_p]$$

6.4 Discussion and Summary

The purpose of the research described in this chapter has been to explore approaches for combining semantic similarity measures based on ontologies with distributional similarity measures based on corpus statistics. We suggest three approaches that combine these two types of similarity measures in a common measure of similarity. The first approach combines the result of any semantic similarity measure with the distributional similarity of the concepts and the path between the two concepts. The second approach combines the result of any semantic similarity measure with the distributional density of the path between the two concepts, and, as such, is a generalization of the first approach. The third approach modifies the basis of an edge-based semantic similarity measure by including knowledge about the co-occurrence of parent-child concepts in a corpus. This method can be seen as a supplement to existing edge-based approaches, thereby offering another option for adjusting the link distance used for similarity measurement in ontologies. Though distributional similarity and semantic similarity are related it is a novel idea to combine the two measures in one.

This chapter introduces the notion of the distributional density of two concepts, a and b, as the distributional similarity of the pairwise combination of all concepts closely situated to the two concepts in the ontology. Two possible expressions of what closely situated implies are suggested, namely as the average of the distributional similarity of all the concepts on the path from a to b via either their least upper bound or their greatest lower bound, and as the average distributional similarity of concepts at a certain distance from either a or b. Future work should include a measure of

distributional density based on semantic expansion as described in chapter 4, rather than on edge distance. An open research question is naturally how much the different definitions of distributional density positively affect information retrieval based on different semantic similarity measures.

An issue that has yet to be examined is what the optimal definition of context is. With the first two approaches suggested, the attempt is to capture a more general relatedness as expressed in the corpus; hence, first order affinity concept pairs found using a sliding window might be most useful. With the integrated approach, on the other hand, the attempt is to find child-parent concepts that are used synonymously to a certain extent. Therefore, second order affinity word pairs using a syntactically derived context might be most useful.

Chapter 7

Index Expansion in the Vector Space Model

One of the main reasons for introducing ontology-based information retrieval is the possibility of retrieving documents semantically related to the query rather than similar documents based on pure lexical match. Chapter 2, for instance, used the example *"I made her duck"* and *"she cooked me drake"*. Though the two sentences has very little in common on a lexical level they are semantically similar because *drake* is special kind of *duck*, and *make* and *cook* are synonyms. A match between the query *I made her duck* and the document containing *she cooked me drake* can be achieved by expanding the query with related terms as defined by the ontology. If WordNet 3.0 (WordNet 2009) is used as the basis, and we chose to expand nouns and verbs with synonyms and immediate specializations, the query *I made her duck* can expanded to the query:

> {*I, made, cooked, fixed, prepared, her, duck, drake, quack-quack, duckling, diving duck, dabbling duck, dabbler, mallard, Anas platyrhynchos, black duck, Anas rubripes, teal, widgeon, wigeon, Anas penelope, shoveler, shoveller, broadbill, Anas clypeata, pintail, pin-tailed duck, Anas acuta, sheldrake, ruddy duck, Oxyura jamaicensis, bufflehead, butterball, dipper, Bucephela albeola, goldeneye, whistler, Bucephela clangula, canvasback, canvasback duck, Aythya valisineria, pochard, Aythya ferina, redhead, Aythya americana, scaup, scaup duck, bluebill, broadbill, wild duck, wood duck, summer duck, wood widgeon, Aix sponsa, mandarin duck, Aix galericulata, muscovy duck, musk duck, Cairina moschata, sea duck*}[1]

The expanded query would provide the basis for at least a partial lexical match to the document containing *she cooked me drake* because *drake* is added to the query.

[1] Please note that this is example contains a simplification in that we have ignored morphology by expanding to verbs in the same tense as *made*.

Thus by expanding the query based on the ontology we have made it possible to retrieve semantically related documents to the original query *I made her duck*. An obvious alternative to expanding the query is to expand the index, and this chapter will present a previously unpublished model for ontology-based expansion of the index in the vector space model. The principle idea in the presented model is to expand the index rather than the query in order to make the expansion sensitive to the discriminative power of the concepts included in the expansion. This is achieved by introducing a new generalized measure of term frequency and the parameter λ that emphasizes either lexical matching or conceptual matching. The presented model has three major strengths. First, it can with relative ease be adopted in information retrieval systems already based on the vector space model because it only modifies the term weighting, and thus leaves the rest of model unchanged. Second, the model takes of advantage of expanding at the time of indexing by including the resolution power of the concepts in the expansion.

The chapter will begin with presenting some basic considerations in ontology-based query expansion, and then continue to the presentation of the model for index expansion.

7.1 Query Expansion

Query expansion or query refinement has been an active research area for a long time. Query expansion is the process of supplementing the original query with additional terms, and it can be done manually, automatically or interactively also known as semi-automatic, user mediated, and user assisted respectively (Efthimiadis 1996). The focus here is not a general account of the various perspectives on query expansion, but to give the reader a basic understanding of ontology-based automatic query expansion. This will in turn serve as the basis for understanding how expansion can be used in the model for index expansion considered in the next section.

In ontology-based query expansion the selection of good concepts for query expansion is based on their similarity to the original concept c in the query[2]. Identifying similar concepts can be performed by including closely related concepts defined by the ontology as expressed by nearness in the ontology. The original concept is used as a starting point and the concepts most similar to this are added to the description. The fundamental assumption is that concepts situated close together in the ontology are more closely related than concepts situated far apart.

The principle of concept expansion is illustrated in figure 7.1 with the descriptor *dietary*[CHR : *treatment*], where the light gray items indicate a decreasing degree of similarity compared to the original concept measured as simple path length.

Chapter 3 discusses why the logical reasoning capabilities in some aspects of

[2]Here we shall only consider a query with one concept but the formalism introduced here is straightforward to generalize to sets of concepts.

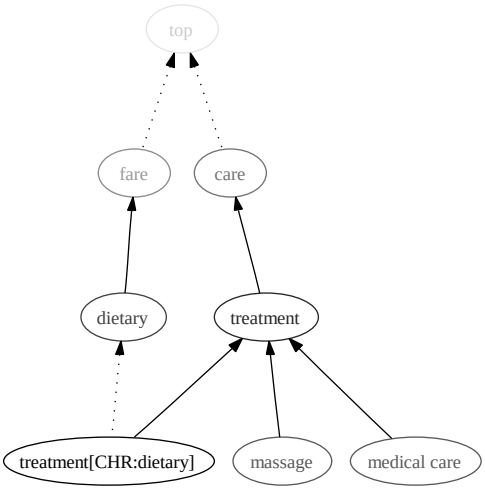


Figure 7.1: An illustration of the principle of the semantic expansion of the concept *treatment*[CHR : *dietary*], where the gray shading indicates a decreasing degree of similarity with the original concept

ontology-based information retrieval are less important and interesting than the graph representation of the ontology. Figure 7.1 also illustrates this issue.

Since the continuous addition of related concepts quickly can result in a large number of concepts being included in the expansion, the process typically continues until some stop criterion is met, allowing only pertinent concepts to be included in the expansion. As the example with $\{I$, *made*, *her*, *duck*$\}$ illustrated the expanded query can quickly become significantly larger than the original query. In general, the stop criterion can take two different forms. First, it can be a similarity threshold, which results in a stop in the spreading when the similarity of concepts is too low compared to the original concept. If the fuzzy set of similar concepts to concept c is defined as:

$$S = \{sim(c, c')/c' | c, c' \in C\} \tag{7.1}$$

then the set of expanded concepts based on the similarity threshold, π, can be expressed as the $\alpha - cut$ on S:

$$^{\alpha}S = \{c | \mu_S(c) \geq \alpha, \alpha = \pi\} \tag{7.2}$$

Similarity can be measured here by, e.g. path length, so concepts far away in the ontology will not be included. Alternatively, the addition of similar concepts to the query can be stopped when the expansion has reached a certain size measured by a threshold on the cardinality of the expanded query. Let τ denote the cardinality threshold, then the $\alpha - cut$ on S, which gives a cardinality less than τ, can be defined as:

$$\alpha = \begin{cases} 1 \text{ if } |^1S| > \tau \\ \min\{\beta | |^{\beta}S| < \tau\}, \text{otherwise} \end{cases} \tag{7.3}$$

The similarity threshold ensures a relatively small set of nodes only if a few concepts have a high similarity with c. On the other hand, if many concepts have a high similarity with c, only the cardinality threshold ensures a relatively small set. The challenge, however, is that both problems can occur within the same taxonomy. Therefore, a conjunction of the two thresholds will in most cases be appropriate. The set of expanded concepts can thus be defined as:

$$^{\alpha}S, \text{ where } \alpha = \min\{\beta || ^{\beta}S| < \tau, \alpha > \pi\} \tag{7.4}$$

For a recent and very thorough overview of ontology-based query expansion see Bhogal et al. (2007) which also describes numerous experiments applying either the similarity or the cardinality threshold. Another important and often cited reference on the topic is Voorhees (1994) which showed that assigning lower weights to related concepts improves retrieval accuracy, and that automatic query expansion using weighted synsets and hyponyms yielded the best results. Mihalcea & Moldovan (2000) achieved even better results with unweighted synset-based expansion which is likely due to an improved word sense disambiguation compared to Voorhees (1994).

That synsets and hyponyms yields the best results in query expansion is supported by experiments with thesaurus-based query expansion. However in thesaurus-based query expansion the relations are lumped together in equivalence (synonymy), hierarchical (include both hyponymy and meronymy), and associative relationships (all other relations) (Kristensen 1993, Greenberg 2001*a*,*b*, Tudhope et al. 2006). The findings by Kristensen (1993), Greenberg (2001*a*,*b*) are that narrower terms and synonyms can increase recall at only a slight precision penalty. Broader terms and related terms naturally increase recall but results in lower precision. Tudhope et al. (2006) does not report in quantitative measures on their findings. Kristensen (1993), and Greenberg (2001*a*,*b*) both used unweighted expansion and include only concepts in the immediate vicinity of the original query. In other words the similarity threshold in the expansion is thus applied. Tudhope et al. (2006) on the other hand uses a weighted expansion and directly applies a similarity threshold in their expansion.

Besides being of general interest in ontology-based information retrieval, expansion plays an important role in the next section, which presents an ontology-based information retrieval model based on the vector space model. An important component here is the expansion of the index with related concepts.

7.2 An Ontology-Based Vector Space Model

A fundamental prerequisite for ontology-based information retrieval is obviously ontologies. Unfortunately, for a great number of languages, an expansive publically available ontology like WordNet and Cyc simply does not exist. Also, domain specific ontologies are a scarce resource even for English. For Danish the only available general language ontology for a number of years has been SIMPLE, which covers

an estimated 10,000 words. This is too little coverage for general information retrieval systems relying solely on the ontology for indexing and matching. Despite these obstacles, even a small ontology offers the addition of semantics to information retrieval. The challenge is thus to design flexible information retrieval models or techniques that provide the means for integrating keyword-based and ontology-based information retrieval.

An ontology-based vector space model will be introduced here. The idea is to expand the index with related terms based on a similarity measure, for instance, by adding *car* to the index if *jeep* and *coupe* appear in the document. The expansion of the index, rather than the query, has the advantage of making it possible to take the frequencies of the terms in the document collection into account, i.e. if both *jeep* and *coupe* appear in a document, *car* should probably be given a higher weight than if only one of them appears.

Before proceeding to the index expansion itself, let us briefly recapitulate from chapter 2. In the vector space model, the set of terms in the document collection constitutes the dimensions of the vector space, so the number of dimensions of the vector space is equal to the cardinality of the set of terms, $|T|$. The j'th document can be represented as a weighted vector where each weight, w_{ij}, indicates the weight of term t_i in document d_j as assigned by some index function, $index(d_j, t_i)$. This weight is typically some variation of the $tfidf$ as, for instance, given by:

$$\begin{aligned} tf_{i,j} &= \frac{f_{i,j}}{\max_l(f_{l,j})} \\ idf_i &= \log\left(\frac{N}{n_i}\right) \\ tfidf_{i,j} &= \frac{f_{i,j}}{\max_l(f_{l,j})} \cdot \log\left(\frac{N}{n_i}\right) \end{aligned}$$

where $f_{i,j}$ is the frequency of term t_i in document d_j, n_i is the number of documents in which term t_i appears and N is the number of documents in the collection. Document d_j can thus be described as the vector $\underline{d}_j$, and similarity between query and document vectors can, for example, be measured by the cosine.

7.2.1 A generalized $tfidf$ measure

In the ontology-based vector space model the term space, T, is the union of the set of terms in the document collection and the set of terms denoting concepts in the ontology. As a weighting scheme, we apply a generalized measure, $tfidf'$, where the frequency of a term is influenced by the frequency of related terms. How much the frequency is affected is determined by how similar the terms are. First an ontology-based term frequency measure f', as the local weight, is introduced followed by an

example of how to calculate the weight given a toy example. Subsequently we turn to a global weight in the form of an ontology-based inverse document frequency idf'.

Local weight

Let us assume a similarity measure of $sim(t_a, t_b) \in [0, 1]$ that expresses the similarity between the terms t_a and t_b. In the ontology-based expansion, this similarity is measured by how similar the underlying concepts are, derived, for instance, from the shortest path between the concepts. We propose that the generalized frequency, $f'_{i,j}$, of a term, t_i, can be calculated as:

$$f'_{i,j} = \sum_{k=1}^{m} sim(t_k, t_i) \times f_{k,j} \tag{7.5}$$

The generalized frequency, $f'_{i,j}$, of a term, t_i, is thus the sum of t_i's frequency and the frequency of conceptually related terms modified by their similarity to term t_i. Here m denotes the size of the vector space. When the similarity measure is strict (in the sense that it assigns a similarity of zero to anything but the terms itself), the frequency of a term given by equation 7.5 is identical to the frequency applied in standard $tfidf$. In all other cases, terms are expanded based on the similarity measure.

Global weight

With the term frequency in place, we now turn to inverse document frequency. The most direct approach would be to simply use the generalized frequency f'. The intuition being that if a term is in a document it should be counted fully towards its document frequency; this irrespective of whether the term is present in the document or whether its frequency is due to the expansion of related terms. In this approach, the number of documents where term t_i appears, n^*_i, can be expressed as:

$$n^*_i = |\{d_j | f'_{i,j} > 0\}| \tag{7.6}$$

The problem with using such an approach is that many terms, and especially the more general terms, will fully appear in most documents if the similarity measure is not restrictive. An alternative and more refined approach could be to use a weighted document frequency where documents are not counted fully if the term is only present due to the expansion of a related term. We propose that the weight of t_i in the document frequency, n'_i, could, for instance, be measured as the similarity of the most similar term, t_k, in the document, d_j, to term t_i:

$$n'_i = \sum_{j=1}^{N} \max_{t_k \in d_j} (sim(t_k, t_i)) \tag{7.7}$$

With N being the the size of document collection. The generalized term frequency inverse document frequency, $tfidf'$, is thus:

$$tfidf'_{i,j} = \frac{f'_{i,j}}{\max_l(f_{l,j})} \cdot \log\left(\frac{N}{n'_i}\right) \tag{7.8}$$

An example

In order to illustrate the generalized term frequency, f', consider table 7.1, which lists the frequency, $f_{i,j}$, of the different terms in the document collection d_1, d_2, d_3, and the ontology excerpt from WordNet in figure 7.2.

	f		
Document / Term	d_1	d_2	d_3
vehicle	0	0	11
motor vehicle	0	1	1
automotive vehicle	0	0	0
truck	0	4	16
car	6	19	0
auto	0	3	0
automobile	3	11	0
offroader	0	0	0
jeep	1	0	0
coupe	8	4	0

Table 7.1: A table of the frequencies, $f_{i,j}$, in the document collection of the terms in the ontology excerpt in figure 7.2.

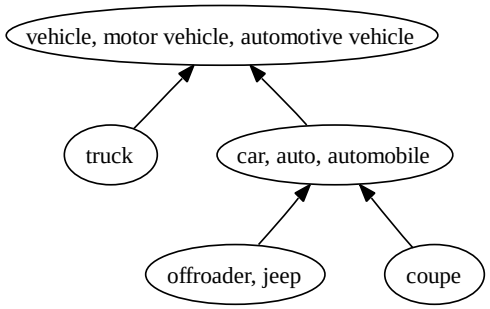


Figure 7.2: An ontology excerpt from WordNet.

Further, to keep things uncomplicated, we use the following simple similarity measure. The measure assigns a similarity of 1.0 to all synonyms, a similarity of 0.5

to immediate specializations, and a similarity of 0.0 to all other terms. For instance is $sim(vehicle, car) = 0.5$ and $sim(car, auto) = 1$ since *car* and *auto* are in the same synset. Table 7.2 lists the generalized frequencies, $f'_{i,j}$, based on the expansion by means of the ontology and by means of this similarity measure.

	f			f'		
Document / Term	d_1	d_2	d_3	d_1	d_2	d_3
vehicle	0	0	11	4.5	19.5	22.5
motor vehicle	0	1	1	4.5	19.5	22.5
automotive vehicle	0	0	0	4.5	19.5	22.5
truck	0	4	16	0	4	16
car	6	19	0	13.5	35	0
auto	0	3	0	13.5	35	0
automobile	3	11	0	13.5	35	0
offroader	0	0	0	1	0	0
jeep	1	0	0	1	0	1
coupe	8	4	0	8	4	0

Table 7.2: A table of the frequencies $f_{i,j}$ and $f'_{i,j}$ of the terms in the document collection.

For instance, the frequency of *coupe*, *car*, and *vehicle* in document d_2 by means of this similarity measure is given by:

$$\begin{aligned} f'_{coupe,d_2} &= 1.0 \times f_{coupe,d_2} \\ &= 1.0 \times 4 \\ &= 4 \end{aligned}$$

$$\begin{aligned} f'_{car,d_2} &= 1.0 \times f_{car,d_2} + 1.0 \times f_{auto,d_2} + 1.0 \times f_{automomobile\ vehicle,d_2} + \\ &\quad 0.5 \times f_{offroader,d_2} + 0.5 \times f_{jeep,d_2} + \\ &\quad 0.5 \times f_{coupe,d_2} \\ &= 1.0 \times 19 + 1.0 \times 3 + 1.0 \times 11 + \\ &\quad 0.5 \times 0 + 0.5 \times 0 + \\ &\quad 0.5 \times 4 \\ &= 35 \end{aligned}$$

$$
\begin{aligned}
f'_{vehicle,d_2} &= 1.0 \times f_{vehicle,d_2} + 1.0 \times f_{motor\ vehicle,d_2} + 1.0 \times f_{automotive\ vehicle,d_2} + \\
&\quad 0.5 \times f_{truck,d_2} + \\
&\quad 0.5 \times f_{car,d_2} + 0.5 \times f_{auto,d_2} + 0.5 \times f_{automobile,d_2} \\
&= 1.0 \times 0 + 1.0 \times 1 + 1.0 \times 0 + \\
&\quad 0.5 \times 4 + \\
&\quad 0.5 \times 19 + 0.5 \times 3 + 0.5 \times 11 + \\
&= 19.5
\end{aligned}
$$

Using equation 7.8 the $tfidf'$ of *coupe*, *car*, and *vehicle* in document d_2 can be calculated as:

$$
\begin{aligned}
tfidf'_{coupe,d_2} &= \frac{4}{35} \log\left(\frac{3}{2}\right) \\
&= 0.07 \\
tfidf'_{car,d_2} &= \frac{35}{35} \log\left(\frac{3}{2}\right) \\
&= 0.58 \\
tfidf'_{vehicle,d_2} &= \frac{19.5}{35} \log\left(\frac{3}{2.5}\right) \\
&= 0.15
\end{aligned}
$$

The standard $tfidf$ weights would in contrast be:

$$
\begin{aligned}
tfidf_{coupe,d_2} &= \frac{4}{19} \log\left(\frac{3}{2}\right) \\
&= 0.12 \\
tfidf_{car,d_2} &= \frac{19}{19} \log\left(\frac{3}{2}\right) \\
&= 0.58 \\
tfidf_{vehicle,d_2} &= \frac{0}{19} \log\left(\frac{3}{1}\right) \\
&= 0.00
\end{aligned}
$$

The example containing *coupe*, *car*, and *vehicle* illustrates some important aspects of the proposed approach.

First, it shows how conceptually related terms are added to the index. *Vehicle* does not appear as a term in either document 2 or 3, but is added to the index because of the expansion of the synonym *automotive vehicle* and the specializations *car*, *auto*, and *automotive*. Naturally, the exact choice of similarity measure is of great importance, since it determines the weight of the terms that are the result of an expansion. If one compares the $tfidf$ and $tfidf'$ of *coupe* and *car*, it also becomes clear that the

weight of *car* relative to the weight of *coupe* increases. This is due to the fact that document d_2 simply contains more on *cars* if one takes related terms into account.

Second, it is clear, all things being equal, that the resolving power of many terms will be smaller simply because they appear in more documents due to the expansion. The resolving power of *vehicle* is thus reduced because it appears in all three documents after the index expansion if the more naive approach of measuring document frequency is used. The addition of related terms has thus come at the cost of reducing the resolving power of these related terms. However, this challenge is met by introducing a more refined weighted inverse document frequency.

Third, unless a special similarity measure is devised, synonyms are given the same weight. All the ontology-based similarity measures presented in chapter 6 treated terms denoting the same concepts as identical. From a general information retrieval perspective, the description of the documents would have a higher fidelity if the terms actually appearing in the documents where emphasized in some manner - even though, from a more strict ontological perspective, it is irrelevant which terms actually denote the concepts. In the following, we thus refine the approach in order to make it possible to emphasize lexical matching. By emphasizing lexical matching, the problem of having to devise special similarity measures to tackle the synonyms issue can be avoided, and emphasizing lexical matching is also one method for controlling the influence of the expansion on the term weights.

7.2.2 Emphasizing lexical match

In order to make it possible to emphasize lexical matching, we initially describe the documents using two vectors. The first vector is the standard unexpanded term vector, $\underline{d}$, where term weights are the standard $tfidf$. The second vector, $\underline{d}'$, contains the expansion of the terms in $\underline{d}$, where the $tfidf$ is based on a modification of the term frequency, $f'_{i,j}$:

$$f'_{i,j} = \sum_{k=1,k\neq i}^{m} sim(t_i, t_k) \times f_{k,j} \tag{7.9}$$

The only difference between equation 7.5 and equation 7.9 is that the frequency of the term is not part of the sum, $k \neq i$. Since the frequency of the term is not part of the sum, the weighting of terms in $\underline{d}'$ is only based on the frequencies of related terms. Using the simple similarity measure described above, the frequencies of related terms are shown in table 7.3.

In order to emphasize lexical or conceptual matching, documents and queries can now be represented as the sum, $\underline{d}''$, of the two vectors, $\underline{d}$ and $\underline{d}'$:

$$\underline{d}''_j = \underline{d}_j + \lambda \underline{d}'_j, \lambda \in [0; 1] \tag{7.10}$$

λ adjusts the weight of the term and the related terms in the matching. Thus λ adjusts the weight of the ontological contribution to the match. In the case of $\lambda = 1$, related

Term \ Document	1	2	3
vehicle	4.5	19.5	11.5
motor vehicle	4.5	18.5	21.5
automotive vehicle	4.5	19.5	22.5
truck	0	4	16
car	7.5	14	5.5
auto	13.5	32	5.5
automobile	10.5	24	5.5
offroader	1	0	0
jeep	0	0	1
coupe	8	4	0

Table 7.3: A table of the frequencies, $f'_{i,j}$, of related terms based on equation 7.9.

terms and the terms themselves are considered to be of equal importance. In all other cases where $\lambda < 1$, lexical matching is emphasized.

7.3 Related work

Some of the more well-known work in this area includes that of Voorhees (1994, 1999) on query extension using WordNet. Voorhees uses a $tfidf$ weighted vector to represent documents but an extension of the vector space model to represent queries. A query is composed of up to 11 subvectors of the same size as the document vectors, where these subvectors are a representation of the original stems of the query, the expansion through synsets and the expansion through each one of the different kinds of relations in WordNet. Similarity is the sum of the similarity between the composed query vector and the document vector. The studies showed that semantic expansion can degrade performance and that inadequate word sense disambiguation plays a vital role in this respect. All the later work cited here supports these findings.

Gonzalo et al. (1998) index in the vector space model using WordNet synsets rather than terms, but they do not expand the index with related synsets. The exact weighting of the synsets is unaccounted for. The experiments show a remarkable improvement of 29% in precision, but this might be due to a very special test collection where the queries were summaries of SEMCOR documents. Precision is measured as the system's ability to return the document belonging to a summary as the most relevant given the summary as the query.

Mihalcea & Moldovan (2000) combine lexical and semantic indexing in the Boolean model. The index is based on the stem of the word, its part of speech and the ID of the synset to which the word belongs. The results are encouraging, especially concerning automatic word sense disambiguation, where the work shows that word sense

disambiguation has achieved an acceptable level for doing ontology-based information retrieval. The issue of word sense disambiguation is somewhat orthogonal to the question of semantic expansion, and there is little doubt that irrespective of what specific expansion scheme is used, word sense disambiguation is a fundamental building block of information retrieval systems incorporating semantic expansion.

The work of Hotho et al. (2003) is the most similar to the model presented in this chapter, although their focus is on clustering instead of information retrieval. Hotho et al. represent documents as vectors in a vector space but with the difference that the space of terms is extended or concatenated with a conceptual space. With respect to indexing, they test three different strategies:

1. The term vector, $\underline{t}$, is extended with a concept vector, $\underline{c}$, based on synsets.
2. All terms appearing in $\underline{c}$ are removed from $\underline{t}$, leaving the rest of the terms.
3. The documents are only represented by the extension of the term vector, $\underline{c}$.

A concept weighting is also presented based on the sum of the raw concept frequency and the summed frequency of all subsumed concepts up to a given level measured by the shortest path. The results of the experiments show that the addition of conceptual indexing with some expansion of generalized concepts improves clustering. Unfortunately, the results of the replacement strategy (2) are unaccounted for.

One of the differences between the model presented in this chapter and the related work described here is in the integration of the term space and the concept space. In contrast to Hotho et al. (2003), who introduce a conceptual subspace concatenated with the term space, the model presented in this chapter integrates the two levels of description in one space. This space, however, can possibly be extended with concepts having no lexical forms in the corpus. Why Hotho et al. chose this concatenated space is not explained further, but having a separate concept space and lexical space is naturally more flexible than treating them as one space. One drawback, however, is that the cardinality of the vector space increases, thereby increasing the complexity of measuring similarity.

Another difference is the expansion of the concepts and the concept weighting. The extension of the query done by Voorhees (1994) has the possibility of emphasizing certain relations by weighting each of the subvectors, which is also tested. The weighting of the terms is simply $tfidf$, and in the subvectors the concepts are weighted based on their raw occurrence in the document excluding any expansion. Hotho et al. (2003) use the upwards expansion of a concept and include this expansion in the calculation of concept frequency. Our model differs in this respect in that it is entirely flexible as to what kind of concept expansion is preferred. Moreover, the addition of a measure of the specificity of the expansion , the idf measure, is a great improvement of the weighting scheme. For instance, an expansion strategy of upwards expansion will not result in all documents being similar due to shared concepts at a high conceptual level. The global weighting thus ensure that concepts with a low

resolving power will not adversely effect the precision of the retrieved results. A final difference is the smoother scaling of the term-based index versus the concept-based index as introduced by lambda in equation 7.10.

The introduction of term weighting is the main difference between the approach presented here and he fuzzy information retrieval using a fuzzy thesaurus presented in 2. However, the idea of weighting the ontology-based expansion presented here could also have been framed within the model for fuzzy information retrieval using a fuzzy thesaurus. It is thus not the model that is essential here but the term weighting.

In the presentation of the ontology-based vector space model it has been presumed that the similarity measure was based on the ontology. This was done intentionally since the purpose has been to design flexible methods for combining keyword-based and ontology-based information retrieval. However, there is nothing inherent in the model model that requires the similarity measure to be ontology-based. Chapter 6 presents distributional similarity measures based on the co-occurrence of words or concepts in a document collection with no knowledge of their ontological relatedness. For instance, *oil* and *OPEC* often co-occur but are situated far apart, at least along the hypernymy/hyponymy relations. Statistical knowledge in the form of co-occurrence patterns could thus also be used in the expansion.

The generalized $tfidf$ weighting and the flexibility provided by representing each document and query as $\underline{d}''$ in our ontology-based modification of the vector space model mean we have succeeded in providing a mean for integrating keyword-based and ontology-based information retrieval.

7.4 Discussion and Summary

Ontology-based information retrieval offers a semantic matching not possible with information retrieval relying solely on the lexical level for the representation of meaning. This is the motivation for the approach to ontology-based index expansion in the vector space model introduced in this chapter. The approach consists of a generalized term frequency inverse document frequency which makes it possible to expand the index by using any kind of similarity measure. The approach presented also consists of a flexible weighting scheme which allows an emphasis on either conceptual or lexical matching. Besides the model, the chapter presented the automatic query expansion framed in fuzzy logic.

Besides the flexible weighting scheme noted above the model introduced here has two advantages. First, from a system architectural viewpoint it can be adopted with relative ease into existing information retrieval systems. Only the term weighting part of the indexing component needs modification, and the rest of the system can be used unchanged. Given the popularity of the vector space model Meadow et al. (2007), Baeza-Yates & Ribeiro-Neto (1999) it is important for the advance of ontology-based information retrieval that the models can be easily integrated in the existing systems. A second advantage of the model presented in this chapter is the expansion of the

index rather than the query. This is the most important feature because it enables the expansion to take into account the resolving power of the expansion. Continuing on the example from the beginning of the chapter with $\{I, made, her, duck\}$. It will adversely effect the precision of the results if *duck* has a high resolving power and *drake* has a very low resolving power. Thus taking into account the resolving power of the expansion is highly important, and this can not be archived through regular query expansion; we need to expand the index.

One of the limitations of the model introduced is its limited ability to deal with the relations. Different kinds of relations can be used differentiated in the indexing by means of the similarity measure but then the information about the nature of the relations is discarded. As a result, if a document contains the two compound concepts $car[\text{CHR}: blue]$ and $television[\text{LOC}: kitchen]$, it would not be possible to infer from the index vector whether *big* or *kitchen* refers to *car* or *television*. The model is consequently unable to utilize fully the noun phrase analysis presented in chapter 4.

A final interesting perspective in using the vector space model is that most machine learning algorithms for, e.g. text classification, presume a fixed length feature space. The model presented here also makes it possible to easily integrate ontologies in machine learning disciplines like text classification or clustering.

Chapter 8

Conceptual Summaries

In information retrieval systems, a common method for visualizing the search result is a ranked list of references to documents that match the query. Most modern search engines also supply the references with small text snippets with excerpts from the documents. By examining this list, the user can try to identify which documents are relevant, and by browsing through the listed documents, get an idea of what the document collection in general contains about the topic(s) specified in the query. For instance, a query for "USA" might reveal that a particular document collection contains, for the most part, documents on geographical rather than political aspects of the country.

This chapter presents an alternative method for browsing the content of a set of documents called "conceptual summaries" in order to provide an option for navigating the conceptual content of a set of documents by means of a higher level conceptual summary of the content. The main concern is approaches for summarization given a set of concepts already extracted from a set of documents through some form of ontological indexing. In this case, summarization can be viewed as a process of transforming sets of similar low level objects into more abstract conceptual representations. More specifically, a summary for a set of concepts is an easy to grasp short description in the form of a smaller set of concepts. The characteristics of summaries are not looked at in more detail here, but Yager & Petry (2006) offer further considerations on this issue. We loosely assume, however, that if the set of concepts covers several distinct aspects, a summarizing description should include them. Thus, intuitively, if there are two distinct aspects, as in $\{convertible, van, cottage, estate\}$, the summary should probably have two concepts, $\{car, house\}$. With one single aspect, as in $\{poodle, Alsatian, golden\ retriever, bulldog\}$, the summary should have only a single summarizer $\{dog\}$. In principle, summaries are intended to describe any collection of texts, including a single document, a set of documents, a query result or an entire text base.

Two different approaches to conceptual summaries are presented in this chapter. For both approaches, an ontology plays a key role as a reference for the conceptual-

ization. The general idea is to form a so-called "instantiated ontology" to be used as a basis for summarization. An instantiated ontology is an excerpt of a world knowledge ontology restricted to a set of concepts that, for example, appear in a result set.

After having presented the instantiated ontology, we look at a strictly ontology-based approach where summaries are derived solely from the connectivity in the instantiated ontology. The presentation is, for the most part, a rendering of what has previously been presented in Andreasen et al. (2008). Second, we consider the conceptual clustering of the instantiated concepts based on their division into groups or clusters based on a semantic similarity measure, and, for each a cluster, derive a representative concept. The representative concept can be the least upper bounds or what we will refer to as the "supported least upper bounds" and "fuzzy least upper bounds" of the clusters. The majority of this is a restatement of research presented in Bulskov et al. (2007). It is important to notice that the work presented here is not concerned with ontology-based clustering of documents. Much work on this has been done before especially by Hotho & Stumme Hotho (2005), Hotho et al. (2002, 2003), Hotho & Stumme (2002) and more recently by e.g. Zhang et al. (2008), Recupero (2007), Jing et al. (2006). Neither is it concerned with selecting natural language excerpt from documents using ontology-based analysis of the documents in order to select representative natural language sentences that can be given as a summary. Work on this has also been done before by e.g. Wu & Liu (2003), Lee et al. (2005). The work here is based on the novel idea that a set of concepts from the ontology is presented as a summary for a given collection of documents. Besides providing a general alternative to browsing and providing an overview of a result set, the application of such conceptual summaries are situations where an alternative to natural language summary is desired. This could for instance be in research where a summary of a collection of papers is required, e.g. what are the main topics being treated and how are they related. It could also be in a company setting where news surveillance is being performed on a daily basis, e.g. which products are mentioned in the news in connection with what.

8.1 Instantiated Ontologies

In this section, we present the notion of instantiated ontologies as described in Andreasen et al. (2005*b*). In general, the set of well-formed terms, $\mathcal{L}$, in ONTOLOG is (Andreasen & Bulskov 2007*a*):

$$\mathcal{L} = \{C\} \cup \{a[r_1 : b_1, \ldots, r_n : b_n] | a \in C, r_i \in R, b_i \in \mathcal{L}\} \tag{8.1}$$

For instance, with R= {WRT, CHR, CBY, TMP, LOC, ...} [1] and C={*entity, physical_entity, abstract_entity, location, town, cathedral, old*}, we get:

$$\begin{aligned}\mathcal{L} \;=\; & \{entity, physical\_entity, abstract\_entity,\\ & location, town, cathedral, old,\\ & \ldots, \textit{cathedral}[\text{LOC}: \textit{town}, \text{CHR}: \textit{old}], \ldots\\ & \textit{cathedral}[\text{LOC}: \textit{town}[\text{CHR}: \textit{old}]], \ldots\}\end{aligned}$$

Given the world knowledge ontology, O, and a set of concepts, C, the instantiated ontology $O_C = \langle \mathcal{L}_C, \leq_C, R\rangle$ is a restriction of O to cover only the concepts in C and corresponds to "upper expansion" $\mathcal{L}_C$ of C in O:

$$\begin{aligned}\mathcal{L}_C &= C \cup \{x | y \in C, y \leq x\}\\ \text{"}\leq_C\text{"} &= \text{"}\leq\text{"} \cap (\mathcal{L}_C \times \mathcal{L}_C) = \{<x, y> | x, y \in \mathcal{L}_C, x \leq y\}\end{aligned}$$

Figure 8.1 shows an example of an instantiated ontology. The general ontology is based on (and includes) WordNet and the ontology shown is "instantiated" with respect to the following set of concepts:

$$\begin{aligned}C \;=\; & \{\textit{ruin}, \textit{church}, \textit{fortress}[\text{CHR}: \textit{big}], \textit{stockade},\\ & \textit{fortification}[\text{CHR}: \textit{large}, \text{CHR}: \textit{old}]\}\end{aligned}$$

An instantiated ontology like this forms the basic structure in both of the clustering approaches. A presentation of the connectivity clustering is given first. Apart from the inclusion relation, "$\leq$", the relation "$<$" is used in the following. The latter refers to the strict variant of the former.

8.2 Connectivity Clustering

Connectivity clustering is clustering based solely on connectivity in the ontology, O_C. Specifically, the idea is to cluster a given set of concepts based on their connections to common ancestors, for instance, grouping two siblings according to their common parent, and also replacing the group with the common ancestor. Thus, connectivity clustering is about moving towards a smaller number of more general concepts, rather than moving towards a smaller number of larger clusters as typically is the case in bottom-up hierarchical clustering.

For a set of concepts, $C = \{c_1, \ldots, c_n\}$, we can consider as a *generalizing description*, a new set of concepts, $\delta(C) = \{\widehat{c}_1, \ldots, \widehat{c}_k\}$, where $\widehat{c}_i$ is either a concept generalizing concepts in C or an element from C. Each generalizer in $\delta(C)$ is a least

[1] *With respect to* (*wrt*), *characterized by* (*chr*), *caused by* (*cby*), *temporal* (*tmp*), *location* (*loc*).

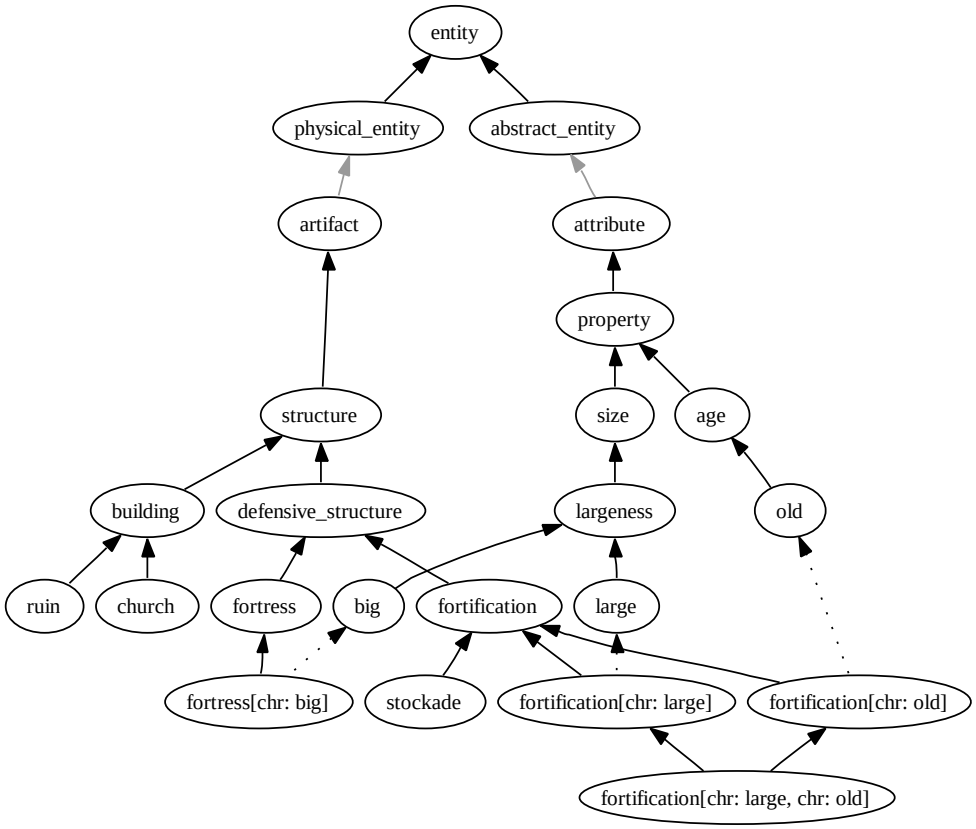


Figure 8.1: An instantiated ontology based on the WordNet ontology and the set of instantiated concepts {*ruin*,*church*,*fortress*[CHR: *big*], *stockade*,*fortification*[CHR: *large*, CHR: *old*]}.

upper bound of a subset of C, $\widehat{c}_i = lub(C_i)$, where $\{C_1, \ldots, C_k\}$ is a division (clustering) of C. Note that the least upper bound of a singleton set is the single element in this.

Summarization is here an iterative process where at each step the summary in the form of a set of descriptors are reduced to the most *most specific generalizing description*. We define the *most specific generalizing description*, $\delta(C)$, for a given $C = \{c_1, \ldots, c_n\}$ as a description restricted by the following properties :

(a) $\forall \widehat{c} \in \delta(C) : \widehat{c} \in C \vee \exists c', c'' \in C \wedge c' \neq c'' \wedge c' <_C \widehat{c} \wedge c'' <_C \widehat{c}$

(b) $\forall \widehat{c}', \widehat{c}'' \in \delta(C), \widehat{c}' \neq \widehat{c}'' : \widehat{c}' \not<_C \widehat{c}''$

(c) $\forall c', c'' \in C, c' \neq c'', \widehat{c}' \in \delta(C), \neg \exists x \in \mathcal{L}_C :$
$c' \leq_C x \wedge c'' \leq_C x \wedge x \leq_C \widehat{c}'$

(d) $\forall c \in C, \exists \widehat{c} \in \delta(C) : c \leq \widehat{c}$

Thus, (a) restricts $\delta(C)$ to elements that either originate from C or generalize two or more concepts from C. Second, (b) restricts $\delta(C)$ so that it is without redundancy (no element of $\delta(C)$ may be subsumed by another element). Third, (c) reduces $\delta(C)$ to the most specific concept in the sense that no subsumer for two elements of C may be subsumed by an element of $\delta(C)$. Finally, (d) ensures $\delta(C)$ covers all the elements in C.

Note that $\delta(C)$ has the same form as C as a subset of $\mathcal{L}_C$, and that we can thus refer to an m'th order summarizer, $\delta^m(C)$. Obviously, to obtain an appropriate description of C, we will in most cases need to consider higher orders of δ. At some point, m, in most cases, $\delta^m(C) = Top$, where Top is the top element in the ontology. Exceptions are when a more specific single summarizer is found or when *Top* has only one successor. In the latter case, we only reach the single topmost concepts with more than one successor.

The most specific generalizing description, $\delta(C)$, for a given C is obviously not unique and there are several different sequences of most specific generalizing descriptions of C from C towards Top. For instance in the instantiated ontology in figure 8.1 could *ruin* and *church*, *stockade* and *fortification*[CHR : *large*, CHR : *old*], or both be summarized in the first step. A possible approach is to take the largest possible steps as is done in algorithm 8.1.

Input: Set of concepts $C = \{c_1, \ldots, c_n\}$
Output: Generalizing description $\delta(C)$ for C.

1. Let the instantiated ontology for C be $O_C = \langle \mathcal{L}_C, \leq$
2. Let $U = \{u | u \in \mathcal{L}_C \wedge \exists c_i, c_j \in C : c_i <_C u \wedge c_j <$
3. $U' = U - \{u | u \in U \wedge \exists v \in U : v <_C u\}$, and
4. $L = \cup_{u \in U'} \{c | c \in C \wedge c <_C u\}$
5. Set $\delta(C) = C \cup U' - L$

Algorithm 8.1: Connectivity clustering

C is presumed to be without redundancy, i.e. there are no concepts in C that subsume other concepts. If the instantiated ontology is looked at as a graph, this corresponds to having only concepts represented by leaf nodes in C. In step 2) in the algorithm, all concepts, U, that generalize two or more concepts are derived. U thus contains all the elements that satisfies property (a) in that they originate from C or generalize two or more concepts. 3) reduces U to the most specific generalizers, U'. The elements in U' thus satisfies both property (a), (b), and (c) in that we from U remove all elements that subsume other elements. In order to find the most specific generalizing description

of C, that covers C and thus satisfies property (d), we need a union of C and U' without all the concepts from C subsumed by elements in U'. Thus 4) defines the set of concepts, L, that specializes the generalizers in U'. Finally 5) derives $\delta(C)$ from C by adding the most specific generalizers and subtracting concepts specializing these. The result is a summary $\delta(C)$ which satisfies all the four properties.

With reference to the instantiated ontology presented in figure 8.1, we have, for instance:

$$
\begin{aligned}
C &= \{\textit{ruin}, \textit{church}, \textit{fortress}[\text{CHR}: \textit{big}], \textit{stockade}, \\
&\quad \textit{fortification}[\text{CHR}: \textit{large}, \text{CHR}: \textit{old}]\} \\
U &= \{\textit{building}, \textit{fortification}, \textit{structure}, \textit{defensive\_structure}, \\
&\quad \textit{artifact}, \textit{physical\_entity}, \textit{entity}\} \\
U' &= \{\textit{building}, \textit{fortification}\} \\
L &= \{\textit{ruin}, \textit{church}, \textit{stockade}, \textit{fortification}[\text{CHR}: \textit{large}, \text{CHR}: \textit{old}]\} \\
\delta(C) &= \{\textit{building}, fortification, \textit{fortress}[\text{CHR}: \textit{big}]\} \\
&\ldots \\
\delta^2(C) &= \{building, defensive\_structure\} \\
\delta^3(C) &= \{structure\}
\end{aligned}
$$

This approach can be viewed as a greedy approach where every concept that can be grouped will be grouped. Thus, in the first step, both *ruin* and *church* are summarized to *building*, while *stockade* and *fortress*[CHR: *big*] is summarized to *fortification*. As noted, this approach is, of course, not the only one possible, and priority could be given to summarize specific clusters. *Deepness* and *support* are examples of important properties that might contribute to priority.

8.2.1 Prioritized connectivity clustering

The deepest concepts, the ones positioned at the greatest depth in the ontology, are structurally and, thereby often, also conceptually the most specific ones. Thus, collecting these first would probably lead to a better balance with regard to how specific the participating concepts are in candidate summaries. Alternatively, support for candidate summarizers could be considered. One option is simply to measure support in terms of the number of subsumed concepts in the input set while more refinement could be obtained by also taking the frequencies of concepts as well as their distribution in documents in the original text into consideration.[2] Support indicates how much a concept covers in the input and can thus be considered as an importance weight for the concept as a summarizer for the input. High importance should probably infer more reluctance with regard to further generalization.

[2]Corresponding to term and document frequencies in information retrieval.

Let $f(x)$ be a function which assigns a priority to the most specific generalizers based on the principle of assigning the highest priority to the generalizers that generalize the least. This can, for instance, happen by letting, $f(x)$, be the depth in the ontology measured as the path link to the top concept:

$$f(x) = depth(x) \tag{8.2}$$

or as the inverse support of a concept $f(x)$ with respect to a given set of concepts, C, by:

$$f(x) = support(x, C) = \left(\frac{|\{y|y \in C, y \leq x\}|}{|C|} \right)^{-1} \tag{8.3}$$

A prioritized connectivity algorithm can then be expressed in the following manner:

Input: Set of concepts $C = \{c_1, \ldots, c_n\}$
Output: Generalizing description $\delta(C)$ for C.

1. Let the instantiated ontology for C be $O_C = \langle \mathcal{L}_C, \leq$
2. Let $U = \{u|u \in \mathcal{L}_C \wedge \exists c_i, c_j \in C : c_i <_C u \wedge c_j <$
3. $U' = \{u|u \in U \wedge \exists v \in U : v <_C u\}$,
4. $U'' = U' - \{u|u \in U' \wedge \exists v \in U : f(v) < f(u)\}$, an
5. $L = \cup_{u \in U''} \{c|c \in C \wedge c <_C u\}$
6. set $\delta(C) = C \cup U'' - L$

Algorithm 8.2: Prioritized connectivity clustering

In 2) all concepts, U, that generalize two or more concepts are derived. Note that these may include concepts from C when C contains concepts subsuming other concepts. 3) reduces U to the most specific generalizers, U'. 4) removes the concepts from U' with the lowest support or that are deepest in the ontology. 5) defines the set of concepts, L, that specialize the generalizers in U''. 6) derives $\delta(C)$ from C by adding the most specific generalizers in U'' and subtracting concepts specializing these.

The suggested functions are naturally only examples of how to express the properties deepness and support, and other functions expressing other properties are straightforwardly included in prioritized connectivity clustering. Another property could thus be *coherence*, which for a given generalizer, g, expresses to what extent the support of g is due to concepts being structurally close to or further away from g. This can, for instance, be measured as the average semantic similarity to g of the concepts generalized by g.

8.2.2 Connectivity clustering versus similarity clustering

Prioritized connectivity clustering opens up for a more fine grained approach to clustering than the greedy steps taken by the first connectivity clustering presented. However, connectivity clustering as such is strictly based on the hierarchy. As a result, the

size of the summary can never be reduced by generalizing non-leaf concepts in the instantiated ontology. For instance, in figure 8.2, *vehicle* and *trailer* are summarized by *transport* but not before *car* and *truck* are summarized by *vehicle*.

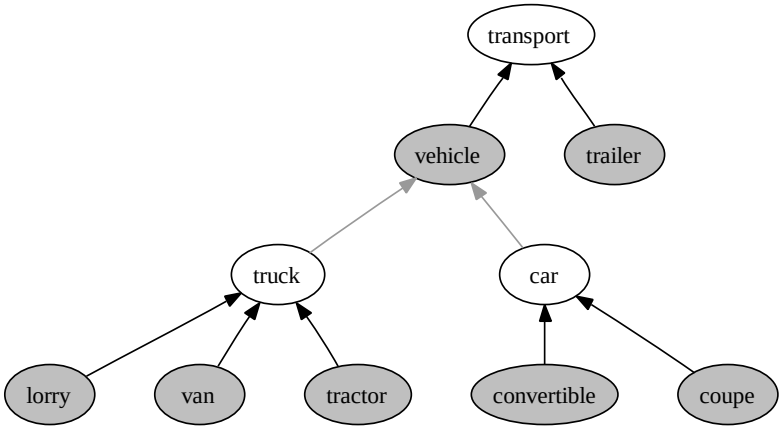


Figure 8.2: An instantiated ontology that can be used to illustrate the difference between connectivity clustering and similarity clustering. Shaded lines indicate longer paths and shaded concepts indicate concepts in C.

This is due to the definition of connectivity clustering, which in b) on page 108 restricts $\delta(C)$ from having redundancy (no element of $\delta(C)$ may be subsumed by another element). Thus, *truck* and *car* can never appear in the summary with *vehicle* or *transport*. In the case of the instantiated ontology depicted in figure 8.2, it might nevertheless be reasonable to replace *vehicle* and *trailer* with *transport* but keep *truck* and *car* in the summary, i.e. create a summary with the three clusters $\delta^m(C) = \{truck, car, transport\}$, which indicates that C covers *transport* in general but, to a large extent, covers the more specific concepts *trucks* and *cars*.

Similarity clustering can be seen as an alternative in that it offers a clustering of concepts based on a semantic similarity measure. If the similarity measure is derived at least partly from the ontology, a clustering based on this similarity measure will naturally reflect the relational knowledge of concepts embedded in the ontology.

8.3 A Hierarchical Similarity-Based Approach

Various approaches have been proposed to derive similarity or distance from the ontology. We will assume an ontology-based similarity measure, sim, below but make no further assumptions about the type and characteristics of this measure; see chapter 6. With a given similarity measure derived from the ontology, a least upper bound centered, agglomerative, hierarchical clustering can be performed as described in the following.

Initially, each cluster corresponds to a single element of the set to be summarized. At each particular stage the two clusters that are most similar are joined together. This is the principle of conventional hierarchical clustering. However, rather than replacing the two joined clusters with their union as in the conventional approach, they are replaced by their least upper bound. Thus, given a set of concepts, $C = \{c_1, \ldots, c_n\}$, summarizers can be derived as described in algorithm 8.3.

Input: Set of concepts $C = \{c_1, \ldots, c_n\}$
Output: Generalizing description $\delta(C)$ for C.

1. Let the instantiated ontology for C be $\mathcal{O}_C = \langle \mathcal{L}_C, \leq$
2. Let $T = \{\langle x, y\rangle | sim(x, y) = max_{z,w \in C}(sim(z, w)$
3. Let $U = \{u | u \in \mathcal{L}_C \wedge \exists \langle x, y\rangle \in T : x <_C u \wedge y <_C$
4. $U' = U - \{u | u \in U \wedge \exists v \in U : v <_C u\}$, and
5. $L = \{x | \langle x, y\rangle \in T \vee \langle y, x\rangle \in T\}$
6. Set $\delta(C) = C \cup U' - L$

Algorithm 8.3: Hierarchical clustering summary

In 2), all the concept pairs T with the highest similarity is found. 3) derives all concepts, U, that generalize a concept pair from T. 4) reduces U to the most specific generalizers, U'. 5) defines the set of concepts, L, that specializes the generalizers in U'. In other words L are all the concepts forming the pairs in T since U' are the most specific generalizers of the concepts in T. 6) derives $\delta(C)$ from C by adding the most specific generalizers in U' and subtracting concepts specializing these. As also was the case with connectivity clustering, δ might have to be applied several times to obtain an appropriate description of C. At some point, m, we have $\delta^m(C) = Top$.

8.3.1 A supported least upper bound approach

One straightforward similarity-based approach is simply to apply a crisp clustering to the set of concepts, $C = \{c_1, \ldots, c_n\}$, leading to $\{C_1, \ldots, C_k\}$ and then provide the set of least upper bounds, $\{\widehat{c}_1, \ldots, \widehat{c}_k\} = \{lub(C_1), \ldots, lub(C_k)\}$, for the division of C as a summary.[3] However, to also take into account the importance of clusters in terms of their sizes, the summary can be modified by the support of the generalizing concepts, with support as defined in equation 8.2. This leads to a fuzzy summary based on the division (crisp clustering) of C into $\{C_1, \ldots, C_k\}$:

$$\Sigma_i support(lub(C_i), C)/lub(C_i) \tag{8.4}$$

Note that if C had been defined as a multi-set, this definition of support would be identical to the probability estimates Resnik (1999) used in his similarity measure.

To illustrate the supported least upper bounds-based approach, consider the set of clusters and their least upper bounds in table 8.1, which shows the clusters resulting

[3]Note that we do not put any restrictions on the clustering, but of course the general idea is that the clustering applies a similarity measure that is ontology-based and that the ontology reflected is the instantiated ontology over the set of concepts, C.

Cluster	lub
size number	magnitude
government state committee	organization
defender man servant woman	person
cost bribe price fee	cost
fortress fortification stockade	defensive structure

Table 8.1: A set of crisp clusters and their least upper bounds from WordNet.

from applying $\delta^{11}(C)$ on an instantiated ontology from WordNet, with C naturally being all the concepts in the clusters. From these clusters, the fuzzyfied summary $\{$*0.13/magnitude + 0.19/organization + 0.25/person + 0.250/cost + 0.19/defensive structure*$\}$ can be generated.

This approach to summarization is not very tolerant with regard to noise in the clusters given. Consider the following example where *bribe* is replaced by *politics* and *stockade* by *radiator*. With the shortest path as the similarity measure, $\delta^{11}(C)$ will still result in the same clusters being formed. However, the least upper bounds of the respective clusters become more general, as illustrated in table 8.2

The summary thus becomes $\{$*0.13/magnitude+0.19/organization+0.25/ person+0.25/relation+0.19/artifac*
This summary is clearly more general than the summary for the more homogeneous clusters. *Radiator* and *politics* can therefore, to a certain extent, be regarded as semantic outliers or noisy elements from a summary perspective. To get around this problem, we introduce a soft definition of least upper bound and then combine this definition again with crisp clusters to obtain more robust cluster-based summaries.

8.3.2 A fuzzyfied least upper bound approach

A soft definition of least upper bound for a (sub)set of concepts, C, should comprise "upper boundness" as well as "leastness" (or "least upperness") expressing, respectively, the portion of concepts in C that are generalized and the degree to which a concept is least upper with regard to one or more of the concepts in C.

"Upper boundness" can be expressed for a set of concepts, C, by $\mu_{ub(C)}$ simply

	Cluster	lub	New Cluster	New lub
1	size number	magnitude	size number	magnitude
2	government state committee	organization	government state committee	organization
3	defender man servant woman	person	defender man servant woman	person
4	cost bribe price fee	cost	cost **politics** price fee	relation
5	fortress fortification stockade	defensive structure	fortress **radiator** stockade	artifact

Table 8.2: A set of crisp clusters with noise and their least upper bounds from WordNet.

as support with respect to C:

$$\mu_{ub(C)}(x) = support(x, C) \tag{8.5}$$

covering all generalizations of one or more concepts in C and including all concepts that generalize all of C (including the top concept Top) as full members.

"Leastness" can be defined on top of a function that expresses how close a concept is to a set of concepts, C, such as $dist(C, y) = min_{x \in C} dist(x, y)$, where $dist(x, y)$ expresses the shortest path upwards[4] from x to y, as follows:

$$\mu_{lu(C,\lambda)}(x) = \begin{cases} 1 & \text{when } \lambda = 0 \vee x = Top \\ 1 - \frac{dist(C,x)}{dist(C,Top)+\frac{1}{\lambda}-1} & \text{otherwise} \end{cases} \tag{8.6}$$

where $0 \leq \lambda \leq 1$ is a "leastness" parameter where $\lambda = 1$ corresponding to the most restrictive version of "leastness" and with the other extreme, $\lambda = 0$, corresponding to no restriction at all (all upper concepts become full members).

A soft least upper bound, $flub$, can now be defined by combining the two:

$$\mu_{flub(C,\lambda)}(x) = \mu_{lu(C,\lambda)}(x) * \mu_{ub(C)}(x) \tag{8.7}$$

Note that a least upper bound for C is not necessarily the best candidate among the elements in the $flub$. Thus, again with the division (crisp clustering) of C into $\{C_1, \ldots, C_k\}$, the basis for the summary here is the set of fuzzy sets $\{flub(C_1), \ldots, flub(C_k)\}$.

[4]Upwards only refers to paths consisting solely of edges in the direction of $\leq$. It must be strictly emphasized that the graph in question corresponds to the transitively reduced ontology.

As in the supported least upper bound approach, the summarizers can be weighted by support. To bring this support weighting into play, we can begin by invoking the weighting on the elements of C producing the fuzzy set W_C:

$$W_C = \sum_i \sum_{x \in C_i} \frac{|C_i|}{|C|} / x \tag{8.8}$$

And, second, by combining the union of the $flub$'s $flub(C_i)$ with W_C by fuzzy intersection leading to the $flub$-based summary:

$$\left(\bigcup_i flub(C_i) \right) \bigotimes W_C \tag{8.9}$$

where $\bigotimes$ is a t-norm, possibly with the product as an appropriate choice.

Given the previous example of noisy crisp clusters, and with λ being 1, the use of a $flub$-based summary gives:

$$\{.19/cost + .15/outgo + .15/relation + \\ .15/person + .13/organization + .13/government + \\ .11/financialloss + .11/artifact + \\ .11/defensivestructure + \ldots\}$$

where the supported least upper bound-based summary was:

$$\{.13/magnitude + .19/organization + \\ .25/person + .25/relation + .19/artifact\}$$

In the $flub$-based summary, *cost* has a high degree of membership due to the fact that it is a very good description of three of the four elements in the cluster {*cost, price, fee, politics*}. Thus, the introduction of the $flub$ reduces the effect of noise caused by the noisy element *politics*. Also, the degree of membership of *artifact* is comparable to the degree of membership of *defensive structure*, which is the immediate generalization of *fortress* and *stockade*. Again, the result of using a $flub$-based summary is that the effect of the noisy element *radiator* in the cluster {*fortress, stockade, radiator*} is reduced.

8.4 Summarization Examples with WordNet

Preliminary experiments have been performed on texts from SEMCOR 2.0 (Miller et al. 1994) on connectivity clustering as well as supported least upper bound hierarchical clustering. The purpose of this study has been to explore the utility of the two approaches. Given their preliminary nature, the current results of the findings are presented in the following using examples rather than descriptive statistics.

SEMCOR is a subset of the documents in the Brown corpus which has the advantage of being semantically tagged with senses from WordNet (Miller 1995). Below, we show the results of summarizations of the following text.

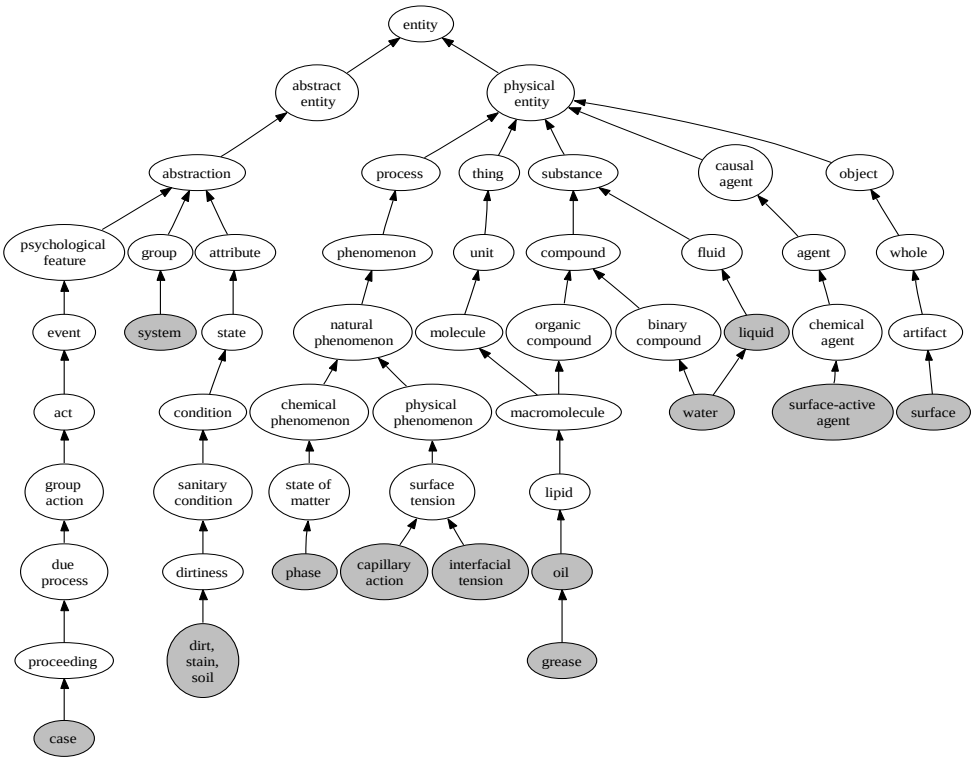


Figure 8.3: An instantiated ontology based on an excerpt from SEMCOR.

> *Greases*, *stains*, and miscellaneous *soils* are usually absorbed onto the *soiled surface*. In most *cases*, these *soils* are taken up as *liquids* through *capillary action*. In an essentially static *system*, an *oil* cannot be replaced by *water* on a *surface* unless the *interfacial tensions* of the *water phase* are reduced by a *surface-active agent*.

The words in italics indicate nouns that are mapped into WordNet. Note that due to the use of SEMCOR, there is no attribution of concepts (e.g. *system*$[\text{CHR}: \textit{static}]$) in the initial set of concepts.

The 17 nouns appearing in the paragraph are mapped into 12 WordNet synsets constituting the set of elements C. Based on this mapping, the instantiated ontology can be created as depicted in figure 8.3, where shaded nodes indicate concepts in C. The connectivity clustering thus includes these steps:

$$
\begin{aligned}
C &= \{\textit{case, system, dirt, phase, capillary action, interfacial tension, grease, oil, water, liquid, surface-active agent, surface}\} \\
\delta(C) &= \{\textit{abstraction, phase, surface tension, oil, water, liquid, surface-active agent, surface}\} \\
\delta^2(C) &= \{\textit{abstraction, natural phenomenon, liquid, compound, surface-active agent, surface}\} \\
\delta^3(C) &= \{\textit{abstraction, natural phenomenon, substance, surface-active agent, surface}\} \\
\delta^4(C) &= \{\textit{abstraction, physical entity}\} \\
\delta^5(C) &= \{\textit{entity}\}
\end{aligned}
$$

This example illustrates well some of the challenges that arise when applying connectivity clustering. First, the set of concepts are reduced by a larger factor in the first steps of the clustering process compared to the steps close to the end of the clustering process. This is due to the greedy nature of connectivity clustering where as many clusters as possible are merged at each step, and where clusters conceptually far apart are merged because the resulting cluster is the most specific generalizing description. The merge of *case, system* and *dirt* to the much more general *abstraction* illustrates this aspect. Second, small summaries tend, not surprisingly, to be very general.

Using the shortest path as the similarity measure, the hierarchical clustering introduced above follows these steps:

$$
\begin{aligned}
C &= \{\textit{case, system, dirt, phase, capillary action, interfacial tension, grease, oil, water, liquid, surface-active agent, surface}\} \\
\delta(C) &= \{\textit{case, system, dirt, phase, capillary action, interfacial tension, oil, liquid, surface-active agent, surface}\} \\
\delta^2(C) &= \{\textit{case, system, dirt, phase, surface tension, oil, liquid, surface-active agent, surface}\} \\
\delta^3(C) &= \{\textit{case, system, dirt, natural phenomenon, oil, liquid, surface-active agent, surface}\} \\
\delta^4(C) &= \{\textit{case, system, dirt, physical entity, oil, surface-active agent, surface}\} \\
\delta^5(C) &= \{\textit{case, system, dirt, physical entity, oil}\} \\
\delta^6(C) &= \{\textit{case, dirt, entity, oil}\} \\
\delta^7(C) &= \{\textit{case, dirt, entity}\} \\
\delta^8(C) &= \{\textit{case, entity}\} \\
\delta^9(C) &= \{\textit{entity}\}
\end{aligned}
$$

Compared to connectivity clustering, a hierarchical clustering based on the shortest path will preserve concepts deep in the ontology until late in the clustering. However, this is due to using the shortest path as a similarity measure rather than hierarchical clustering. Using shared nodes (Andreasen et al. 2005*a*), a measure based on the cardinality of the shared upper bounds, would, e.g. result in *dirt* and *case* being merged at an earlier step.

8.5 Discussion and Summary

The purpose of this chapter, which presented approaches to ontology-based conceptual summaries, is to provide a means for browsing the content of a set of documents

rather than documents directly. Connectivity clustering is clustering based solely on connectivity in the ontology and one approach to conceptual summaries where two siblings are iteratively replaced in the summary by their common ancestor. Hierarchical similarity clustering is an alternative that distinguishes itself from the former by relying on a semantic similarity measure in the grouping of concepts instead of the connectivity of the ontology. The notion of supported least upper bound is introduced as a means of summarizing a group of concepts by their least upper bound with an indication of how many concepts in the input set are covered by the group. Also, the notion of fuzzy least upper bounds is introduced as a means of expressing to what extent the upper bound of a set of concepts is characterized by "leastness" or "upperness". The purpose here being that a soft definition of least upper bound can capture how close to a set of concepts the least upper bound is. Finally, a preliminary experiment with SEMCOR is presented that illustrates the approaches and some of their differences.

In the hierarchical similarity-based approach, each step in the clustering process results in a cluster being replaced by the least upper bound of the participating concepts. The least upper bound is then used for succeeding similarity calculations. From a summary perspective, this is reasonable because at each application of δ, the summary should be in the form of a smaller set of concepts, easy to grasp and brief. However, this is merely a question of presenting the summary to the user. Thus, another option in the clustering process would be to preserve the original concepts in the clusters, perform similarity calculations based on this set, and then only present the least upper bound to the user. This opens up for a more versatile approach to measuring similarity between the elements as is done in a more classical clustering, for example (Manning et al. 2007):

Single link:	The similarity of the closest elements in two clusters
Complete link:	The similarity of the most distant elements in two clusters
Average link:	The average pairwise similarity of all the concepts in two clusters

The advantage of using a least upper bound approach with respect to complexity is lost, however, which for large scale clustering can be an important property of the hierarchical similarity-based clustering.

A compromise between keeping all the elements of clusters throughout the clustering process and replacing them by their least upper bounds could be to base the similarity calculation on the fuzzy least upper bounds. The problem is, though, that depending on the actual setting of the thresholds, the fuzzy least upper bounds risk containing more elements than the original concepts in the clusters. This is clearly a disadvantage with regard to the complexity of the clustering, but it also makes the summaries larger than the original set of concepts to be summarized. The solution to this problem could be to chose a few of the largest fuzzy least upper bounds and use these elements when measuring similarity and when presenting the summary to the user.

Finally, there is the question of *semantic outliers*. In machine learning, noise

is often regarded as irregular objects that do not fit the general patterns of the rest of objects, and which are evenly spread thinly across the entire spectrum of input. Outliers are typically single irregular objects that are situated far apart from the main groups of data. For both kinds of objects they are characterized by being few in numbers compared to the general distribution of data.

When creating a conceptual summary our aim is an easy to grasp, short description that also includes the important aspects of the concepts. Naturally irrelevant aspects like noise and outliers should be excluded first, but the question is what constitutes semantic noise and semantic outliers? At first it might seem obvious to base the definitions on the frequency of the concepts. If the frequency of a concept in a document is very low compared to other concepts in the document, it is a matter of secondary importance and could thus be excluded from the summary. If the concept is also situated far from the other concepts in the ontology, it could be characterized as a semantic outlier. But consider the set of concepts: {*cylinder block*, *cooler*, *sprinkler system*, *carburetor*, *car*}. Based on the above perspective of outliers, *car* can be viewed as an outlier in as much as it is situated far apart from the group of mechanical devices if inclusion only is considered. On the other hand, from an ontological and a summary perspective all but one concept are mechanical devices that are part of a car, and the summary could thus be the single compound concept *mechanical devices*[POF: *car*]. This indicates that the semantic outlier, *car*, by no means can be defined solely due to its position in the subsumption hierarchy in relation to the other concepts or simply due to its frequency.

To further underline the challenge of establishing a suitable definition for semantic outliers, consider the supposition that a query poses a special perspective on a result set. For instance, if we are looking for the causes of diabetes even concepts peripheral to others are important if they are in causal relation to diabetes or possibly related illnesses in the documents. One could also argue that a result set by itself presents a perspective on a document collection. But if documents are the basic retrieved textual unit, many concepts irrelevant to the query are thus also retrieved, which in turn can adversely affect the quality of the summary. A semantic outlier should thus not only be defined with regard to the result set or the outliers position in the ontology related to other concepts in the result set, but also with regard to the query. Further work on an appropriate definition of what constitutes a *semantic* outlier is clearly needed.

Chapter 9

Conclusions and Perspectives

This dissertation has sought to explore how statistical analysis of a document collection can be combined with ontological knowledge in information retrieval. The motivation for pursuing such an endeavor lies in the intriguing possibilities available for combining the two different frameworks. By using statistical means the specific document collection at hand can be analyzed and used for matching a user's information need. Ontologies, on the other hand, offer a world-knowledge-based semantic analysis simply not within the reach of a statistical analysis. This dissertation has shown a number of different paths that can be taken in pursuing a goal of combining statistics and ontologies in information retrieval. The first part of the dissertation described the foundations in the form of information retrieval, ontologies and the ONTOQUERY project.

Following the strain of content analysis from chapter four, chapter **five** presented a preliminary machine learning approach to the semantic analysis of prepositions. This approach is based on the assumption that in *NP-P-NP* constructs, there exists an affinity between the relation denoted by the preposition and the concepts appearing as heads of the noun phrases. Initially, a Danish corpus was compiled from within the area of nutrition with an annotation of the concepts denoted by the heads of the noun phrases and the relation type denoted by the preposition. Based on standard machine learning, it was shown that there is a clear affinity, and the experiments indicate that the type of concepts are more important in the determination of the relation than the lemmatized word form of the head or the preposition itself. An encouraging fact was that most of the statistical patterns within the corpus could be expressed with a relatively small set of rules, and thus the approach could, with relative ease, be applied in content extraction for information retrieval. As a result much more specific ontological descriptors can be generated in the indexing of both queries and documents, thus improving ontology-based content analysis.

In chapter **six** the focus shifted towards semantic and distributional measures of similarity. The reason for this focus is that though ontologies are important for identifying relations between concepts, there can be patterns of distributional similarity

between concepts that are specific to a given document collection and these patterns are not reasonable to model as relations within the ontology. Thus, two sources of information exist about the similarity between concepts, and this chapter presents three different approaches for combining them. The notion of distributional density is introduced as the distributional similarity of the pairwise combination of all the concepts situated closely in the ontology to the two concepts. Though distributional similarity and semantic similarity are recognized as being highly related it is a novel idea to combine the two measures in one. A single measure is required if both types of measures are to be used in e.g. fuzzy information retrieval using a thesaurus, as it was presented in chapter two, or in the model for index expansion in the vector space model introduced in chapter seven.

Chapter **seven** presented the approach to ontology-based index expansion in the vector space model. The fundamental idea of the proposed model is to enable indexing and matching to rely both on a lexical and a conceptual analysis of documents and queries. A generalized term frequency inverse document frequency, $tfidf$, is described that is based on term frequency and the frequency of related terms. Given the different relations between concepts in an ontology, the index of a term can be expanded to include related terms, for instance, given *insulin*, the subsumer *hormone* could be added to the index. In addition to the generalized $tfidf$ measure, an approach for a weighted combination of conceptual indexing with lexical indexing is proposed. Compared to previous research the model it is novel to use a measure of concept specificity in the expansion; though it is well established within information retrieval that a measure of specificity is of great importance for retrieval performance.

Finally, chapter **eight** presented how conceptual summaries through different kinds of clustering can serve as a mean of summarizing a set of concepts. Given an instantiated ontology, which is an ontology restricted to the concepts in the upward expansion of the concepts in a given set, connectivity clustering was the first approach suggested. In connectivity clustering the idea is to cluster a given set of concepts based on their connections to common ancestors, for instance, by grouping two siblings based on their common parent, and also by replacing the group with the common ancestor. In its pure form, connectivity clustering is based solely on the instantiated ontology but a prioritized connectivity clustering is also suggested that can refine the clustering process by giving priority to certain concepts, e.g. concepts with little support. In addition, hierarchical similarity clustering based on an ontological similarity measure was presented. Compared to connectivity clustering, hierarchical similarity clustering is not bound strictly to the ontological connections but naturally reflects the ontology due to the semantic similarity measure being applied in the clustering. In hierarchical similarity clustering, clusters are replaced by their common parent, i.e. their least upper bound, similar to connectivity clustering. A fuzzyfied least upper bound is, however, also introduced that results in a more robust hierarchical similarity clustering. Using an instantiated ontology in itself as summary is novel idea, and the proposed method will find their application in a situation where

an alternative to a natural language summary is desired.

In summary, the research presented in chapter five, six, seven and eight of this dissertation have among other things shown:

1. An approach to improving the specificity of ontology-based descriptors using machine learning and that only apply general knowledge about the concepts being analyzed.

2. Three approaches for combining semantic and distributional similarity measures in order to improve the similarity measures used in expansion.

3. An approach for ontology-based index expansion as an alternative to the term weighting applied in the Vector Space Model. The approach takes into account the specificity of the concepts in expansion. The model includes a generalized term weight for expanded terms, and a mean for emphasizing lexical or conceptual match.

4. Two general approaches for generating conceptual summaries based on an ontology with the inclusion of corpus statistics.

Thus, the four chapters demonstrates novel approaches to combining corpus statistics with ontologies in content analysis, measures of similarity, indexing and representation, and the presentation of a result set.

However, given the broad nature of the proposals, future work is naturally possible in several areas, some of which have already been presented in previous chapters, but two issues of a more general concern will be added here.

9.1 Further Work

The first issue to be treated here concerns the need for developing a test bed for ontology-based information retrieval as a vehicle for future research in the area. The second issue is concerned with the development of an approach for adjusting the similarity between concepts in the ontology by means of a form of relevance feedback.

9.1.1 A common test bed

A common challenge within the ontology-based information retrieval community is the establishment of a test bed for the contributions being made within the field. Given the complexity and scale of the task, it is unlikely that one can be established by a single research group; the TExt Retrieval Conference is a good example of the benefits that can be achieved by cooperating in order to establish document collections, queries, rankings, etc.

A test bed could serve as a vehicle for several interesting research issues, including the exploration of what type of queries, documents, and possibly domains

for which ontology-based information retrieval is a more fertile approach than traditional keyword-based information retrieval. Obviously, ontology-based information retrieval has something to offer especially with regard to, e.g. complex queries involving relations between concepts, and short documents with little term variance. Clearer insight into the general strengths of ontology-based information retrieval, however, would help direct research efforts within the field. There is also a need for experiments that do not compare ontology-based information retrieval to keyword-based retrieval, but that focus on the interaction between the different processes. Is there, for instance, a difference in the robustness of the different semantic similarity measures towards faulty word sense disambiguation? And, can the extraction of conceptual content from the text focus only on the concepts and the relations in the ontology be relied upon for measuring similarities? Answers to these and similar question would be much easier to give with a common test bed.

One of the challenges in the empirical validation of different contributions within the field is that a large number of them include highly parameterized models. There are some fundamental challenges in performing experiments with parameterized models. First, determining which parts of a model contribute, and to what degree, to the successful solution to the problem the model is intended to solve can be difficult. Second, at least in the natural sciences, the simplest explanation to a problem is usually preferred because there is the danger of highly parameterized models coincidentally fitting the data. The axiom behind the preference is sometimes termed Occam's razor (Mitchell 1997, Cover & Thomas 1991). Jiang & Conrath's similarity measure is a prime example of a highly parameterized model. Their shortest path approach is a scaling of each edge in the ontology by a parameterized factorization of the density of the graph (i.e. a modification of Sussna's (1993) fan-out-factor), relative depth scaling, the difference in information content of a subsumer and a subsumed, and, finally, a weighting of the relation type. Though empirically evaluated as one of the best measures of semantic similarity, the extent to which the various factors contribute to the success is unknown. Evidently, this would also be the case with the similarity measures proposed in chapter 6.

In general, a common test bed could be just the glue necessary to join together the most promising components of an ontology-based information retrieval system.

9.1.2 Learning similarities

One main thread in each chapter is semantic similarity measures. In the ontology-based vector space model, these measures are the basis of the expansion, while in the semantic analyses of prepositions, they are advanced as a natural improvement to the learning process, and they are the basis of the similarity measures suggested in chapter 6, and, finally, they are used in hierarchical clustering to determine the closeness of two clusters. In ontology-based information retrieval, similarity measures are thus only rivaled in importance by an accurate word sense disambiguation. Word sense

disambiguation can be studied in its own right independent of other processes and with clearly stated success criteria, but similarity measures are inherently difficult to evaluate directly because they are dependent on the human assessment of how similar concepts are, e.g. how similar is a *pizza* to a *croissant* on a scale from zero to one? Even when given several other concepts to compare, thereby making the scale relative, it is inherently difficult for humans to asses these kinds of similarities using a scale.

At this point, experiments comparing different semantic similarity measures have been done on very small data sets based on the human assessment of semantic similarity or on areas such as malapropisms (Budanitsky & Hirst 2006). To the best of our knowledge, no empirical comparisons of the different semantic similarity measures for document retrieval have been made. From an information retrieval perspective, it is namely partly irrelevant which measure is the best at measuring concept to concept similarities. Rather, we are concerned about the measure's ability to match queries and documents correctly at the more aggregate level. Naturally, the development of a test bed for ontology-based information retrieval could facilitate this type of comparison, where comparing each of the proposed similarities would be possible in equal settings. However, the performance of the different similarity measures may turn out to be dependent on the concrete ontology, the document collection and the type of queries.

Thus, the real question is how to design a semantic similarity measure that measures concept to concept similarity but is evaluated on the aggregate level of information retrieval, and possibly an approach that adjusts the measure according to the evaluation. One possible solution and future work could involve developing a learning framework where the structure of the ontology and the derived similarity are altered iteratively based on some kind of user feedback, which is, in principle, similar to backpropagation in neural networks, where an error in the output gets propagated backwards to the network and the weights on the edges between the nodes are updated. To visualize the idea, imagine the ontology as a mesh that is stretched and contracted locally depending on how similar the concepts in that part of the mesh should be considered. In especially the late 1980s and until the mid 1990s, there were several contributions made on the usage of neural networks in information retrieval (see e.g. Chen (1995), Wilkinson & Hingston (1991), Belew (1989)), but the focus was different in the sense that the objective was to learn the relation between query terms and documents. However, it is not certain that the task of learning semantic similarity should be cast within a connectionist learning framework. Moreover, research on the topic should certainly draw upon the recent attention from the machine learning community on learning graphs (see e.g. Frasconi et al. (2008) and the conferences MLG (2008), TextGraphs (2008), ICML (2008)). The document vector modification approach initially proposed by Brauen (1971) could also be adopted for the purpose of adjusting the similarity based on the index expansion model suggested in chapter 4. The approach proposed by Brauen is to adjust the term weights in the relevant

documents based on the terms appearing in the query, thereby moving relevant document vectors closer to the query vector. This altering of the weights could possibly be propagated further back to the similarity measure, and thus create a framework for learning "the right" similarity between concepts.

Bibliography

Andreasen, T. & Bulskov, H. (2007*a*), On browsing domain ontologies for information base content, *in* 'Proceedings IFSA 2007', Vol. 4529 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 135–144.

Andreasen, T. & Bulskov, H. (2007*b*), Query expansion by taxonomy, *in* J. Galindo, ed., 'Handbook of Research on Fuzzy Information Processing in Databases', Information Science Reference, an imprint of Idea Group Inc., chapter 13, pp. 325–351.

Andreasen, T., Bulskov, H. & Knappe, R. (2003), Similarity from conceptual relations, *in* E. Walker, ed., '22nd International Conference of the North American Fuzzy Information Processing Sociaty, NAFIPS 2003, Chicago, Illinois USA', IEEE, pp. 179–184.

Andreasen, T., Bulskov, H. & Knappe, R. (2005*a*), Domain-specific similarity and retrieval, *in* 'Proceedings IFSA 2005', pp. 496–502.

Andreasen, T., Bulskov, H. & Knappe, R. (2005*b*), On automatic modeling and use of domain-specific ontologies, *in* M.-S. Hacid, N. V. Murray, Z. W. Ras & S. Tsumoto, eds, 'Foundations of Intelligent Systems, 15th International Symposium, ISMIS 2005, Saratoga Springs, NY, USA, May 25-28, 2005, Proceedings', Vol. 3488 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 74–82.

Andreasen, T., Bulskov, H. & Terney, T. (2008), Ontological summaries through hierarchical clustering, *in* A. An, S. Matwin, Z. W. Ras & D. Slezak, eds, 'Foundations of Intelligent Systems, 17th International Symposium, ISMIS 2008, Toronto, Canada, May 20-23, 2008, Proceedings', Vol. 4994 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 497–507.

Andreasen, T., Jensen, P. A., Nilsson, J. F., Paggio, P., Pedersen, B. S. & Thomsen, H. E. (2002), Ontological extraction of content for text querying, *in* B. Andersson, M. Bergholtz & P. Johannesson, eds, 'Proceedings of the 6th International Conference on Applications of Natural Language to Information Systems', Vol. 2553 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 123 – 136.

Andreasen, T., Jensen, P. A., Nilsson, J. F., Paggio, P., Pedersen, B. S. & Thomsen, H. E. (2004), 'Content-based text querying with ontological descriptors', *Data & Knowledge Engineering* **48**(2), 199–219.

Andreasen, T. & Nilsson, J. F. (2004), 'Grammatical specification of domain ontologies.', *Data & Knowledge Engineering* **48**(2), 221–230.

Andreasen, T., Nilsson, J. F. & Thomsen, H. E. (2001), Introduction: The ontoquery project, *in* P. A. Jensen & P. Skadhauge, eds, 'Proceedings of the First International OntoQuery Workshop', Dept of Business Communication and Information Science. University of Southern Denmark., pp. 1–10.

Avrahami, J. & Kareev, Y. (1993), 'What do you expect to get when you ask for "a cup of coffee and a muffin or a croissant"?', *International Journal of Man-Machine Studies* **38**(3), 429–434.

Baader, F. & Nutt, W. (2003), Basic description logics, *in* F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi & P. F. Patel-Schneider, eds, 'The Description Logic Handbook: Theory, Implementation, and Applications', Cambridge University Press, New York, NY, USA, chapter 2, pp. 43–95.

Baeza-Yates, R. & Ribeiro-Neto, B. (1999), *Modern Information Retrieval*, Addison-Wesley.

Baldwin, T. (2006), Distributional similarity and preposition semantics, *in* P. Saint-Dizier, ed., 'Computational Linguistics Dimensions of Syntax and Semantics of Prepositions', Springer Verlag, chapter 13, pp. 197–210.

Belew, R. K. (1989), 'Adaptive information retrieval: using a connectionist representation to retrieve and learn about documents', *SIGIR Forum* **23**(SI), 11–20.

Bhogal, J., Macfarlane, A. & Smith, P. (2007), 'A review of ontology based query expansion', *Information Processing & Management* **43**(4), 866–886.

Bittner, T., Donnelly, M. & Smith, B. (2004), 'Endurants and perdurants in directly depicting ontologies', *AI Communications* **17**(4), 247–258.

Braasch, A., Olsen, S. & Pedersen, B. S. (1998), Large scale lexicon for danish in the information society, *in* A. Rubio, N. Gallardo & A. Tejada, eds, 'Proceedings from the First Conference on Language Resources and Evaluation. Granada.', pp. 249–255.

Brauen, T. L. (1971), Document vector modification, *in* G. Salton, ed., 'The SMART Retrieval System', Prentice-Hall, chapter 24, pp. 456–484.

Brink, C., Britz, K. & Schmidt, R. A. (1994), 'Peirce algebras', *Formal Aspects of Computing* **6**(3), 339–358.

Budanitsky, A. & Hirst, G. (2006), 'Evaluating wordnet-based measures of lexical semantic relatedness', *Computational Linguistics* **32**(1), 49–82.

Bulskov, H., Andreasen, T. & Terney, T. V. (2007), Conceptual summaries as query answers, *in* 'Annual Meeting of the North American Fuzzy Information Processing Society NAFIPS '07. Proceedings.', IEEE, pp. 458–462.

Bulskov, H., Andreasen, T. & Terney, T. V. (2008), Ontological summaries, *in* B. N. Madsen & H. E. Thomsen, eds, 'Managing Ontologies and Lexical Ressources : 8th International Conference on Terminology and Knowledge Engineering. TKE 2008', Litera, pp. 219–230.

Bulskov, H., Knappe, R. & Andreasen, T. (2002), On measuring similarity for conceptual querying, *in* J. G. Carbonell & J. Siekmann, eds, 'FQAS '02: Proceedings of the 5th International Conference on Flexible Query Answering Systems', Vol. 2522 of *Lecture Notes in Artificial Intelligence*, Springer Berlin / Heidelberg, London, UK, pp. 100–111.

Bulskov, H., Knappe, R. & Andreasen, T. (2004), On querying ontologies and databases., *in* H. Christiansen, M.-S. Hacid, T. Andreasen & H. L. Larsen, eds, 'LNAI 3055, 6th International Conference on Flexible Query Answering Systems, Lyon, France, Flexible Query Answering Systems (Lecture Notes in Computer Science 3055)', Springer Verlag, pp. 191–202.

Chen, H. (1995), 'Machine learning for information retrieval: Neural networks, symbolic learning, and genetic algorithms', *Journal of the American Society for Information Science* **46**(3), 194–216.

Church, K. W. & Hanks, P. (1990), 'Word association norms, mutual information, and lexicography', *Computational Linguistics* **6**, 22–29.

CLAWS (2008), 'Free claws www trial service.', Online trial version of the CLAWS tagger made available by University Centre for Computer Corpus Research on Language.

Cohen, W. W. (1995), Fast effective rule induction, *in* A. Prieditis & S. Russell, eds, 'Proceedings of the 12th International Conference on Machine Learning', Morgan Kaufmann, Tahoe City, CA, pp. 115–123.

Consortium, T. G. O. (2000), 'Gene ontology: tool for the unification of biology', *Nature Genetics* **25**, 25–29.

Cooper, W. S. (1997), 'Getting beyond boole', *Information Processing and Management* **24**, 265–267.

Cover, T. M. & Thomas, J. A. (1991), *Elements of information theory*, Wiley-Interscience, New York, NY, USA.

Cristianini, N. & Shawe-Taylor, J. (2000), *An introduction to support Vector Machines: and other kernel-based learning methods*, Cambridge University Press, New York, NY, USA.

Dagan, I. (2000), Contextual word similarity, *in* R. Dale, H. Moisl & H. Somers, eds, 'Handbook of Natural Language Processing', Marcel Dekker Inc., chapter 19, pp. 459–476.

Dagan, I., Lee, L. & Pereira, F. C. N. (1999), 'Similarity-based models of word cooccurrence probabilities', *Machine Learning* **34**(1-3), 43–69.

Davis, R., Shrobe, H. E. & Szolovits, P. (1993), 'What is a knowledge representation?', *AI Magazine* **14**(1), 17–33.

Dowty, D. (1991), 'Thematic proto-roles and argument selection', *Language* **67**, 547–619.

Efthimiadis, E. N. (1996), 'Query expansion', *Annual Review of Information Systems and Technology (ARIST)* **31**, 121–187.

Fellbaum, C. D. (1998*a*), Wordnet: An electronic lexical database (language, speech, and communication), *in* C. Fellbaum, ed., 'A Semantic Network of English Verbs', The MIT Press, chapter 3, pp. 69–104.

Fellbaum, C. D., ed. (1998*b*), *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*, The MIT Press.

Frasconi, P., Kersting, K., Toivonen, H. & Tsuda, K., eds (2008), *Special Topic of the Journal of Machine Learning Research: Mining and Learning with Graphs and Relations*, Microtome Publishing.

Gangemi, A., Guarino, N., Masolo, C. & Oltramari, A. (2001), Understanding top-level ontological distinctions, *in* A. G. Pérez, M. Gruninger, H. Stuckenschmidt & M. Uschold, eds, 'Proceedings of the 2001 IJCAI Workshop on Ontologies and Information Sharing', AAAI Press.

Gangemi, A., Guarino, N., Masolo, C. & Oltramari, A. (2003), 'Sweetening wordnet with dolce', *AI Mag.* **24**(3), 13–24.

Garside, R. & Smith, N. (1997), A hybrid grammatical tagger: Claws4, *in* R. Garside, G. Leech & T. Mcenery, eds, 'Corpus Annotation: Linguistic Information from Computer Text Corpora', Addison Wesley Longman, pp. 102–121.

Gómez-Pérez, A., Fernández-López, M. & Corcho, O. (2004), *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, Springer, Heidelberg.

Gonzalo, J., Verdejo, F., Chugur, I. & Cigarrán, J. M. (1998), Indexing with WordNet synsets can improve text retrieval, *in* S. Harabagiu, ed., 'Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems', Université de Montréal.

Greenberg, J. (2001*a*), 'Automatic query expansion via lexical-semantic relationships', *Journal of the American Society for Information Science and Technology* **52**(5), 402–415.

Greenberg, J. (2001*b*), 'Optimal query expansion (qe) processing methods with semantically encoded structured thesauri terminology', *Journal of the American Society for Information Science and Technology* **52**(6), 487–498.

Grefenstette, G. (1994*a*), Corpus-derived first, second and third-order word affinities, *in* 'EURALEX 1994. EURALEX International Congress on Lexicography (6th 1994 Amsterdam)'. Paper presented.

Grefenstette, G. (1994*b*), *Explorations in Automatic Thesaurus Discovery*, Kluwer Academic Publishers, Norwell, MA, USA.

Grenon, P. & Smith, B. (2004), 'SNAP and SPAN: Towards dynamic spatial ontology', *Spatial Cognition & Computation* **4**, 69–104.

Guarino, N. & Giaretta, P. (1995), Ontologies and knowledge bases: Towards a terminological clarification, *in* N. J. I. Mars, ed., 'Ontologies and Knowledge Bases: Towards a Terminological Clarification', IOS Press, chapter 3, pp. 25–32.

Guarino, N. & Welty, C. A. (2000), A formal ontology of properties, *in* R. Dieng & O. Corby, eds, 'EKAW '00: Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management', Vol. 1937 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 191–230.

Hansen, D. H. (2005), 'Compiled nutrition corpus from the danish national encyclopedia with pos tagging and ontological annotation', Centre for Language Technology.

Harris, Z. (1968), *Mathematical Structures of Language*, John Wiley & Sons.

Hindle, D. (1990), Noun classification from predicate-argument structures, *in* 'Proceedings of the 28th annual meeting on Association for Computational Linguistics', Association for Computational Linguistics, Morristown, NJ, USA, pp. 268–275.

Hliaoutakis, A., Varelas, G., Voutsakis, E., Petrakis, E. G. & Milio, E. (2006), 'Information retrieval by semantic similarity', *International Journal on Semantic Web and Information Systems, Special Issue of Multimedia Semantics* **3**(3), 55–73.

Hotho, A. (2005), Using ontologies to improve the text custering and classification task, *in* N. Kushmerick, F. Ciravegna, A. Doan, C. Knoblock & S. Staab, eds, 'Dagstuhl Seminar Machine Learning for the Semantic Web. Proceedings'.

Hotho, A., Maedche, A. & Staab, S. (2002), 'Ontology-based text document clustering.', *KI* **16**(4), 48–54.

Hotho, A., Staab, S. & Stumme, G. (2003), Wordnet improves text document clustering, *in* 'Proceedings of the Semantic Web Workshop at SIGIR-2003, 26th Annual International ACM SIGIR Conference'.

Hotho, A. & Stumme, G. (2002), Conceptual clustering of text clusters, *in* 'Proceedings of FGML Workshop', Special Interest Group of German Informatics Society (FGML — Fachgruppe Maschinelles Lernen der GI e.V.), pp. 37–45.

ICML (2008), 'Prior knowledge for text and language processing', Workshop at The 25th International Conference on Machine Learning.

Ide, N. & Véronis, J. (1998), 'Special issue on word sense disambiguation: Introduction to the special issue on word sense disambiguation: the state of the art', *Computational Linguistics* **24**(1), 2–40.

Ingwersen, P. (1992), *Information Retrieval Interaction.*, Taylor Graham.

Jacquemin, C. & Bourigault, D. (2003), Term extraction and automatic indexing, *in* R. Mitkov, ed., 'Handbook of computational linguistics', Oxford University Press, chapter 33, pp. 599–615.

Jacquemin, C. & Tzoukermann, E. (1999), Nlp for term variant extraction: A synergy of morphology, lexicon and syntax., *in* T. Strzalkowski, ed., 'Natural Language Information Retrieval', Kluwer Academic Publishers, chapter 2, pp. 25–74.

Jansen, B. J. & Pooch, U. (2001), 'A review of web searching studies and a framework for future research', *Journal of the American Society for Information Science and Technology* **52**(3), 235–246.

Jensen, P. A. & Nilsson, J. F. (2003), Ontology-based semantics for prepositions, *in* P. Saint-Dizier, ed., 'Syntax and Semantics of Prepositions', Springer Verlag, chapter 15, pp. 229–244.

Jensen, P. A., Nilsson, J. F. & Vikner, C. (2001), Towards an ontology-based interpretation of noun phrases, *in* P. A. Jensen & P. Skadhauge, eds, 'Proceedings of the First International OntoQuery Workshop.', Dept of Business Communication and Information Science. University of Southern Denmark., pp. 43–56.

Jiang, J. J. & Conrath, D. W. (1997), Semantic similarity based on corpus statistics and lexical taxonomy, *in* 'The 10th International Conference on Research in Computational Linguistics, ROCLING X. Proceedings.'.

Jing, L., Zhou, L., Ng, M. K. & Huang, J. Z. (2006), Ontology-based distance measure for text clustering, *in* M. W. Berry & M. Castellanos, eds, 'Proceedings of the Fourth Workshop on Text Mining. Sixth SIAM International Conference on Data Mining'.

Jones, K. S. (1972), 'A statistical interpretation of term specificity and its application in retrieval', *Journal of Documentation* **28**(1), 11–21.

Jurafsky, D. & Martin, J. H. (2000), *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice-Hall.

Keerthi, S. S., Shevade, S. K., Bhattacharyya, C. & Murthy, K. R. K. (2001), 'Improvements to Platt's SMO Algorithm for SVM Classifier Design', *Neural Computation* **13**(3), 637–649.

Klir, G. J. & Yuan, B. (1995), *Fuzzy sets and fuzzy logic: theory and applications*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Knappe, R., Bulskov, H. & Andreasen, T. (2007), 'Perspectives on ontology-based querying', *International Journal of Intelligent Systems* **22**(7), 739–761.

Kraft, D. H., Bordogna, G. & Pasi, G. (1999), Fuzzy sets techniques in information retrieval, *in* J. Bezdek, D. Dubois & H. Prade, eds, 'Fuzzy Sets in Approximate Reasoning and Information Systems.', The Handbooks of Fuzzy Sets, Springer, chapter 8, pp. 469–502.

Krauthammer, M. & Nenadic, G. (2004), 'Term identification in the biomedical literature', *Journal of Biomedical Informatics* **37**(6), 512–526.

Kristensen, J. (1993), 'Expanding end-users query statements for free text searching with a search-aid thesaurus', *Information Processing & Management* **29**(6), 733–744.

Lancaster, F. W. (1968), *Information Retrieval Systems: Characteristics, Testing, and Evaluation*, Wiley.

Lassen, T. (2007), Noter om præpositioner og parafrasetests. Unpublished note which can be obtained from the author.

Lassen, T. & Terney, T. V. (2006*a*), An ontology-based approach to disambiguation of semantic relations, *in* R. Basili & A. Moschitti, eds, 'Proceedings of the

Workshop on Learning Structured Information In Natural Language Applications, EACL 2006', European Chapter of the Association for Computational Linguistics (EACL).

Lassen, T. & Terney, T. V. (2006*b*), Ontology-based disambiguation of the semantic relation between the heads of two noun phrases, *in* G. Sutcliffe & R. Goebel, eds, 'Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference', The AAAI Press, Menlo Park, California., pp. 791–793.

Lassila, O. & McGuinness, D. (2001), The role of frame-based representation on the semantic web, Technical report, Knowledge Systems Laboratory, Stanford University.

Lau, T. & Sure, Y. (2002), Introducing ontology-based skills management at a language insurance company, *in* M. Glinz & G. Müller-Luschnat, eds, 'LNI. Modellierung in der Praxis - Modellierung für die Praxis', Vol. 12, GI, pp. 123–134.

Leacock, C. & Chodorow, M. (1998), Combining local context and wordnet similarity for word sense identification, *in* C. Fellbaum, ed., 'WordNet: An electronic lexical database and some of its applications', The MIT Press., Cambrige, MA.

Lee, C.-S., Jian, Z.-W. & Huang, L.-K. (2005), 'A fuzzy ontology and its application to news summarization.', *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics* **35**(5), 859–880.

Lee, J. H., Kim, M. H. & Le, Y. J. (1993), 'Information retrieval based on conceptual distance in is-a hierarchies', *Journal of Documentation* **49**(2), 188–207.

Lenat, D. B. (1995), 'Cyc: a large-scale investment in knowledge infrastructure', *Communications of the ACM* **38**(11), 33–38.

Lenci, A., Bel, N., Busa, F., Calzolari, N., Gola, E., Monachini, M., Ogonowski, A., Peters, I., Peters, W., Ruimy, N., Villegas, M. & Zampolli, A. (2000), 'Simple: A general framework for the development of multilingual lexicons.', *International Journal of Lexicography* **13**(4), 249–263.

Lenci, A., Busam, F., Ruimy, N., Gola, E., Monachini, M., Calzolari, N. & Zampolli, A. (2000), Linguistic specifications deliverable d2.1, Technical report, University of Pisa and Institute of Computational Linguistics of CNR, Pisa.

Li, Z. & Ramani, K. (2007), 'Ontology-based design information extraction and retrieval', *AI EDAM: Artificial Intelligence for Engineering Design, Analysis, and Manufacturing* **21**(2), 137–154.

Litkowski, K. (2004), Senseval-3 task: Automatic labeling of semantic roles, *in* R. Mihalcea & P. Edmonds, eds, 'Proceedings of Senseval-3: The Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text', Association for Computational Linguistics, Barcelona, Spain, pp. 9–12.

López, M. F. & Pérez, A. G. (2002), 'The integration of OntoClean in WebODE', Paper presented at the EKAW2002 Workshop on Evaluation of Ontology-Based Tools (EON2002).

Luhn, H. P. (1958), 'The automatic creation of literature abstracts', *IBM Journal of Research and Development* **2**(2), 159–165.

Lund, J., ed. (1994), *Den Store Danske Encyklopædi*, Gyldendal.

Maedche, A. & Staab, S. (2004), Ontology learning, *in* S. Staab & R. Studer, eds, 'Handbook on Ontologies', Springer, chapter 9, pp. 173–190.

Manning, C. D., Raghavan, P. & Schütze, H. (2007), *Introduction to Information Retrieval*, Cambridge University Press.

Manning, C. D. & Schütze, H. (2003), *Foundations of Statistical Natural Language Processing*, The MIT Press, Cambridge, Massachusetts.

Markey, K. (2007), 'Twenty-five years of end-user searching, part 1: Research findings', *Journal of the American Society for Information Science and Technology* **58**(8), 1071–1081.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A. & Schneider, L. (2003), Wonderweb deliverable d17, Technical report, ISTC-CNR.

Meadow, C. T., Boyce, B. R., Kraft, D. H. & Barry, C. (2007), *Text information retrieval systems*, 3 edn, Emerald Group Publishing.

Mendes, S. (2006), Adjectives in wordnet.pt, *in* P. Sojka, K.-S. Choi, C. Fellbaum & P. Vossen, eds, 'Proceedings of the Third International WordNet Conference, GWC 2006.', Masaryk University, Brno, pp. 225–230.

Mihalcea, R. & Moldovan, D. (2000), Semantic indexing using WordNet senses, *in* 'Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval', Association for Computational Linguistics, Morristown, NJ, USA, pp. 35–45.

Miller, G. A. (1995), 'Wordnet: a lexical database for english', *Commun. ACM* **38**(11), 39–41.

Miller, G. A. (1998), Nouns in WordNet, *in* C. Fellbaum, ed., 'WordNet: An Electronic Lexical Database', MIT press, chapter 1, pp. 24–46.

Miller, G. A., Chodorow, M., Landes, S., Leacock, C. & Thomas, R. G. (1994), Using a semantic concordance for sense identification, *in* 'Human Language Technology, Proceedings of a Workshop held at Plainsboro, New Jerey, USA, March 8-11, 1994', Association for Computational Linguistics, Morristown, NJ, USA, pp. 240–243.

Mitchell, T. M. (1997), *Machine Learning*, McGraw-Hill, New York.

Miyamoto, S. (1990), *Fuzzy Sets in Information Retrieval and Cluster Analysis*, Springer.

MLG (2008), '6th international workshop on mining and learning with graphs'.

Mohammad, S. & Hirst, G. (2005), 'Distributional measures as proxies for semantic relatedness', Obtained from Graeme Hirst.

Nagypal, G. (2005), Improving information retrieval effectiveness by using domain knowledge stored in ontologies, *in* R. Meersman, Z. Tari & P. Herrero, eds, 'On the Move to Meaningful Internet Systems 2005: OTM Workshops', Vol. 3762 of *Lecture Notes in Computer Science.*, Springer Berlin / Heidelberg, pp. 780–789.

Nardi, D. & Brachman, R. J. (2002), An Introduction to Description Logics, *in* F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi & P. F. Patel-Schneider, eds, 'The description logic handbook: theory, implementation, and applications', Cambridge University Press, chapter 1, pp. 5–44.

NASA (2007), 'Hubble's top achievements', web page. http://www.nasa.gov/externalflash/hubble_gallery/index_noaccess.html.

Nilsson, J. F. (2001), A logico-algebraic framework for ontologies, ontolog, *in* P. anker Jensen & P. Skadhauge, eds, 'Proceedings of the First International Onto-Query Workshop.', University of Southern Denmark, Kolding.

Nirenburg, S. & Raskin, V. (2004), *Ontological Semantics*, The MIT press.

NIST (2007), 'Text REtrieval conference', http://trec.nist.gov/.

Partee, B. H., ter Meulen, A. & Wal, R. E. (1990), *Mathematical Methods in Linguistics*, Springer.

Pasi, G. (2008), Fuzzy sets in information retrieval: State of the art and research trends, *in* H. Bustince, F. Herrera & J. Montero, eds, 'Fuzzy Sets and Their Extensions: Representation, Aggregation and Models', Springer, pp. 517–535.

Pedersen, B. S. (1999), Den danske simple-ordbog. en semantisk, ontologibaseret ordbog, *in* C. Poulsen, ed., 'DALF 99, Datalingvistisk Forenings årsmøde 1999', Center for sprogteknologi.

Peters, I. & Peters, W. (2000), The treatment of adjectives in simple: Theoretical observations, *in* 'Proceedings of LREC 2000', European Language Resources Association (ELRA).

Pustejovsky, J. (1991), 'The generative lexicon', *Comput. Linguist.* **17**(4), 409–441.

Pustejovsky, J. (1995), *The generative lexicon*, The MIT Press.

Rada, R., Mili, H., Bicknell, E. & Blettner, M. (1989), 'Development and application of a metric on semantic nets', *IEEE Transactions on Systems, Man and Cybernetics* **19**(1), 17–30.

Recupero, D. R. (2007), 'A new unsupervised method for document clustering by using wordnet lexical and conceptual relations', *Information Retrieval* **10**(6), 563–579.

Resnik, P. (1995), Using information content to evaluate semantic similarity in a taxonomy, *in* 'Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95, Montréal, Québec, Canada', Vol. 1, Morgan Kaufmann, pp. 448–453.

Resnik, P. (1999), 'Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language', *Journal of Artificial Intelligence Research* **11**, 95–130.

Robertson, S. E. & Jones, K. S. (1976), 'Relevance weighting of search terms', *Journal of the American Society for Information Science* **27**(3), 129–146.

Salton, G. & Buckley, C. (1988), 'Term-weighting approaches in automatic text retrieval.', *Information Processing and Management* **24**(5), 513–523.

Salton, G. & Buckley, C. (1990), 'Improving retrieval performance by relevance feedback.', *Journal of the American Society for Information Science, JASIS* **41**(4), 288–297.

Salton, G., Buckley, C. & Yu, C. T. (1982), An evaluation of term dependence models in information retrieval, *in* 'SIGIR '82: Proceedings of the 5th annual ACM conference on Research and development in information retrieval', Springer-Verlag New York, Inc., New York, NY, USA, pp. 151–173.

Salton, G. & Lesk, M. E. (1968), 'Computer evaluation of indexing and text processing', *Journal of the ACM* **15**(1), 8–36.

Salton, G. & McGill, M. (1982), *Introduction to Modern Information Retrieval*, McGraw Hill Higher Education.

Salton, G., Wong, A. & Yang, C. S. (1975), 'A vector space model for automatic indexing', *Communications of the ACM* **18**(11), 613–620.

Salton, G. & Yang, C. S. (1973), 'On the specification of term values in automatic indexing', *Journal of Documentation* **29**(4), 351–372.

Sowa, J. F. (2000), *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Course Technology.

Studer, R., Benjamins, R. & Fensel, D. (1998), 'Knowledge engineering: Principles and methods', *Data & Knowledge Engineering* **25**(1-2), 161–198.

Sussna, M. (1993), Word sense disambiguation for free-text indexing using a massive semantic network, *in* B. K. Bhargava, T. W. Finin & Y. Yesha, eds, 'CIKM '93: Proceedings of the second international conference on Information and knowledge management', Vol. 1, ACM Press, New York, NY, USA, pp. 67–74.

Terney, T. V. (2007), On combining semantic and distributional similarity measures, *in* H. R. Arabnia, M. Q. Yang & J. Y. Yang, eds, 'Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007', Vol. 1, CSREA Press, pp. 313–318.

Terra, E. & Clarke, C. L. A. (2003), Frequency estimates for statistical word similarity measures, *in* 'NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology', Vol. 1, Association for Computational Linguistics, Morristown, NJ, USA, pp. 165–172.

TextGraphs (2008), 'Textgraphs-3 graph-based algorithms for natural language processing', Conference.

Topi, H. & Lucas, W. (2005), 'Searching the web: operator assistance required', *Information Processing & Management* **41**(2), 383–403.

TPP (2007), 'The preposition project', http://www.clres.com/prepositions.html.

Tudhope, D., Binding, C., Blocks, D. & Cunliffe, D. (2006), 'Query expansion via conceptual distance in thesaurus indexed collections', *Journal of Documentation* **62**(4), 509–533.

U.S. National Library of Medicine (2007), 'Unified medical language system', Made avaibale by the U.S. National Library of Medicine.

Vallet, D., Fernández, M. & Castells, P. (2005), An ontology-based information retrieval model, *in* C. Bussler, J. Davies, D. Fensel & R. Studer, eds, 'Lecture Notes in Computer Science. The Semantic Web: Research and Applications', Vol. 3532, Springer, pp. 455–470.

Voorhees, E. M. (1994), Query expansion using lexical-semantic relations, *in* W. B. Croft & C. J. van Rijsbergen, eds, 'SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval', ACM/Springer, New York, NY, USA, pp. 61–69.

Voorhees, E. M. (1999), Natural language processing and information retrieval, *in* M. T. Pazienza, ed., 'Information Extraction: Towards Scalable, Adaptable Systems', Vol. 1714 of *Lecture Notes in Computer Science.*, Springer Berlin/Heidelberg, pp. 32–48.

Weeds, J. (2003), Measures and Applications of Lexical Distributional Similarity, PhD thesis, Department of Informatics, University of Sussex.

Weeds, J. & Weir, D. (2005), 'Co-occurrence retrieval: A flexible framework for lexical distributional similarity', *Computational Linguistics* **31**(4), 439–475.

Wilkinson, R. & Hingston, P. (1991), Using the cosine measure in a neural network for document retrieval, *in* 'SIGIR '91: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval', ACM, New York, NY, USA, pp. 202–210.

WordNet (2009), 'Wordnet', Electronic Resource from Cognitive Science Laboratory at Princeton University.

Wu, C.-W. & Liu, C.-L. (2003), Ontology-based text summarization for business news articles, *in* N. C. Debnath, ed., 'Proceedings of the ISCA 18th International Conference Computers and Their Applications, Honolulu, Hawaii, USA', ISCA, pp. 389–392.

Wu, Z. & Palmer, M. (1994), Verbs semantics and lexical selection, *in* 'Proceedings of the 32nd annual meeting on Association for Computational Linguistics', Association for Computational Linguistics, Morristown, NJ, USA, pp. 133–138.

Yager, R. R. & Petry, F. E. (2006), 'A multicriteria approach to data summarization using concept ontologies', *IEEE Transactions on Fuzzy Systems* **14**(6), 767–780.

Zhang, X., Jing, L., Hu, X., Ng, M. & Zhou, X. (2008), 'A comparative study of ontology based term similarity measures on pubmed document clustering', *Advances in Databases: Concepts, Systems and Applications* **4443**, 115–126.

Zhou, X., Hu, X., Lin, X., Han, H. & Zhang, X. (2006), Relation-based document retrieval for biomedical literature databases, *in* S. Istrail, P. Pevzner & M.Waterman, eds, 'Database Systems for Advanced Applications.', Vol. 3882 of *Lecture Notes in Computer Science.*, Springer Berlin / Heidelberg, pp. 689–701.

RECENT RESEARCH REPORTS

#125 Jan Midtgaard and David Van Horn. Subcubic control flow analysis algorithms. 32 pp. May 2009, Roskilde University, Roskilde, Denmark.

#124 Torben Braüner. Hybrid logic and its proof-theory. 318 pp. March 2009, Roskilde University, Roskilde, Denmark.

#123 Magnus Nilsson. *Arbejdet i hjemmeplejen: Et etnometodologisk studie af IT-støttet samarbejde i den københavnske hjemmepleje*. PhD thesis, Roskilde, Denmark, August 2008.

#122 Jørgen Villadsen and Henning Christiansen, editors. *Proceedings of the 5th International Workshop on Constraints and Language Processing (CSLP 2008)*, Roskilde, Denmark, May 2008.

#121 Ben Schouten and Niels Christian Juul, editors. *Proceedings of the First European Workshop on Biometrics and Identity Management (BIOID 2008)*, Roskilde, Denmark, April 2008.

#120 Peter Danholt. *Interacting Bodies: Posthuman Enactments of the Problem of Diabetes Relating Science, Technology and Society-studies, User-Centered Design and Diabetes Practices*. PhD thesis, Roskilde, Denmark, February 2008.

#119 Alexandre Alapetite. *On speech recognition during anaesthesia*. PhD thesis, Roskilde, Denmark, November 2007.

#118 Paolo Bouquet, editor. *CONTEXT'07 Doctoral Consortium Proceedings*, Roskilde, Denmark, October 2007.

#117 Kim S. Henriksen. *A Logic Programming Based Approach to Applying Abstract Interpretation to Embedded Software*. PhD thesis, Roskilde, Denmark, October 2007.

#116 Marco Baroni, Alessandro Lenci, and Magnus Sahlgren, editors. *Proceedings of the 2007 Workshop on Contextual Information in Semantic Space Models: Beyond Words and Documents*, Roskilde, Denmark, August 2007.

#115 Paolo Bouquet, Jérôme Euzenat, Chiara Ghidini, Deborah L. McGuinness, Valeria de Paiva, Luciano Serafini, Pavel Shvaiko, and Holger Wache, editors. *Proceedings of the 2007 workshop on Contexts and Ontologies Representation and Reasoning (C&O:RR-2007)*, Roskilde, Denmark, August 2007.

#114 Bich-Liên Doan, Joemon Jose, and Massimo Melucci, editors. *Proceedings of the 2nd International Workshop on Context-Based Information Retrieval*, Roskilde, Denmark, August 2007.