From Distributional to Semantic Similarity

James Richard Curran

Doctor of Philosophy

Institute for Communicating and Collaborative Systems
School of Informatics
University of Edinburgh

2003

Abstract

Lexical-semantic resources, including thesauri and WORDNET, have been successfully incorporated into a wide range of applications in Natural Language Processing. However they are very difficult and expensive to create and maintain, and their usefulness has been severely hampered by their limited coverage, bias and inconsistency. Automated and semi-automated methods for developing such resources are therefore crucial for further resource development and improved application performance.

Systems that extract thesauri often identify similar words using the *distributional hypothesis* that *similar words appear in similar contexts*. This approach involves using corpora to examine the contexts each word appears in and then calculating the similarity between context distributions. Different definitions of context can be used, and I begin by examining how different types of extracted context influence similarity.

To be of most benefit these systems must be capable of finding synonyms for rare words. Reliable context counts for rare events can only be extracted from vast collections of text. In this dissertation I describe how to extract contexts from a corpus of over 2 billion words. I describe techniques for processing text on this scale and examine the trade-off between context accuracy, information content and quantity of text analysed.

Distributional similarity is at best an approximation to semantic similarity. I develop improved approximations motivated by the intuition that some events in the context distribution are more indicative of meaning than others. For instance, the object-of-verb context `wear` is far more indicative of a clothing noun than `get`. However, existing distributional techniques do not effectively utilise this information. The new context-weighted similarity metric I propose in this dissertation significantly outperforms every distributional similarity metric described in the literature.

Nearest-neighbour similarity algorithms scale poorly with vocabulary and context vector size. To overcome this problem I introduce a new context-weighted approximation algorithm with bounded complexity in context vector size that significantly reduces the system runtime with only a minor performance penalty. I also describe a parallelized version of the system that runs on a Beowulf cluster for the 2 billion word experiments.

To evaluate the context-weighted similarity measure I compare ranked similarity lists against gold-standard resources using precision and recall-based measures from Information Retrieval,

since the alternative, application-based evaluation, can often be influenced by distributional as well as semantic similarity. I also perform a detailed analysis of the final results using WORDNET.

Finally, I apply my similarity metric to the task of assigning words to WORDNET semantic categories. I demonstrate that this new approach outperforms existing methods and overcomes some of their weaknesses.

Acknowledgements

I would like to thank my supervisors Marc Moens and Steve Finch. Discussions with Marc have been particularly enjoyable affairs, and the greatest regret of my time in Edinburgh is neither of us saw fit to schedule more of them.

Thanks to Ewan Klein and Ted Briscoe for reading a dissertation I guaranteed them would be short at such short notice in so short a time.

John Carroll very graciously provided the RASP BNC dependencies used in Chapter 3, Massimiliano Ciaramita providing his supersense data used in Chapter 6 and Robert Curran kindly typed in 300 entries from the New Oxford Thesaurus of English for the evaluation described in Chapter 2. Gregory Grefenstette and Lillian Lee both lent insight into their respective similarity measures. Thank you all for your help.

Edinburgh has been an exceptionally intellectually fertile environment to undertake a PhD and I appreciate the many courses, reading groups, discussions and collaborations I have been involved in over the last three years. In particular, Stephen Clark, Frank Keller, Mirella Lapata, Miles Osborne, Mark Steedman and Bonnie Webber have inspired me with feedback on the work presented in this thesis. Edinburgh has the kind of buzz I would like to emulate for my students in the future.

Along the way I have enjoyed the company of many fellow postgraduate travellers especially Naomei Cathcart, Mary Ellen Foster, Alastair Gill, Julia Hockenmaier, Alex McCauley, Kaska Porayska-Pomsta and Caroline Sporleder. Alastair took the dubious honour of being the sole butt of my ever deteriorating sarcasm with typical good humour. I am just sorry that Sydney is so far away from you all and I am so hopeless at answering email.

I remember fondly the times when a large Edinburgh posse went to conferences, in particular to my room mates in fancy (and not so fancy) hotels David Schlangen and Stephen Clark, and my conference ‘buddy’ Malvina Nissim. You guys make conferences a blast. I will also remember the manic times spent in collaboration with Stephen Clark, Miles Osborne and the TREC Question Answering and Biological Text Mining teams especially Johan Bos, Jochen Leidner and Tiphaine Dalmas. May our system performance one day reach multiple figures.

A statistical analysis of these acknowledgements would indicate that Stephen Clark has made by far the largest contribution to my PhD experience. Steve has been a fantastic friend and inspirational co-conspirator on a ‘wide-range of diverse and overlapping’ [sic] projects many

of which do not even get a mention in this thesis. It is the depth and breadth of his intuition and knowledge of statistical NLP that I have attempted to acquire on long flights to conferences and even longer car journeys to Glasgow. Hindering our easy collaboration is the greatest cost of leaving Edinburgh and I will miss our regular conversations dearly.

Attempting to adequately thank my parents, Peter and Marylyn, seems futile. Their boundless support and encouragement for everything on the path to this point, and their uncountable sacrifices to ensure my success and happiness, is appreciated more than the dedication can possibly communicate. That Kathryn, Robert and Elizabeth have forgone the option of excommunicating their extremely geeky brother whilst in Edinburgh is also appreciated. Returning home to be with you all is the only possible competition for Edinburgh.

Even after working for three years on techniques to automatically extract synonyms, I am still lost for words to describe Tara, my companion and collaborator in everything. Without you, I would not have undertaken our Edinburgh adventure and without you I could not have enjoyed it in the way we did: chatting and debating, cycling, cooking, art-house cinema, travelling, Edinburgh festivals, more chatting, snooker, backgammon and our ever growing book collection. I think of us completing two PhDs together rather than doing one each. You are constantly challenging the way I think about and experience the world; and are truly the most captivating, challenging and inspiring person I have ever met.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*James Richard Curran*)

To my parents, this is the culmination of every opportunity you have ever given me

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

> **introduction**: **launch** 0.052, **implementation** 0.046, advent 0.046, addition 0.045, adoption 0.041, arrival 0.038, absence 0.036, inclusion 0.036, creation 0.036, departure 0.036, availability 0.035, elimination 0.035, emergence 0.035, use 0.034, acceptance 0.033, abolition 0.033, array 0.033, passage 0.033, completion 0.032, announcement 0.032, ...

Natural Language Processing (NLP) aims to develop computational techniques for understanding and manipulating natural language. This goal is interesting from both scientific and engineering standpoints: NLP techniques inspire new theories of human language processing while simultaneously addressing the growing problem of managing information overload. Already NLP is considered crucial for exploiting textual information in expanding scientific domains such as bioinformatics (Hirschman et al., 2002). However, the quantity of information available to non-specialists in electronic form is equally staggering.

This thesis investigates a computational approach to *lexical semantics*, the study of word meaning (Cruse, 1986) which is a fundamental component of advanced techniques for retrieving, filtering and summarising textual information. It is concerned with statistical approaches to measuring *synonymy* or *semantic similarity* between words using raw text. I present a detailed analysis of existing methods for computing semantic similarity. This leads to new insights that emphasise semantic rather than distributional aspects of similarity, resulting in significantly improvements over the state-of-the-art. I describe novel techniques that make this approach computationally feasible and scalable to huge text collections. I conclude by employing these techniques to outperform the state-of-the-art in an application of lexical semantics. The semantic similarity example quoted above has been calculated using 2 billion words of raw text.

1.1 Contribution

Chapter 1 begins by placing *semantic similarity* in the context of the theoretical and practical problems of defining synonymy and other lexical-semantic relations. It introduces the manually constructed resources that have heavily influenced NLP research and reviews the wide range of applications of these resources. This leads to a discussion of the difficulties of manual resource development that motivate computational approaches to semantic similarity. The chapter concludes with an overview of the context-space model of semantic similarity which forms the basis of this work.

Chapter 2 surveys the many existing evaluation techniques for semantic similarity and motivates my proposed experimental methodology which is employed throughout the remainder of the thesis. This chapter concludes by introducing the detailed error analysis which is applied to the large-scale results in Chapter 6. This unified experimental framework allows the systematic exploration of existing and new approaches to semantic similarity.

I begin by decomposing the similarity calculation into the three independent components described in Section 1.8: *context*, *similarity* and *methods*. For each of these components, I have exhaustively collected and reimplemented the approaches described in the literature. This work represents the first systematic comparison of such a wide range of similarity measures under consistent conditions and evaluation methodology.

Chapter 3 analyses several different definitions of context and their practical implementation, from scientific and engineering viewpoints. It demonstrates that simple shallow methods can perform almost as well as far more sophisticated approaches and that semantic similarity continues to improve with increasing corpus size. Given this, I argue that shallow methods are superior for this task because they can process much larger volumes of text than is feasible for more complex approaches. This work has been published as Curran and Moens (2002b).

Chapter 4 uses the best context results from the previous chapter to compare the performance of many of the similarity measures described in the literature. Using the intuition that the most informative contextual information is collocational in nature, I explain the performance of the best existing approaches and develop new similarity measures which significantly outperform all the existing measures in the evaluation. The best combination of parameters in this chapter form the *similarity system* which is used for the remaining experimental results. This work has been published as Curran and Moens (2002a).

Chapter 5 proposes an ensemble approach to further improve the performance of the similarity system. This work has been published as Curran (2002). It also considers the efficiency of the naïve nearest-neighbour algorithm, which is not feasible for even moderately large vocabularies. I have designed a new approximation algorithm to resolve this problem which constrains the asymptotic complexity, significantly reducing the running time of the system, with only a minor performance penalty. This work has been published in Curran and Moens (2002a). Finally, it describes a message-passing implementation which makes it possible to perform experiments on a huge corpus of shallow-parsed text.

Chapter 6 concludes the experiments by providing a detailed analysis of the output of the similarity system, using a larger test set calculated on the huge corpus with the parallel implementation. This system is also used to determine the *supersense* of a previously unseen word. My results on this task significantly outperform the existing work of Ciaramita et al. (2003).

1.2 Lexical Relations

Lexical relations are very difficult concepts to define formally; a detailed account is given by Cruse (1986). Synonymy, the identity lexical relation, is recognised as having various degrees that range from complete contextual substitutability (*absolute synonymy*), truth preserving synonymy (*propositional synonymy*) through to near-synonymy (*plesionymy*). *Hyponymy*, or subsumption, is the subset lexical relation and the inverse relation is called *hypernymy* (or *hyperonymy*). *Hypernymy* can loosely be defined as the *is-a* or *is-a-kind-of* relation.

1.2.1 Synonymy and Hyponymy

Zgusta (1971) defines absolute synonymy as agreement in *designatum*, the essential properties that define a concept; *connotation*, the associated features of a concept; and *range of application*, the contexts in which the word may be used. Except for technical terms, very few instances of absolute synonymy exist. For instance, Landau (1989, pp. 110–111) gives the example of the ten synonyms of Jakob-Creutzfeldt disease, including Jakob's disease, Jones-Nevin syndrome and spongiform encephalopathy. These synonyms have formed as medical experts recognised that each instance represented the same disease.

Near-synonyms agree on any two of designatum, connotation and range of application ac-

DENOTATIONAL DIMENSIONS	CONNOTATIVE DIMENSIONS
intentional/accidental	formal/informal
continuous/intermittent	abstract/concrete
immediate/iterative	pejorative/favourable
emotional/emotionless	forceful/weak
degree	emphasis

Table 1.1: Example near-synonym differentia from DiMarco et al. (1993)

cording to Landau (1989), but this is not totally consistent with Cruse (1986), who defines plesionyms as non-truth preserving (i.e. disagreeing on designatum). Cruse's definition is summarised by (Hirst, 1995) as *words that are close in meaning . . . not fully inter-substitutable but varying in their shades of denotation, connotation, implicature, emphasis or register*. Hirst and collaborators have explored near-synonymy, which is important for lexical choice in Machine Translation and Natural Language Generation (Stede, 1996). In DiMarco et al. (1993), they analyse usage notes in the *Oxford Advanced Learners Dictionary* (1989) and *Longman's Dictionary of Contemporary English* (1987). From these entries they identified 26 dimensions of *differentiae* for designatum and 12 for connotation. Examples of these are given in Table 1.1.

DiMarco et al. (1993) add near-synonym distinctions to a Natural Language Generation (NLG) knowledge base and DiMarco (1994) shows how near-synonym differentia can form lexical relations between words. Edmonds and Hirst (2002) show how a coarse-grained ontology can be combined with sub-clusters containing differentiated plesionyms. They also describe a two-tiered lexical choice algorithm for a NLG sentence planner. Finally, Zaiu Inkpen and Hirst (2001) extract near-synonym clusters from a dictionary of near-synonym discriminations, augment it with collocation information (2002) and incorporate it into an NLG system (2003).

However, in practical NLP, the definition of lexical relations is determined by the lexical resource which is often inadequate (see Section 1.5). For instance, synonymy and hyponymy is often difficult to distinguish in practice. Another example is that WORDNET does not distinguish types from instances in the noun hierarchy: both epistemologist and Socrates appear as hyponyms of philosopher, so in practice we cannot make this distinction using WORDNET.

1.2.2 Polysemy

So far this discussion has ignored the problem of words having multiple distinct senses (*polysemy*). Sense distinctions in Roget's and WORDNET are made by placing words into different

places in the hierarchy. The similarity of two terms is highly dependent on the granularity of sense distinctions, on which lexical resources regularly disagree. Section 2.2.3 includes a comparison of the granularity of the gold-standards used in this work. WORDNET has been consistently criticised for making sense distinctions that are too fine-grained, many of which are very difficult for non-experts to distinguish between.

There have been several computational attempts to reduce the number of sense distinctions and increase the size of each synset in WORDNET (Buitelaar, 1998; Ciaramita et al., 2003; Hearst and Schütze, 1993). This is related to the problem of *supersense tagging* of unseen words described in Section 6.2.

Another major problem is that synonymy is heavily domain dependent. For instance, some words are similar in one particular domain but not in another, depending on which senses are dominant in that domain. Many applications would benefit from topical semantic similarity (the *tennis problem*), for example relating ball, racquet and net. However, Roget's is the only lexical resource which provides this information.

Finally, there is the issue of *systematic* or *regular* relations between one sense and another. For instance, a systematic relationship exists between words describing a beverage (e.g. whisky) and a quantity of that beverage (e.g. a glass of whisky). Acquiring this knowledge reduces redundancy in the lexical resource and the need for as many fine-grained sense distinctions. There have been several attempts to encode (Kilgarriff, 1995) and acquire (Buitelaar, 1998) or infer (Wilensky, 1990) systematic distinctions. A related problem is the semantic alternations that occur when words appear in context. Lapata (2001) implements simple Bayesian models of sense alternations between noun-noun compounds, adjective-noun combinations, and verbs and their complements.

1.3 Lexical Resources

Rather than struggle with a operational definition of synonymy and similarity, I will rely on lexicographers for 'correct' similarity judgements by accepting words that cooccur in thesaurus entries (*synsets*) as synonymous. Chapter 2 describes and motivates this approach and compares it with other proposed evaluation methodologies. The English thesaurus has been a popular arbiter of similarity for 150 years (Davidson, 2002), and is strongly associated with the work of Peter Mark Roget (Emblen, 1970). Synonym dictionaries first appeared for Greek and

Latin in the Renaissance, with French and German dictionaries appearing in the 18th century. In English, synonym dictionaries were slower to appear because the vocabulary was smaller and rapidly absorbing new words and evolving meanings (Landau, 1989, pp. 104–105).

Many early works were either lists of words (*lexicons*) or dictionaries of synonym discriminations (*synonymicons* or *synonymies*). These were often targeted at "coming up members of society and to eligible foreigners, whose inadequate grasp of the nuances of English synonymies might lead them to embarrassing situations" (Emblen, 1970, page 263). A typical example was William Taylor's *English Synonyms Discriminated*, published in 1813. The paragraph distinguishing between mirth and cheerfulness (page 98) is given below:

> Mirth is an effort, cheerfulness a habit of the mind; mirth is transient, and cheerfulness permanent; mirth is like a flash of lightening, that glitters with momentary brilliance, cheerfulness is the day-light of the soul, which steeps it in a perpetual serenity.

Apart from discriminating entries in popular works such as Fowler's *A Dictionary of Modern English Usage* (1926), their popularity has been limited except in advanced learner dictionaries.

1.3.1 Roget's Thesaurus

The popularity of Roget's 1852 work *Thesaurus of English Words and Phrases* was instrumental in the assimilation of the word *thesaurus*, from the Greek meaning *storehouse* or *treasure*, into English. Roget's key innovation, inspired by the importance of classification and organisation in disciplines such as chemistry and biology, was the introduction of a hierarchical structure organising synsets by topic. A testament to the quality of his original hierarchy is that it remains relatively untouched in the 150 years since its original publication (Davidson, 2002). The structure of Roget's thesaurus is described in detail in Section 2.2.3.

Unfortunately, Roget's original hierarchy has proved relatively difficult to navigate (Landau, 1989, page 107) and most descendants include an alphabetical index. Roget's thesaurus received modest critical acclaim and respectable sales although people were not sure how to use it. The biggest sales boost for the thesaurus was the overwhelming popularity of crossword puzzles which began with their regular publication in the *New York World* in 1913 (Emblen, 1970, page 278). Solvers were effectively using Roget's thesaurus to boost their own recall of answers using synonyms. The recall problem has motivated the use of thesauri in Information Retrieval (IR) and NLP. However, the structure of Roget's thesaurus and later work using such structured approaches has proved equally important in NLP.

1.3.2 Controlled vocabularies

Controlled vocabularies have been used successfully to index *maintained* (or *curated*) document collections. A *controlled vocabulary* is a thesaurus of canonical terms for describing every concept in a domain. Searching by subject involves selecting terms that correspond to the topic of interest and retrieving every document indexed by those terms.

Two of the largest and up-to-date controlled vocabularies are the *Library of Congress Subject Headings* (LCSH) and the *Medical Subject Headings* (MeSH). Both contain hierarchically structured canonical terms, listed with a description, synonyms and links to other terms. The LCSH (LOC, 2003) contains over 270 000 entries indexing the entire Library of Congress catalogue. An abridged entry for pathological psychology is given in Figure 1.1:

Psychology, Pathological
Here are entered systematic descriptions of mental disorders. Popular works . . . [on] mental disorders are entered under **mental illness**. Works on clinical aspects . . . are entered under **psychiatry**.
UF Abnormal psychology; Diseases, Mental; Mental diseases; Pathological psychology;
BT Neurology
RT Brain–Diseases; Criminal Psychology; Insanity; Mental Health; Psychiatry; Psychoanalysis
NT Adjustment disorders; Adolescent psychopathology; Brain damage; Codependency; . . .
–Cross-cultural studies

Figure 1.1: An entry from the *Library of Congress Subject Headings*

MeSH (NLM, 2004) is the National Library of Medicine's controlled vocabulary used to index articles from thousands of journals in the MEDLINE and Index Medicus databases. The MeSH hierarchy starts from general topics such as anatomy or mental disorders and narrows to specific topics such as ankle and conduct disorder. MeSH contains 21 973 terms (*descriptors*) and an additional 132 123 names from a separate chemical thesaurus. These entries are heavily cross-referenced. Part of the MeSH hierarchy and entry for psychology is given in Figure 1.2.

Other important medical controlled vocabularies are produced by the Unified Medical Language System (UMLS) project. The UMLS Metathesaurus integrates over 100 biomedical vocabularies and classifications, and links synonyms between these constituents. The SPECIALIST lexicon contains syntactic information for many terms, and the UMLS Semantic Network describes the types and categories assigned to Metathesaurus concepts and permissible relationships between these types.

Behavioural Disciplines and Activities [F04]
Behavioural Sciences [F04.096]
...
Psychology [F04.096.628]
Adolescent Psychology [F04.096.628.065]
...
Psychology, Social [F04.096.628.829]

MeSH Heading Psychology
Tree Number F04.096.628
Scope Note The science dealing with the study of mental processes and behaviour in man and animals.
Entry Term Factors, Psychological; Psychological Factors; Psychological Side Effects; ...
...
Entry Version PSYCHOL

Figure 1.2: An entry from the *Medical Subject Headings*

1.3.3 WordNet

The most influential computational lexical resource is WordNet (Fellbaum, 1998). WordNet, developed by Miller, Fellbaum and others at Princeton University, is an electronic resource, combining features of dictionaries and thesauri, inspired by current psycholinguistic theories of human lexical memory. It consists of English nouns, verbs, adjectives and adverbs organised into synsets which are connected by various lexical-semantic relations. The noun and verb synsets are organised into hierarchies based on the hypernymy relation. Section 2.3 describes the overall structure of WordNet in more detail, as does the application-based evaluation work in Section 6.2.

1.4 Applications

Lexical semantics has featured significantly throughout the history of computational manipulation of text. In IR indexing and querying collections with controlled vocabularies, and query expansion using structured thesauri or extracted similar terms have proved successful (Salton and McGill, 1983; van Rijsbergen, 1979). Roget's thesaurus, WordNet and other resources have been extremely influential in NLP research and are used in a wide range of applications. Methods for automatically extracting similar words or measuring the similarity between words have also been influential.

Recent interest in interoperability and resource sharing both in terms of software (with *web services*) and information (with the *semantic web*) has renewed interest in controlled vocabularies, ontologies and thesauri (e.g. Cruz et al. 2002).

The sections below describe some of the applications in IR and NLP that have benefited from the use of lexical semantics or similarity measures. This success over a wide range of applications demonstrates the importance of ongoing research and development of lexical-semantic resources and similarity measures.

1.4.1 Information Retrieval

Lexical-semantic resources are used in IR to bridge the gap between the user's *information need* defined in terms of concepts and the computational reality of keyword-based retrieval. Both manually and automatically developed resources have been used to alleviate this mismatch.

Controlled vocabulary indexing is used in libraries and other maintained collections employing cataloguers (see Section 1.3.2). In this approach, every document in the collection is annotated with one or more canonical terms. This is extremely time consuming and expensive as it requires expert knowledge of the structure of the controlled vocabulary. This approach is only feasible for valuable collections or collections which are reasonably static in size and topic, making it totally inappropriate for web search for example. Both the LCSH and MeSH require large teams to maintain the vocabulary and perform document classification.

The hierarchical structure of controlled vocabularies can be navigated to select query terms by concept rather than keyword; unfortunately, novices find this difficult as with Roget's thesaurus (cf. Section 1.3.1). However, the structure can help to select more specific concepts (using *narrower term* links), or more general concepts (using *broader term* links) to manipulate the quality of the search results (Foskett, 1997). As full-text indexing became feasible and electronic text collections grew, controlled vocabularies made way for keyword searching by predominantly novice users on large heterogeneous collections.

Lexical semantics is now used to help these novice users search by reformulating user queries to improve the quality of the results. Lexical resources, such as thesauri, are particularly helpful with increasing *recall*, by expanding queries with synonyms. This is because there is no longer a set of canonical index terms and the user rarely adds all of the possible terms that describe a concept. For instance, a user might type cat flu into a search engine. Given no extra information, the computer system would not be able to return results containing the term feline influenza because it does not recognise that the pairs cat/feline and flu/influenza are equivalent.

Baeza-Yates and Ribeiro-Neto (1999) describe two alternatives for adding terms to the query:

global and *local* strategies (and their combination). Local strategies add terms using *relevance based feedback* on the results of the initial query, whereas global strategies use the whole document collection and/or external resources.

Attar and Fraenkel (1977) pioneered feedback based approaches by expanding queries with terms deemed similar based on cooccurrence with query terms in the relevant query results. Xu and Croft (1996) use passage level cooccurrence to select new terms, which are then filtered by performing a correlation between the frequency distributions of query keywords and the new term. These local strategies can take into account the dependency of appropriate query expansion on the accuracy of the initial query and its results. However, they are not feasible for high demand systems or distributed document collections (e.g. web search engines).

Global query expansion may involve adding synonyms, cooccurring terms from the text, or variants formed by stemming and morphological analysis (Baeza-Yates and Ribeiro-Neto, 1999). Previously this has involved the use of controlled vocabularies, regular thesauri such as Roget's, and also more recent work with WORDNET. Query expansion using Roget's and WORDNET (Mandala et al., 1998; Voorhees, 1998) has not been particularly successful, although Voorhees (1998) did see an improvement when the query terms were manually disambiguated with respect to WORDNET senses. Grefenstette (1994) found query expansion with automatically extracted synonyms beneficial, as did Jing and Tzoukermann (1999) when they combined extracted synonyms with morphological information. Xu and Croft (1998) attempt another similarity/morphology combination by filtering stemmer variations using mutual information. Voorhees (1998) also attempts word sense disambiguation using WORDNET, while Schütze and Pedersen (1995) use an approach based on extracted synonyms and see a significant improvement in performance.

1.4.2 Natural Language Processing

NLP research has used thesauri, WORDNET and other lexical resources for many different applications. Similarity measures, either extracted from raw text (see Section 1.6) or calculated over lexical-semantic resources (see Section 1.7), have also been used widely.

One of the earliest applications that exploited the hierarchical structure of Roget's thesaurus was Masterman's work (1956) on creating an interlingua and meaning representation for early machine translation work. Masterman believed that Roget's had a strong underlying mathe-

matical structure that could be exploited using a set theoretic interpretation of the structure. According to Wilks (1998), this involved entering a reduced Roget's thesaurus hierarchy onto a set of 800 punch cards for use in a Hollerinth sorting machine. Spärck Jones (1964/1986, 1971) pioneered work in semantic similarity, defining various kinds of synonymy in terms of *rows* (synsets) for machine translation and information retrieval.

The structure of Roget's thesaurus formed the basis of early work in word sense disambiguation (WSD). Yarowsky (1992) used Roget's thesaurus to define a set of senses for each word, based on the topics that the word appeared in. The task then became a matter of disambiguating the senses (selecting one from the set) based on the context in which the terms appeared. Using a 100 word context, Yarowsky achieved 93% accuracy over a sample of 12 polysemous words.

More recently, Roget's has been effectively superseded by WORDNET, particularly in WSD, although experiments have continued using both; for example, Roget's is used for evaluation in Grefenstette (1994) and in this thesis. The Roget's topic hierarchy has been aligned with WORDNET by Kwong (1998) and Mandala et al. (1999) to overcome the *tennis problem*, and Roget's terms have been disambiguated with respect to WORDNET senses (Nastase and Szpakowicz, 2001). The hierarchy structure in Roget's has also been used in edge counting measures of semantic similarity (Jarmasz and Szpakowicz, 2003; McHale, 1998), and for computing lexical cohesion using lexical chains (Morris and Hirst, 1991). Lexical chains in turn have been used for automatically inserting hypertext links into newspaper articles (Green, 1996) and for detecting and correcting malapropisms (Hirst and St-Onge, 1998). Jarmasz (2003) gives an overview of the applications of Roget's thesaurus in NLP.

Another standard problem in NLP is how to interpret small or zero counts for events. For instance, when a word does not appear in a corpus of 1 million words, does that mean it doesn't exist or just that we haven't seen it in our first million words. I have demonstrated empirically (Curran and Osborne, 2002) that reliable, stable counts are not achievable for infrequent events even when counting over massive corpora. One standard technique is to use evidence from words known to be similar to improve the quantity of information available for each term. For instance, if you have seen cat flu, then you can reason that feline flu is unlikely to be impossible. These class-based and similarity-based smoothing techniques have become increasingly important in estimating probability distributions.

Grishman and Sterling (1994) proposed class-based smoothing for conditional probabilities using the probability estimates of similar words. Brown et al. (1992) showed that class-based

smoothing using automatically constructed clusters is effective for language modelling, which was further improved by the development of *distributional* clustering techniques (Pereira et al., 1993). Dagan et al. (1993, 1995), Dagan et al. (1999) and Lee (1999) have shown that using the distributionally nearest-neighbours improves language modelling and WSD. Lee and Pereira (1999) compare the performance of clustering and nearest-neighbour approaches. Baker and McCallum (1998) apply the distributional clustering technique to document classification because it allows for a very high degree of dimensionality reduction. Lapata (2000) has used distributional similarity smoothing in the interpretation of nominalizations.

Clark and Weir (2002) have shown measures calculated over the WORDNET hierarchy can be used for pseudo disambiguation, parse selection (Clark, 2001) and prepositional phrase (PP) attachment (Clark and Weir, 2000). Pantel and Lin (2000) use synonyms from an extracted thesaurus to significantly improve performance in unsupervised PP-attachment. Abe and Li (1996) use a tree-cut model over the WORDNET hierarchy, selected with the minimum description length (MDL) principle, to estimate the *association norm* between words. Li and Abe (1998) reuse the approach to extract case frames for resolving PP-attachment ambiguities.

Synonymy has also been used in work on identifying significant relationships between words (*collocations*). For instance, (Pearce, 2001a,b) has developed a method of determining whether two words form a strong collocation based on the principle of substitutability. If a word pair is statistically correlated more strongly than pairs of their respective synonyms from WORDNET, then they are considered a collocation. Similarity techniques have also been used to identify when terms are in idiomatic and non-compositional relationships. Lin (1999) has used similarity measures to determine if relationships between words are idiomatic or non-compositional and Baldwin et al. (2003) and Bannard et al. (2003) have used similar techniques to determine whether particle-verb constructions are non-compositional.

Similarity-based techniques have been used for text classification (Baker and McCallum, 1998) and identifying semantic orientation, e.g. determining if a review is positive or negative (Turney, 2002). In NLG, the problem is mapping from the internal representation of the system to the appropriate term. Often discourse and pragmatic constraints require the selection of a synonymous term to describe a concept (Stede, 1996, 2000). Here the near-synonym distinction between terms can be very important (Zaiu Inkpen and Hirst, 2003). Pantel and Lin (2002a) have developed a method of identifying new word senses using an efficient similarity-based clustering algorithm designed for document clustering (Pantel and Lin, 2002b).

In question answering (QA), there are several interesting problems involving semantic similarity. Pasca and Harabagiu (2001) state that lexical-semantic knowledge is required in all modules of a state-of-the-art QA system. The initial task is retrieving texts based on the question. Since a relatively small number of words are available in the user's question, query expansion is often required to boost recall. Most systems in the recent TREC competitions have used query expansion components. Other work has focused on using lexical resources to calculate the similarity between the candidate answers and the question type (Moldovan et al., 2000). Harabagiu et al. (1999) and Mihalcea and Moldovan (2001) created *eXtended* WORDNET by parsing the WORDNET glosses to create extra links. This then allows inference-based checking of candidate answers. Lin and Pantel (2001a) use a similarity measure to identify synonymous paths in dependency trees, by extension of the word similarity calculations. They call this information an *inference rule*. For example, they can identify that X wrote Y and X is the author of Y convey the same information, which is very useful in question answering (Lin and Pantel, 2001b).

This review is by no means exhaustive; lexical-semantic resources and similarity measures have been applied to a very wide range of tasks, ranging from low level processing such as stemming and smoothing, up to high-level inference in question answering. Clearly, further advancement in NLP will be enhanced by innovative development of semantic resources and measures.

1.5 Manual Construction

Like all manually constructed linguistic resources, lexical-semantic resources require a significant amount of linguistic and language expertise to develop. Manual thesaurus construction is a highly conceptual and knowledge-intensive task and thus is extremely labour intensive often involving large teams of lexicographers. This makes these resources very expensive to develop, but unlike many linguistic resources, such as annotated corpora, there is already a large consumer market for thesauri. The manual development of a controlled vocabulary thesaurus, described in detail by Aitchison et al. (2002), tends to be undertaken by government bodies in the few domains where they are still maintained.

The commercial value of thesauri means researchers have access to several different versions of Roget's thesaurus and other electronic thesauri. However, they are susceptible to the forces

of commercialism which drive the development of these resources. This often results in the inclusion of other materials and difficulties with censorship and trademarks (Landau, 1989; Morton, 1994). Since these are rarely marked in any way, they represent a significant problem for future exploitation of lexical resources in NLP. (Landau, 1989, page 108) is particularly scathing of the kind of material that is included in many modern thesauri:

> The conceptual arrangement is associated with extreme inclusiveness. Rarely used words, non-English words, names, obsolete and unidiomatic expressions, phrases: all thrown in together along with common words without any apparent principle of selection. For example, in the fourth edition of Roget's International Thesaurus – one of the best of the conceptually arranged works – we find included under the subheading *orator*: "Demosthenes, Cicero, Franklin D. Roosevelt, Winston Churchill, William Jennings Bryan." Why not Pericles or Billy Graham? When one starts to include types of things, where does one stop? ...

Landau also makes the point (Landau, 1989, page 273) that many modern thesauri have entries for extremely rare words that are not useful for almost any user. However, for some computational tasks, finding synonyms for rare words is often very important.

Even if a strict operational definition of synonymy existed there are still many problems associated with manual resource development. Modern corpus-based lexicography techniques have reduced the amount of introspection required in lexicography. However, as resources constructed by fallible humans, lexical resources have a number of problems including:

bias towards particular types of terms, senses related to particular topics etc. For instance, some specialist topics are better covered in WORDNET than others. The subtree for dog has finer-grained distinctions than for cat and worm which doesn't necessarily reflect finer-grained distinctions in reality;

low coverage of rare words and senses of frequent words. This is very problematic when the word or sense is not rare. Ciaramita et al. (2003) have found that common nouns missing from WORDNET 1.6 occurred once every 8 sentences on average in the BLLIP corpus.

consistency when classifying similar words into categories. For instance, the WORDNET lexicographer file for ionosphere (location) is different to exosphere and stratosphere (object), two other layers of the earth's atmosphere.

Even if it was possible to accurately construct complete resources for a snapshot of the language, it is constantly changing. Sense distinctions are continually being made and merged, new terminology coined, words migrating from technical domains to common language and becoming obsolete or temporarily unpopular.

In addition, many specialised topic areas require separate treatment since many terms that appear in everyday language have specialised meanings in these fields. In some technical domains, such as medicine, most common words have very specialised meanings and a significant proportion of the vocabulary does not overlap with everyday vocabulary. Burgun and Bodenreider (2001) compared an alignment of the WORDNET hierarchy with the medical lexical resource UMLS and found a very small degree of overlap between the two.

There is a clear need for fully automatic synonym extraction or in the least, methods to assist with the manual creation and updating of semantic resources. The results of the system presented in this thesis could easily support lexicographers in adding new terms and relationships to existing resources. Depending on the application, for example supersense tagging in Section 6.2, the results can be used directly to create lexical resources from raw text in new domains or specific document collections.

1.6 Automatic Approaches

This section describes the automated approaches to semantic similarity that are unrelated to the vector-space methods used throughout this thesis. There have been several different approaches to creating similarity sets or similarity scores.

Along with work in electronic versions of Roget's thesaurus, there has been considerable work in extracting semantic information from machine readable dictionaries (MRDs). Boguraev and Briscoe (1989b) gives a broad overview of processing MRDs for syntactic and semantic information. For instance, Lesk (1986) used the *Advanced Oxford Learners Dictionary* for sense disambiguation by selecting senses with the most words in common with the context. This work has been repeated using WORDNET glosses by Banerjee and Pederson (2002, 2003). Fox et al. (1988) extract a semantic network from two MRDs and Copestake (1990) extracts a taxonomy from the *Longman's Dictionary of Contemporary English*.

Apart from obtaining lexical relations from MRDs, there has been considerable success in extracting certain types of relations directly from text using shallow patterns. This work was pioneered by Hearst (1992), who showed that it was possible to extract hyponym related terms using templates like:

- X, . . . , Y and/or other Z.

- Z such as X, . . . and/or Y.

In these templates, X and Y are hyponyms of Z, and in many cases X and Y are similar, although rarely synonymous – otherwise it would not make sense to list them together. This approach has a number of advantages: it is quite efficient since it only requires shallow pattern matching on the local context and it can extract information for words that only appear once in the corpus, unlike vector-space approaches. The trade-off is that these template patterns are quite sparse and the results are often rather noisy.

Hearst and Grefenstette (1992) combine this approach with a vector-space similarity measure (Grefenstette, 1994), to overcome some of these problems. Lin et al. (2003) suggest the use of patterns like from X to Y, to identify words that are incompatible but distributionally similar. Berland and Charniak (1999) use a similar approach for identifying whole-part relations. Caraballo (1999) constructs a hierarchical structure using the hyponym relations extracted by Hearst (1992).

Another approach, often used for common and proper nouns, uses bootstrapping (Riloff and Shepherd, 1997) and multi-level bootstrapping (Riloff and Jones, 1999) to find a set of terms related to an initial seed set. Roark and Charniak (1998) use a similar approach to Riloff and Shepherd (1997) but gain significantly in performance by changing some parameters of the algorithm. Agichtein and Gravano (2000) and Agichtein et al. (2000) use a similar approach to extract information about entities, such as the location of company headquarters, and Sundaresan and Yi (2000) identify acronyms and their expansions in web pages.

1.7 Semantic Distance

There is a increasing body of literature which attempts to use the link structure of WORDNET to make semantic distance judgements. The simplest approaches involve computing the shortest number of links from one node in WORDNET to another (Leacock and Chodorow, 1998; Rada et al., 1989) using breadth-first search. Other methods constrain the breadth-first search by only allowing certain types of lexical relations to be followed at certain stages of the search (Hirst and St-Onge, 1998; St-Onge, 1995; Wu and Palmer, 1994). However, all of these methods suffer from coverage and consistency problems with WORDNET (see Section 1.5). These problems stem from the fact that, intuitively, links deeper in the hierarchy represent a shorter semantic distance than links near the root. Further, there is a changing density of links (the

fanout factor or *out degree*) for different nodes in different subjects.

These problems could either represent a lack of consistent coverage in WORDNET, or alternatively may indicate something about the granularity with which English covers concept space. There are two approaches to correcting the problem. The first set of methods involves weighting the edges of the graph by the number of outgoing and incoming links (Sussna, 1993). The second method involves collecting corpus statistics about the nodes and weighting the links according to some measure over the node frequency statistics (Jiang and Conrath, 1997; Lin, 1998d; Resnik, 1995).

Budanitsky (1999) and Budanitsky and Hirst (2001) survey and compare all of these existing semantic similarity metrics. They use correlation with the human similarity judgements from Rubenstein and Goodenough (1965) and Miller and Charles (1991) to compare the effectiveness of each method. These similarity metrics can be applied to any tree-structured semantic resource. For instance, it is possible calculate similarity over Roget's thesaurus by using the coarse hierarchy (Jarmasz, 2003; Jarmasz and Szpakowicz, 2003).

1.8 Context Space

Much of the existing work on synonym extraction and word clustering, including the template and bootstrapping methods from the previous section, is based on the *distributional hypothesis* that *similar terms appear in similar contexts*. This hypothesis indicates a clear way of comparing words: by comparing the contexts in which they occur. This is the basic principle of *vector-space models* of similarity. Each *headword* is represented by a vector of frequency counts recording the contexts that it appears in. Comparing two headwords involves directly comparing the contexts in which they appear. This broad characterisation of vector-space similarity leaves open a number of issues that concern this thesis.

The first parameter is the formal or computational definition of *context*. I am interested in contextual information at the word-level, that is, the words that appear in the neighbourhood of the *headword* in question. This thesis is limited to extracting contextual information about common nouns, although it is straightforward to extend the work to verbs, adjectives or adverbs. There are many word-level definitions of context which will be described and evaluated in Chapter 3. This approach has been implemented by many different researchers in NLP including Hindle (1990); Brown et al. (1992); Pereira et al. (1993); Ruge (1997) and Lin (1998d),

all of which are described in Chapter 3.

However, other work in IR and text classification often considers the whole document to be the context, that is, if a word appears in a document, then that document is part of the context vector (Crouch, 1988; Sanderson and Croft, 1999; Srinivasan, 1992). This is a natural choice in IR, where this information is already readily available in the inverted file index.

The second parameter of interest is how to compare two contextual vectors. These functions, which I call *similarity measures*, take the two contextual vectors and return a real number indicating their similarity or dissimilarity. IR has a long history of comparing term vectors (van Rijsbergen, 1979) and many approaches have transferred directly from there. However, new methods based on treating the vectors as conditional probability distributions have proved successful. These approaches are described and evaluated in Chapter 4. The only restriction that I make on similarity measures is that they must have time complexity linear in the length of the context vectors. This is true for practically every work in the literature, except for Jing and Tzoukermann (1999), which compares all pairs of context elements using mutual information.

The third parameter is the calculation of similarity over all of the words in the vocabulary (the *headwords*). For the purposes of evaluating the different contextual representations and measures of similarity I consider the simplest algorithm and presentation of results. For a given headword, my system computes the similarity with all other headwords in the lexicon and returns a list ranked in descending order of semantic similarity. Much of the existing work takes the similarity measure and uses a clustering algorithm to produce synonym sets or a hierarchy (e.g. Brown et al., 1992; Pereira et al., 1993). For experimental purposes, this conflates the results with interactions between the similarity measure and the clustering algorithm. It also adds considerable computational overhead to each experiment since my approach can be run on just the words required for evaluation. However, I also describe methods for improving the efficiency of the algorithm and scaling it up to extremely large corpora in Chapter 5.

Finally, there is the issue of how this semantic similarity information can be applied. Section 1.4 has presented a wide range of applications involving semantic similarity. In Chapter 6 I describe the use of similarity measurements for the task of predicting the supersense tags of previously unseen words (Ciaramita et al., 2003).

Chapter 2

Evaluation

> **evaluation**: **assessment** 0.141, examination 0.117, **appraisal** 0.115, **review** 0.091, audit 0.090, **analysis** 0.086, consultation 0.075, monitoring 0.072, testing 0.071, verification 0.069, counselling 0.065, screening 0.064, audits 0.063, consideration 0.061, inquiry 0.060, inspection 0.058, **measurement** 0.058, supervision 0.058, certification 0.058, checkup 0.057, ...

One of the most difficult aspects of developing NLP systems that involve something as nebulous as lexical semantics is evaluating the quality of the result. Chapter 1 describes some of the problems of defining synonymy. This chapter describes several existing approaches to evaluating similarity systems. It presents the framework used to evaluate the system parameters outlined in Section 1.8. These parameters: *context*, *similarity* and *methods* are explored in the next three chapters. This chapter also describes the detailed error analysis used in Chapter 6.1.

Many existing approaches are too inefficient for large-scale analysis and comparison while others are not discriminating enough because they were designed to demonstrate proof-of-concept rather than compare approaches. Many approaches do not evaluate the similarity system directly, but instead evaluate the output of clustering or filtering components. It is not possible using such an approach to avoid interactions between the similarity measure and later processing. For instance, clustering algorithms are heavily influenced by the sensitivity of the measure to outliers. Later processing can also constrain the measure function, such as requiring it to be symmetrical or maintain the triangle inequality. Application-based evaluation, such as smoothing, is popular but unfortunately conflates semantic similarity with other properties, e.g. syntactic substitutability.

This thesis focuses on similarity for common nouns, but the principles are the same for other syntactic categories. Section 2.1 summarises and critiques the evaluation methodologies described in the literature. These methodologies are grouped according to the evidence they use for evaluation: *psycholinguistic* evidence, *vocabulary tests*, *gold-standard* resources, *artificial synonyms* and *application-based* evaluation.

I aim to separate semantic similarity from other properties, which necessitates the methodology described in Section 2.2. Computing semantic similarity is posed in this methodology as the task of extracting a ranked list of synonyms for a given headword. As such, it can be treated as an IR task evaluated in terms of precision and recall, where for a given headword: *precision* is the percentage of results that are headword synonyms; and *recall* is the percentage of all headword synonyms which are extracted. These measures are described in Section 2.2.4.

Synonymy is defined in this methodology by comparison with several gold-standard thesauri which are available in electronic or paper form. This eschews the problem of defining synonymy (Section 1.2) by deferring to the expertise of lexicographers. However, the limitations of these lexical resources (Section 1.5), in particular low coverage, make evaluation more difficult. To ameliorate these problems I also uses the union of entries across multiple thesauri. The gold-standards are described and contrasted in Section 2.2.3.

This methodology is used here, and in my publications, to examine the impact of various system parameters over the next three chapters. These parameters include the context extractors described in Chapter 3 and similarity measures in Chapter 4. To make this methodology feasible a fixed list of headwords, described in Section 2.2.2, is selected, covering a range of properties to avoid bias and allow analysis of performance versus these properties in Section 6.1.

Although the above methodology is suitable for quantitative comparison of system configurations, it does not examine under what circumstances the system succeeds, and more importantly when it fails and how badly. The *error analysis*, described in Section 2.3, uses WORDNET to answer these questions by separating the extracted synonyms into their WORDNET relations, which allows analysis of the percentage of synonyms and antonyms, near and distant hyponyms/hypernyms and other lexical relatives returned by the system.

I also perform an application-based evaluation described in Chapter 6. This application involves classifying previously unseen words with coarse-grained supersense tags replicating the work of Ciaramita and Johnson (2003) using semantic similarity.

2.1 Existing Methodologies

Many approaches have been suggested for evaluating the quality of similarity resources and systems. Direct approaches compare similarity scores against human performance or expertise. Psycholinguistic evidence (Section 2.1.1), performance on standard vocabulary tests (Section 2.1.2), and direct comparison against gold-standard semantic resources (Section 2.1.3) are the direct approaches to evaluating semantic similarity described below. Indirect approaches do not use human evidence directly. Artificial synonym or ambiguity creation by splitting or combining words (Section 2.1.4) and application-based evaluation (Section 2.1.5) are indirect approaches described below. Results on direct evaluations are often easier to interpret but collecting or producing the data can be difficult (Section 1.5).

2.1.1 Psycholinguistics

Both elicited and measured psycholinguistic evidence have been used to evaluate similarity systems. Grefenstette (1994) evaluates against the *Deese Antonyms*, a collection of 33 pairs of very common adjectives and the most frequent response in free word-association experiments. Deese (1962) found that the responses were predominantly a contrastive adjective. However, Deese (1964) found the most common response for rarer adjectives was a noun the adjective frequently modified. Grefenstette's system chose the Deese antonym as the most or second most similar for 14 pairs. In many of the remaining cases, synonyms of the Deese antonyms were ranked first or second, e.g. slow-rapid, rather than slow-fast. Although this demonstrates the psychological plausibility of Grefenstette's method, the large number of antonyms extracted as synonyms is clearly a problem. Further, the Deese (1964) results suggest variability in low frequency synonyms which makes psycholinguistic results less reliable.

Rubenstein and Goodenough (1965) collected semantic distance judgements, on a real scale 0 (no similarity) – 4 (perfect synonymy), for 65 word pairs from 51 human subjects. The word pairs were selected to cover a range in semantic distances. Miller and Charles (1991) repeated these experiments 25 years later on a 30 pair subset with 38 subjects, who were asked specifically for *similarity of meaning* and told to ignore any other semantic relations. Later still Resnik (1995) repeated the subset experiment with 10 subjects via email. The correlation between mean ratings between the two sets of experiments was 0.97 and 0.96 respectively.

Resnik used these results to evaluate his WORDNET semantic distance measure and Budanitsky (1999) and Budanitsky and Hirst (2001) extend this evaluation to several measures described in the literature. McDonald (2000) demonstrates the psychological plausibility of his similarity measure using the Miller and Charles judgements and reaction times from a lexical priming task.

The original 65 judgements have been further replicated, with a significantly increased number of word pairs, by Finkelstein et al. (2002) in the WordSimilarity-353 dataset. They use the WordSimilarity-353 judgements to evaluate an IR system. Jarmasz and Szpakowicz (2003) use this dataset to evaluate their semantic distance measure over Roget's thesaurus. However, correlating the distance measures with these judgements is unreliable because of the very small set of word pairs. The WordSimilarity-353 dataset goes some way to resolving this problem.

Padó and Lapata (2003) use judgements from Hodgson (1991) to show their similarity system can distinguish between lexical-semantic relations. Lapata also uses human judgements to evaluate probabilistic models for logical metonymy (Lapata and Lascarides, 2003) and smoothing (Lapata et al., 2001). Bannard et al. (2003) elicit judgements for determining whether verb-particle expressions are non-compositional. These approaches all use the WEBEXP system (Keller et al., 1998) to collect similarity judgements from participants on the web.

Finally, Hatzivassiloglou and McKeown (1993) ask subjects to partition adjectives into non-overlapping clusters, which they then compare pairwise with extracted semantic clusters.

2.1.2 Vocabulary Tests

Landauer and Dumais (1997) used 80 questions from the vocabulary sections of the *Test of English as a Foreign Language* (TOEFL) tests to evaluate their *Latent Semantic Analysis* (Deerwester et al., 1990) similarity system. According to Landauer and Dumais a score of 64.5% is considered acceptable in the vocabulary section for admission into U.S. universities.

The Landauer and Dumais test set was reused by Turney (2001), along with 50 synonym selection questions from *English as a Second Language* (ESL) tests. Turney et al. (2003) use these tests to evaluate ensembles of similarity systems and added analogy questions from the SAT test for analysing the performance of their system on analogical reasoning problems. Finally, Jarmasz (2003) and Jarmasz and Szpakowicz (2003) extend the vocabulary evaluation by including questions extracted from the *Word Power* section of *Reader's Digest*.

Vocabulary test evaluation only provides four or five alternatives for each question which limits the ability to discriminate between with similar levels of performance. Also, the probability of randomly selecting the correct answer is high for a random guess, and even higher when often at least one option is clearly wrong in multiple-choice questions.

2.1.3 Gold-Standards

Comparison against gold-standard resources, including thesauri, machine readable dictionaries (MRDs), WORDNET, and specialised resources e.g. Levin (1993) classes, is a well established evaluation methodology for similarity systems, and is the approach taken in this thesis.

Grefenstette (1994, chap. 4) uses two gold-standards, *Roget's thesaurus* (Roget, 1911) and the *Macquarie Thesaurus* (Bernard, 1990), to demonstrate that his system performs significantly better than random selection. This involves calculating the probability P_c of two words randomly occurring in the same topic (*colliding*) and comparing that with empirical results. For Roget's thesaurus, Grefenstette assumes that each word appears in two topics (approximating the average). The simplest approach involves calculating the complement – the probability of placing the two words into two (of the thousand) different topics without collision:

$$\begin{aligned} P_c &= 1 - P_{\bar{c}} && (2.1)\\ &\approx 1 - (\frac{998}{1000})^2 && (2.2)\\ &\approx 1 - (\frac{998}{1000}\frac{997}{999}) && (2.3)\\ &\approx 0.4\% && (2.4) \end{aligned}$$

Equation 2.2 is used by Grefenstette, but this ignores the fact that a word rarely appears twice in a topic, which is taken into account by Equation 2.3. P_c is calculated in a similar way for the Macquarie except the average number of topics per word is closer to three.

Grefenstette uses his system (SEXTANT) to extract the 20 most similar pairs of words from the MERGERS corpus (Section 2.2.1). These pairs collided 8 times in Roget's, significantly more often than the one collision for 20 random pairs and the theoretical one collision in approximately 250 pairs. Grefenstette analysed the 20 most similar pairs from the HARVARD corpus (Section 2.2.1) and found around 40% of non-collisions were because the first word in the pair did not appear in Roget's. Results were significantly better on the Macquarie, which suggests caution when using low-coverage resources, such as Roget's (1911). A smaller number of pairs

were synonyms in some domain-specific contexts which are outside the coverage of a general English thesaurus. Other pairs were semantically related but not synonyms. Finally, there were the several pairs which were totally unrelated.

Grefenstette also uses definition overlap, similar to Lesk (1986), on content words from *Webster's 7th edition* (Gove, 1963) as a gold-standard for synonym evaluation.

Comparison with the currently available gold-standards suffers badly from topic sensitivity. For example, in Grefenstette's medical abstracts corpus (MED), injection and administration are very similar, but no general gold-standard would contain this information. This is exacerbated in Grefenstette's experiments by the fact that he did not have access to a large general corpus. Finally, measures that count overlap with a single gold-standard are not fine-grained enough to represent thesaurus quality because overlap is often quite rare.

Practically all recent work in semantic clustering of verbs evaluates against the Levin (1993) classes. Levin classifies verbs on the basis of their alternation behaviour. For instance, the Vehicle Names class of verbs includes balloon, bicycle, canoe, and skate. These verbs all participate in the same alternation patterns.

Lapata and Brew (1999) report the accuracy of a Bayesian model that selects Levin classes for verbs which can be disambiguated using just the subcategorisation frame. Stevenson and Merlo (1999, 2000) report the accuracy of classifying verbs with the same subcategorisation frames as either unergatives (manner of motion), unaccusatives (changes of state) or object-drop (unexpressed object alternation) verbs. In their unsupervised clustering experiments Stevenson and Merlo (1999) discuss the problem of determining the Levin class label of the cluster. Schulte im Walde (2000) reports the precision and recall of verbs clustered into Levin classes. However, in later work for German verbs, Schulte im Walde (2003) introduces an alternative evaluation using the *adjusted Rand index* (Hubert and Arabie, 1985).

Finally, Hearst (1992) and Caraballo and Charniak (1999) compare their hyponym extraction and specificity ordering techniques against the WORDNET hierarchy. Lin (1999) uses an idiom dictionary to evaluate the identification of non-compositional expressions.

2.1.4 Artificial Synonyms

Creating *artificial synonyms* involves randomly splitting the individual occurrences of a word into two or more distinct tokens to synthesise a pair of absolute synonyms. This method is in-

spired by *pseudo-words* which were first introduced for word sense disambiguation (WSD) evaluation, where two distinct words were concatenated to produce an artificial ambiguity (Gale et al., 1992; Schütze, 1992a). This technique is also used by Banko and Brill (2001) to create extremely large 'annotated' datasets for disambiguating confusion sets e.g. {to, too, two}.

Grefenstette (1994) creates artificial synonyms by converting a percentage of instances of a given word into uppercase. This gives two results: the ranking of the 'new' word in the original word's results and the ranking of the original term in the new word's list. In practice, raw text often contains relationships like artificial synonymy, such as words with multiple orthographies caused by spelling reform (e.g. colour/color), or frequent typographic errors and misspelling.

Artificial synonyms are a useful evaluation because they don't require a gold-standard and can measure performance on absolute synonymy. They can be created after context vectors have been extracted, because a word can be split by randomly splitting every count in its context vector, which makes these experiments very efficient. Further, the split ratio can easily be changed which allows performance to be compared for low and high frequency synonyms.

There are several parameters of interest for artificial synonym experiments:

frequency: the frequency of the original word. Grefenstette split the terms up into 4 classes: *frequent* (top 1%), *common* (next 5%), *ordinary* (next 25%) and *rare* (the remainder). From each class 20 words were selected for the experiments.

split: the percentage split used. Grefenstette used splits of 50%, 40%, 30%, 20%, 10%, 5% and 1% for each frequency class.

contexts: the number of unique contexts the word appears in, which is often correlated with frequency except for idiomatic expressions where a word appears in very few contexts.

polysemy: the number of senses of the original word.

Grefenstette shows that for frequent and common terms, the artificial synonyms are ranked highly, even at relatively uneven splits of 20%. However, as their frequency drops, so does the recall of artificial synonyms. Gaustad (2001) has noted that performance estimates for WSD using pseudo-word disambiguation are overly optimistic even when the distribution of the two constituent words matches the senses for a word. Nakov and Hearst (2003) suggest this is because polysemous words often have related senses rather than randomly selected pseudo-word pairs. They use MeSH to select similar terms for a more realistic evaluation.

2.1.5 Application-Based Evaluation

Application-based evaluation involves testing whether the performance on a separate task improves with the use of a similarity system. Many systems have been evaluated in the context of performing a particular task. These tasks include smoothing language models (Dagan et al., 1995, 1994), word sense disambiguation (Dagan et al., 1997; Lee, 1999), information retrieval (Grefenstette, 1994) and malapropism detection (Budanitsky, 1999; Budanitsky and Hirst, 2001). Although many researchers compare performance against systems without similarity components, unfortunately only Lee (1999) and Budanitsky (1999) have actually performed evaluation of multiple approaches within an application framework.

2.2 Methodology

The evaluation methodologies described above demonstrate the utility of the systems developed for synonym extraction and measuring semantic similarity. They show that various models of similarity can perform in ways that mimic human behaviour in psycholinguistic terms, human intuition in terms of resources we create to organise language for ourselves and human performance as compared with vocabulary testing. These methods also show how performance on wider NLP tasks can be improved significantly by incorporating similarity measures.

However, the evaluation methodologies described above are not adequate for a large-scale comparison of different similarity systems, nor capable of fully quantifying the errors and omissions that a similarity system produces. This section outlines my evaluation methodology, which is based on using several gold-standard resources and treating semantic similarity as information retrieval, evaluated in terms of precision and recall.

The overall methodology is as follows: A number of single word common nouns (70 initially and 300 for detailed analysis) are selected, covering a range of properties described in Section 2.2.2. For each of these *headwords*, synonyms from several gold-standard thesauri are either taken from files or manually entered from paper. The gold-standards used are described and compared in Section 2.2.3. The 200 most similar words are then extracted for each headword and compared with the gold-standard using precision- and recall-inspired measures described in Section 2.2.4.

2.2.1 Corpora

One of the greatest limitations of Grefenstette's experiments is the lack of a large general corpus from which to extract a thesaurus. A general corpus is important because it is not inconceivable that thesaurus quality may be better on topic specific text collections. This is because one particular sense often dominates for each word in a particular domain. If the corpus is specific to a domain the contexts are more constrained and less noisy.

Of course, it is still a significant disadvantage to be extracting from specific corpora but evaluating on a general thesaurus (Section 2.1.3). Many fields, for example medicine and astronomy, now have reasonably large ontologies which can be used for comparison and they also have large electronic collections of documents. However, evaluation on domain-specific collections is not considered in this thesis.

Grefenstette (1994, chap. 6) presents results over a very wide range of corpora including: the standard Brown corpus (Francis and Kucera, 1982); HARVARD and SPORT corpora which consist of entries from extracted from Grolier's encyclopedia containing a hyponym of institution and sport from WORDNET; MED corpus of medical abstracts; and the MERGERS corpus of Wall Street Journal articles indexed with the merger keyword. The largest is the Brown corpus.

Other research, e.g. Hearst (1992), has also extracted contextual information from reference texts, such as dictionaries or encyclopaedias. However, a primary motivation for developing automated similarity systems is replacing or aiding expensive manual construction of resources (Section 1.5). Given this, the raw text fed to such systems should not be too expensive to create and be created in large quantities, neither of which is true of reference works. However, newspaper text, journal articles and webpages satisfy these criteria.

Corpus properties that must to be considered for evaluation include:

- corpus size
- topic specificity and homogeneity
- how much noise there is in the data

Corpus size and its implications is a central concern of this thesis. Chapter 3 explores the trade-off between the type of extracted contextual information and the amount of text it can be extracted from. It also describes some experiments on different types of corpora which assess the influence of the second and third properties.

2.2.2 Selected Words

Many different properties can influence the quality of the synonyms extracted for a given headword. The most obvious property is the frequency of occurrence in the input text, since this determines how much contextual evidence is available to compare words. Other properties which may potentially impact on results include whether the headword is:

- seen in a restricted or wide range of contexts
- abstract or concrete
- specific/technical or general
- monosymous or polysemous (and to what degree)
- syntactically ambiguous
- a single or multi-word expression

It is infeasible to extract synonym lists for the entire vocabulary over a large number of experiments, so the evaluation employed in Chapters 3–5 uses a representative sample of 70 single word nouns. These nouns are shown in Table 2.1, together with counts from the Penn Treebank (PTB, Marcus et al., 1994), British National Corpus (BNC, Burnard, 1995) and the Reuters Corpus Volume 1 (RCV1, Rose et al., 2002) and sense properties from the Macquarie and Oxford thesauri and WORDNET. To avoid sample bias and provide representatives covering the parameters described above, the nouns were randomly selected from WORDNET such that they covered a range of values for the following:

occurrence frequency based on counts from the Penn Treebank, BNC and RCV1;

number of senses based on the number of Macquarie, Oxford and WORDNET synsets;

generality/specificity based on depth of the term in the WORDNET hierarchy;

abstractness/concreteness based on distribution across all WORDNET unique beginners.

The detailed evaluation uses a larger set of 300 nouns, covering several frequency bands, based on counts from the PTB, BNC, the Brown Corpus, and 100 million words of New York Times text from the ACQUAINT Corpus (Graff, 2002). The counts combine both singular, plural and alternative spelling forms. The 300 nouns were selected as follows: First, the 100 most frequent nouns were selected. Then, 30 nouns were selected from the ranges 100–50 occurrences per million (opm), 50–20 opm, 20–10 opm and 10–5 opm. 15 nouns each were selected that appeared 2 opm or 1 opm. The remaining 20 words were those missed from the original 70 word evaluation set. The 300 nouns are listed in Appendix A.

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
company	38	4 098	57 723	459 927	8	5	9	5/8	ENT, GRP, STT
market	45	3 232	33 563	537 763	4	3	4	4/10	ACT, ENT, GRP
stock	69	2 786	9 544	248 868	15	11	17	5/11	ABS, ENT, GRP, POS, STT
price	106	1 935	27 737	335 369	2	3	7	6/10	ABS, ENT, POS
government	110	1 051	66 892	333 080	3	2	4	5/9	ACT, GRP, PSY
time	116	1 318	180 053	173 378	14	8	10	3/8	ABS, EVT, PSY
people	118	907	123 644	147 061	4	5	4	3/8	GRP
interest	138	925	38 007	147 376	12	8	7	4/10	ABS, ACT, GRP, POS, STT
industry	151	927	24 140	121 348	5	3	3	7/7	ABS, ACT, GRP
chairman	184	744	10 414	65 285	1	1	1	7/7	ENT
house	230	687	49 954	69 124	10	7	12	5/8	ACT, ENT, GRP
index	244	545	4 587	123 960	5	3	5	9/11	ABS, ENT
concern	268	550	12 385	39 354	7	6	5	5/7	GRP, PSY, STT
law	311	470	31 004	61 579	8	7	7	4/10	ABS, ACT, GRP, PSY
value	315	440	25 308	56 954	12	3	6	4/9	ABS, PSY
dollar	321	581	3 700	153 394	2	–	4	7/14	ABS
street	326	431	14 777	47 275	2	1	5	5/8	ENT, GRP, STT
problem	344	623	56 361	63 344	4	3	3	5/9	ABS, PSY, STT
country	374	502	48 146	172 593	5	5	5	4/7	ENT, GRP
work	382	354	75 277	36 454	9	10	7	4/8	ACT, ENT, PHE, PSY
power	414	367	38 447	86 578	16	9	9	3/10	ABS, ENT, GRP, PHE, PSY, STT
change	536	407	40 065	55 487	9	3	10	4/14	ABS, ACT, ENT, EVT, PHE
thing	566	373	77 246	27 601	7	16	12	3/8	ABS, ACT, ENT, EVT, PSY, STT
car	595	390	35 184	45 867	4	2	5	9/10	ENT
gas	623	242	8 176	64 562	10	1	6	5/10	ENT, PHE, STT
statement	666	226	13 988	126 527	7	1	7	4/10	ABS, ACT, ENT
magazine	742	260	6 008	8 417	5	1	6	7/10	ENT, GRP
man	929	269	98 731	43 989	9	6	11	3/11	ENT, GRP
floor	1 008	138	12 690	12 056	6	4	9	5/12	ENT, GRP, PSY
hand	1 086	206	53 432	25 307	13	7	14	4/11	ABS, ACT, ENT, GRP, PSY
size	1 102	116	14 422	14 290	6	1	5	4/8	ABS, ENT, STT
energy	1 142	174	12 191	41 054	3	2	6	5/12	ABS, GRP, PHE, STT
idea	1 220	134	32 754	13 535	10	6	5	5/9	ENT, PSY
newspaper	1 220	164	8 539	58 723	1	1	4	7/10	ENT, GRP
image	1 466	97	11 026	6 697	10	8	7	5/9	ABS, ENT, PSY
book	1 487	151	37 661	16 270	7	3	8	7/9	ABS, ENT
aircraft	1 586	94	6 200	17 165	1	1	1	9/9	ENT
limit	1 661	116	6 741	14 530	2	4	6	5/8	ABS, ENT
word	1 766	124	43 744	8 839	8	11	10	6/10	ABS, ACT, PSY
opinion	1 935	80	9 295	16 378	4	1	6	6/10	ABS, ACT, PSY
apple	2 000	100	3 237	5 927	4	–	2	10/11	ENT
fear	2 187	109	9 936	19 814	2	4	2	5/6	PSY
radio	2 267	98	9 072	26 060	2	–	3	8/10	ENT
patient	2 432	63	21 653	8 048	1	1	1	7/7	ENT
crop	2 467	65	3 011	32 327	9	4	3	7/10	ACT, ENT
purpose	3 006	74	15 180	9 031	3	6	3	6/6	ABS, PSY

70 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
promotion	3 071	61	3 696	4 258	5	3	4	5/9	ABS, ACT
star	3 199	65	8 563	11 538	11	4	7	6/10	ABS, ENT
location	3 265	57	5 499	5 470	7	1	3	3/8	ACT, ENT
wine	3 603	49	7 349	3 559	1	1	2	9/10	ABS, ENT
apparel	3 788	29	67	1 978	2	1	1	7/7	ENT
human	5 099	37	2 593	7 138	1	1	2	4/11	ENT
knowledge	5 099	19	14 580	2 836	3	5	1	3/3	PSY
dream	5 537	23	6 416	2 223	8	4	6	4/8	PSY, STT
entity	6 077	28	1 818	4 352	2	3	1	2/2	ENT
taste	6 401	18	4 413	1 173	6	7	7	5/9	ABS, ACT, EVT, PSY
ball	7 130	29	8 750	7 730	9	3	10	5/11	ABS, ACT, ENT, GRP
chaos	7 130	13	1 633	2 445	2	1	3	5/9	PHE, PSY, STT
boat	8 136	17	7 345	6 128	2	1	2	10/10	ENT
fish	8 721	9	9 711	3 042	3	1	2	7/8	ENT
village	10 247	13	13 359	9 949	2	–	3	6/7	ENT, GRP
pants	12 636	5	547	282	3	2	2	9/11	ENT
religion	12 636	7	5 127	1 596	2	1	2	6/10	ABS, GRP, PSY
forum	14 425	9	1 823	6 067	4	3	3	6/10	ENT, GRP
moisture	20 917	2	699	1 806	2	1	1	5/5	STT
tightness	28 728	1	122	2 025	5	–	3	6/7	ABS, STT
announcement	–	120	2 391	22 222	2	2	2	8/9	ABS
hair	–	16	14 999	1 388	3	3	6	6/7	ABS, ENT
handful	–	36	1 489	2 574	2	2	2	6/6	ABS
mix	–	26	1 908	2 933	3	1	3	6/8	ACT, ENT, EVT

70 headword evaluation set

2.2.3 Gold-Standards

There are several drawbacks of evaluation by comparing against gold-standards (Section 2.1.3), many of which are a result of the problems that manually constructed resources suffer from (Section 1.5). This section describes one approach to overcoming these problems by combining multiple resources for the purposes of evaluation. I also describe and compare the different gold-standards to give a sense of the difficulty of the task. A comparison of Roget's, Macquarie, and information retrieval thesauri and WORDNET can be found in Kilgarriff and Yallop (2000). The gold-standard thesauri used in this evaluation are as follows:

Roget's *Roget's 1911* (Roget, 1911) and *Roget's II* (Hickok, 1995)

Moby *Moby Thesaurus* (Ward, 1996)

Oxford *The New Oxford Thesaurus of English* (Hanks, 2000)

Macquarie *The Macquarie Encyclopedic Thesaurus* (Bernard, 1990)

Abstract Relations	*Intellectual Faculties*	*Voluntary Powers*	*Sentiment and Moral P.*
Order	*Communication of Ideas*	*Individual Volition*	*Sympathetic Affections*
Collective Order	*Means of C. Ideas*	*Antagonism*	*Social Affections*
72. Assemblage	*Conventional Means*	*Conditional A.*	892. Sociality
Number	599. The Drama	712. Party	
Determinate Number		726. Combatant	
88. Accompaniment			

Figure 2.1: **company** in Roget's *Thesaurus of English words and phrases* (Roget, 1911)

Also for comparison purposes I have included the most recently completed paper thesaurus *Roget's Thesaurus: 150th Anniversary Edition* (Davidson, 2002). In terms of coverage and structure these thesauri are very different.

Roget's *Thesaurus of English words and phrases* (1911) is included in these experiments for comparison with Grefenstette (1994) and because it is freely distributed on the Internet by Project Gutenberg. It was created by MICRA, Inc. in May 1991, who scanned the out of copyright 1911 edition and released it under the name *Thesaurus-1911*. MICRA Inc. also added more than 1000 words not present in the 1911 edition. However, it has very limited coverage in many areas because of its age and many uncorrected OCR errors. In other places it suffers from Landau's extreme inclusiveness (Section 1.5), including many famous phrases, obsolete and idiomatic expressions and French, Greek and Latin expressions which are often not indicated as such. However, it has been used by a number of NLP researchers and it also is reasonably representative of Roget's 1852 edition.

The distinguishing feature of Roget's original work was the use of a hierarchical topic structure to organise the synsets. Hierarchy paths for synsets containing company are shown in Figure 2.1. The hierarchy consists of 6 top level semantic roots: Abstract Relations, Space, Matter, Intellectual Faculties, Voluntary Powers and Sentiment and Moral Powers. These were further broken down into two or three further distinctions leading to a list of 1000 *topics* ranging from 1. Existence, . . . , 622. Pursuit, . . . , 648. Goodness to 1000. Temple. These topics are often found in contrasting pairs, such as Existence-Inexistence and Goodness-Badness.

Roget's contains approximately 60 000 terms in total of which 32 000 are unique. Roget's hierarchy is relatively deep (up to seven levels) with words grouped in 8 696 small synsets within the 1000 topics at the leaves of the hierarchy. These synsets are often difficult to identify in their electronic transcription because they are delimited by punctuation alone, which is often incorrect or missing. Topics frequently contain relation pointers to other topics.

company *noun*

1. A number of persons who have come or been gathered together : assemblage, assembly, body, conclave, conference, congregation, congress, convention, convocation, crowd, gathering, group, meeting, muster, troop. *Informal:* get-together. *See* COLLECT. **2.** A person or persons visiting one : guest, visitant, visitor. *See* ACCOMPANIED. **3.** A pleasant association among people : companionship, fellowship, society. *See* CONNECT, GROUP.
4. A commercial organization : business, concern, corporation, enterprise, establishment, firm[2], house. *Informal:* outfit. *See* GROUP. **5.** A group of people acting together in a shared activity : band[2], corps, party, troop, troupe. *See* PERFORMING ARTS. **company** *verb* To be with or go with (another): accompany, attend, companion, escort. *Obsolete:* consort. ***Idiom:*** go hand in hand with. *See* ACCOMPANIED.

Figure 2.2: *Roget's II: the New Thesaurus* (Hickok, 1995) entry for **company**

Roget's II: The New Thesaurus, Third Edition (Hickok, 1995), like many modern "Roget's" thesauri, is in fact a dictionary-style thesaurus where synonyms are listed for each headword in alphabetical order. The entry for company appears in Figure 2.2. There are some similarities to the original Roget's Thesaurus including a thematic index and a large number of cross references to other entries, but it contains a larger, modern vocabulary.

The *Moby thesaurus*, released as part of the *Moby lexicon project* by Grady Ward in June 1996, consists of 30 259 head terms. It is alphabetically ordered with each entry consisting of a single large synonym list which conflates all the headword senses. Unfortunately, Moby does not make part of speech distinctions. However, it is freely available in electronic form.

The New Oxford Thesaurus of English (Hanks, 2000) is a modern alphabetically organised thesaurus, which claims to contain over 600 000 "alternative and opposite words". The Oxford contains a lot of other information: entries for discriminating between near-synonyms e.g. credulous-gullible; for selecting commonly confused terms e.g. affect-effect; and lists on subjects such as monetary units. The entry for the word company appears in Figure 2.3. The 300 entries for this thesaurus were typed from the paper edition. For consistency, this did not include entering any cross-referenced synsets, nor any lists that the entry referred to. For several nouns, including aircraft, gas and pants, this policy resulted in a headword appearing in the thesaurus, but no synonyms being associated with that headword.

The Macquarie Encyclopedic Thesaurus (Bernard, 1990) is an electronic version of a large modern thesaurus of Australian English, which claims to contain over 180 000 synonyms. The Macquarie thesaurus consists of 812 topics (similar to Roget's) containing 5602 distinct subtopic and syntactic distinctions, which are further split into 21 174 small synonym sets.

company ▶ noun ❶ *he works for the world's biggest oil company* FIRM, business, corporation, house, establishment, agency, office, bureau, institution, organization, operation, concern, enterprise, venture, undertaking, practice; conglomerate, consortium, syndicate, group, chain, combine, multiple, multinational; *informal* outfit, set-up.
– RELATED WORD: corporate.
❷ *I was greatly looking forward to the pleasure of his company* COMPANIONSHIP, presence, friendship, fellowship, closeness, amity, camaraderie, comradeship; society, association.
❸ *I'm expecting company* GUESTS, a guest, visitors, a visitor, callers, a caller, people, someone; *archaic* visitants.
❹ *he disentangled himself from the surrounding company of poets* GROUP, crowd, body, party, band, collection, assembly, assemblage, cluster, flock, herd, horde, troupe, swarm, stream, mob, throng, congregation, gathering, meeting, convention; *informal* bunch, gang, gaggle, posse, crew, pack; *Brit. informal* shower.
❺ *he recognized the company of infantry as French* UNIT, section, detachment, troop, corps, squad, squadron, platoon, battalion, division.

Figure 2.3: *New Oxford Thesaurus of English* (Hanks, 2000) entry for **company**

There is no hierarchy above the topics, instead navigation is via an alphabetical index into the topics at the back of the paper version. Slightly abridged subtopics containing company appear in Figure 2.4. From these diagrams it is clear that selecting the whole subtopic for evaluation would be unsatisfactory. The same is true for the entries for company in the 150th Anniversary Roget's shown in Appendix B. In general I have chosen the smallest sense distinctions, which is a very tough comparison.

Since some of the extracted thesauri do not distinguish between different senses, I convert the structured thesauri into headword ordered format by concatenating the synsets that the headword appears in. Initially I had planned to evaluate against the individual gold-standards and their union. Although the performance differed on each gold-standard, the ordering between systems did not vary significantly between different thesauri. For this reason, only evaluations against the union gold-standard are presented in Chapters 3–5. For the 70 noun evaluation sample, this resulted in a gold-standard thesaurus containing a total of 23 207 synonyms.

There is also a significant number of multi-word expressions. For the 70 noun testset, multi-word terms account for approximately 23% of synonyms in Roget's, 25% in the Macquarie, 14% in the Oxford and 23% in Moby. However, almost none of the context extractors described in Chapter 3 recognise multi-word words explicitly, the exception being Lin's MINIPAR, giving it a potential advantage of at least 14%. This again makes the evaluation tougher, but will allow comparison with later systems that can extract multi-word expressions.

company
n. band 701.2
company (companionship) 133.1
company (society) 701.7
company (trade) 761.3
group 307.1
ship's crew 468.1
social relations 699.1
squad 269.3
v. accompany 133.4
partner 541.4

COMPANIONSHIP 133

n. **1 companionship**, coexistence, commensalism, commensality, comradeship, partnership, presence, togetherness; **accompaniment**, backing, obbligato, support, vamp; **company**, association, concomitance, conjunction.
. . .
v. **4 accompany**, associate with, assort (*Archaic*), bear company with, chaperone, companion, company (*Archaic*), consort, join with, keep company with, run with; **escort**, arm, conduct, convoy, guide, walk; **follow**, dangle, go around with, hang about, hang round, run around with, string along with; **partner**, see, squire, take out.

FIGHTER 269

n. **3 armed forces**, armed services, army, artillery, cavalry, foot, general staff, horse, infantry, light horse, military, musketry (*Obs.*), rifles, soldiery; **navy**, flotilla, marine, R.A.N., senior service; **air force**, Kamikaze, R.A.A.F., R.A.F.; **nation in arms**, army of occupation, host (*Archaic*), land power, Sabaoth, standing army; **unit**, arm, battalion, battery, battle (*Archaic*), brigade, century, cohort, column, command, contingent, division, force, garrison, legion, maniple, regiment, section; **squad**, cadre, company, element, escadrille (*U.S.*), group, platoon, squadron, sub-unit, troop; . . .

GATHERING 307

n. **1 gathering**, association, bee, get-together, meet, muster, roll-up, turnout; **assembly**, assemblage, body, confluence, conflux, congregation, constellation, convocation; **meeting**, hui (*N.Z.*), indaba (*S. Africa*), witan, witenagemot; **group**, band, cohort, company, outfit, party, phalanx; **gang**, crew, emu parade, mob, pack, rabble, ruck, shower; **crowd**, crush, huddle, multitude, press, sea of faces, throng; **jam**, bunfight, squeeze; **coroboree**; **grouping**, class, college, school; **stable**, string; **pack**, pride; **bevy**, covey, flight, flock, gaggle; **herd**, drove, horde, troop; **shoal**, school; **association** . . .

MARINER 468

n. **1 mariner**, boatie, hearty, jack, lascar, matelot, raftsman, sailor, salt, sea-dog, seafarer, shellback, shipman (*Archaic*), shipmate, submariner, tar, tarpaulin (*Rare*); **ship's crew**, company, complement, crew, ship; **navy**, mercantile marine, merchant marine, merchant navy, senior service; **yachtsman**, rock-hopper, sailor, windsurfer, yachtswoman, yachty, yottie; **windjammer**, reefer, sheethand; **ferryman**, bargee, boatman, bumboatman, gondolier, lighterman, wherryman; **oarsman**, bow, bowhand, bowman, bow oar, canoeist, galley slave, oar, paddler, punter, rower, sculler, stroke, waterman; **rowing crew**, bank, eight, four.

PARTNER 541

v. **4 partner**, accompany, associate, assort (*Archaic*), chaperone, company (*Archaic*), consociate, consort, mate, squire; **ally with**, go into business with, hang around with, hang with, keep company with, latch on to, mess with, pal up with, string along with, take up with, tie up with; **haunt**, follow, shadow; **assist**, attend, have a hand in, help, participate, take a hand in.

Figure 2.4: *The Macquarie Thesaurus* (Bernard, 1990) entries for **company**

company
n. band 701.2
company (companionship) 133.1
company (society) 701.7
company (trade) 761.3
group 307.1
ship's crew 468.1
social relations 699.1
squad 269.3
v. accompany 133.4
partner 541.4

SOCIABILITY 699

n. **1 sociability**, companionableness, conviviality, good fellowship, gregariousness, hospitableness, hospitality, party spirit, sociableness, sociality; **cordiality**, advances, approachability, approachableness, backslapping, bonhomie, cordialness, expansiveness, gladhanding, joviality, mellowness; **social relations**, commerce, companionship, company, comradeship, fellowship, routs and revels, social intercourse, socialness, society; **open house**, welcome.

SOCIETY 701

n. **2 crowd**, army, cohort, galaxy, host; **band**, caravan, choir, chorus, company, consort (*Obs.*), flock (*Rare*), rout (*Archaic*), squad, tribe, troop (*Rare*), troupe; **corps**, body, brigade, phalanx, regiment; **meeting**, assembly, jamboree, mass meeting, muster, parade, rally, unlawful assembly; **congregation**, communion, ecclesia, parish council, vestry; **reception**, audience, durbar, levee.
...
7 corporation, body corporate, business house, enterprise, no-liability company, incorporated association, unlimited company; **establishment**, aunty, organisation; **cartel**, combine, conference (*Shipping*), consortium, monopoly, pool, ring, syndicate; **trading bloc**, common market, co-op, EEC, farmers' cooperative, OPEC; **company**, cast, firm, line-up, outfit; **partnership**, duumvirate, group practice, triumvirate; **team**, rink, side.

TRADE 761

n. **3 company**, holding company, joint-stock company, limited company, private company, private enterprise, proprietary limited company, straw company, subsidiary company, unlimited company; **conglomerate**, amalgamation, cartel, concern, cooperative, cooperative society, empire, firm, group, group, house, industry, interest, mixed business, mixed industry, multinational, pool (*U.S.*), pyramid, syndicate, transnational; **commercial centre**, entrepot, fort, marketplace, mart, office; **chamber of commerce**.

Figure 2.4: *The Macquarie Thesaurus* (Bernard, 1990) entries for **company** (continued)

Finally, it is interesting to look at the coverage of the existing thesauri by comparing the entries for the same headwords. For the company example, every thesaurus contains synonyms that do not appear in the other thesauri, that is, they all contribute words to the union gold-standard. There is also a range in the sense distinctions for company, including the use of company as a verb in only the Oxford thesaurus. Table 2.1 shows significant variation in the number of senses attributed by the Macquarie and Oxford thesauri and WORDNET for the 70 tests nouns. Also, there is no trend for a single thesaurus to prefer less or more senses than the others. The size of each entry varies dramatically between the alphabetical and topic ordered thesauri. There is also considerable disagreement on marking obsolete, slang and foreign expressions.

2.2.4 Evaluation Measures

The evaluation methodology frames semantic similarity as an information retrieval or extraction task, which involves extracting a list of synonyms for a given headword. On this basis the standard measures of *precision*, *recall* and *F-score* are applicable. *Precision* is the percentage of correct synonyms that have been extracted against the total number of terms extracted as judged by comparison with the gold-standards defined above. We fix the number of terms retrieved to 200 which makes precision effectively an accuracy measure. This is because determining a sensible cutoff for the various systems evaluated is very difficult. 200 synonyms is larger than the number of synonyms that have been used in application-based evaluations. For some very rare words several systems cannot return the full 200 synonyms.

Recall is the percentage of correct synonyms against the total number of correct synonyms in the given gold-standard. However, the recall measure is influenced heavily by the significant differences in each gold-standard. Also, it is not possible to determine whether the corpus actually contains instances of a word in every sense in the gold-standard. Thus it is not possible to measure the true recall. For instance, if the word firm only appears in an adjectival sense (solid) in the input corpus, then a system should not be penalised for missing synonyms of the noun sense (company). Finally, most applications of extracted synonyms, including the supersense classifier in Chapter 6, use a constant number of synonyms. My approach assumes that the number of correct synonyms is larger than the 200 returned by the system and focuses on precision as an evaluation measure. *F-score* is the harmonic mean of precision and recall.

The simplest evaluation measure is direct comparison of the extracted thesaurus with gold-standards (DIRECT). The comparison of the 200 proposed synonyms is a very coarse-grained measure of the performance of the system which is badly affected by low coverage in the gold-standards. Also, DIRECT does not take into consideration the ranking within the 200 terms, which is important, given that most applications will not use all 200 synonyms.

We also consider metrics that relate to how systems might use the thesaurus list, by considering the precision of the term list at certain intervals. We consider the precision of the top n synonyms (P(n)) for the top ranked term, the top 5 terms, the top 10 terms and in the detailed evaluation the top 1–20 terms. This is in keeping with much of the work that has been done to use thesaurus terms. The DIRECT evaluation is proportional to $P(200)$.

The final evaluation score is the sum of the inverse ranks (INVR) of each matching synonym

from the gold-standard. For example, if the gold-standard matches terms at ranks 3, 5 and 28, the inverse rank score is calculated as $\frac{1}{3} + \frac{1}{5} + \frac{1}{28} \approx 0.569$. With at most 200 synonyms, the maximum INVR score is approximately 5.878 ($1 + \frac{1}{2} + \cdots + \frac{1}{200}$). The inverse rank scoring method has been used in the IR community to evaluate systems that return a fixed, limited number of answers (e.g. the TREC Question Answering track). Inverse rank is a useful measure of the subtle differences between ranked results.

In the results presented in Chapters 3–5, the direct match score is summed over all test terms and each P(n) and INVR score is averaged over the extracted synonym lists for all 70 thesaurus terms. For the purposes of comparing systems it turns out that the different evaluation metrics are fairly strongly correlated (Curran, 2002).

2.3 Detailed Evaluation

The evaluation measures proposed above are effective for distinguishing between extraction systems, but are not designed to measure the quality and usability of the similarity system. In particular, it is important to know the types and seriousness of the errors a system makes and also how the system performs depending on the properties of the headword. The detailed evaluation in Chapter 6 will use the evaluation method described below to analyse the final output of my similarity system.

2.3.1 Types of Errors and Omissions

Until now the definition of error for this task has not taken into consideration the *types* of errors that can occur, proposed synonyms either appear in the gold-standard entry or they do not. However, under some circumstances and for some applications these errors may not be significant, but for others they may be critical.

The most significant problem with existing system are obvious errors – those terms which are blatantly unrelated to the headword. These make the extracted thesaurus unusable for many practical purposes. Part of the problem with Grefenstette's work was that there was no quantification of the serious errors, only the number of synonymous words that were extracted.

For the purposes of development and comparison of thesaurus extraction algorithms, it is interesting to identify the types of omissions and errors. Omissions range from total omission to

being just below the cutoff:

- the headword or synonym does not appear in the corpus
- headword and synonym share no contexts in common
- headword and synonym share no synonymous contexts in common
- headword and synonym share some contexts in common but not enough to rank highly
- headword and synonym attribute vectors are dominated by particular attributes which are not representative of their similarity
- headword and synonym share many contexts in common but not enough to rank highly

Before describing the types of errors within a complete synonym set, it is necessary to describe relationships between headword/synonym pairs:

- the words are synonymous in general English
- the words are synonymous within subdomains of English (as indicated)
- the words are synonymous within subdomains of English (not indicated)
- the words are synonymous only in specific contexts in the corpus. For example idiomatic or stylised expressions and metaphorical usage in the corpus.
- the words share one or more common hypernyms but are not synonymous. For example, dog and cat are hyponyms of animal, but they are not synonyms. These are called *sisters*.
- the words are in some other kind of lexical-semantic relation:

  antonymy which appears to be very difficult to distinguish contextually. The problem is that this relation is as strong as synonymy but negative in value. The usefulness of a thesaurus that cannot distinguish synonyms from antonyms is dubious at best.

  hyponymy/hypernymy hypernyms of a headword are rarely considered synonymous with the headword. However, hyponyms are quite regularly considered synonymous in gold-standard thesauri.

  meronymy/holonymy meronyms may or may not be considered to be synonymous. For instance, in some varieties of English, a car is commonly called a motor or wheels. In fact this appears to be a major mechanism for creating new synonyms in English.

The detailed evaluation in Section 6.1 analyses the top 10 synonyms for each of the 300 large evaluation set nouns. It counts the number of times each lexical-semantic relation appears in the synonym list. It also gives an indication of WORDNET coverage by counting the number of synonyms not seen in WORDNET.

2.4 Summary

This chapter has described existing evaluation methodologies for synonym extraction. In doing so I have discussed the strengths and weaknesses of each approach. This motivates my own evaluation methodology which is focused on distinguishing semantic similarity from other factors, such as distributional similarity or syntactic substitutability. The remainder of this thesis will use this new evaluation methodology to compare practically all of the existing vector-space similarity systems component by component.

This chapter also introduces the detailed error analysis performed on the best results that my similarity system has produced in Chapter 5. This will be complemented by an application-based evaluation described in Chapter 6.

Chapter 3

Context

> **context**: perspective 0.107, significance 0.095, **framework** 0.086, implication 0.083, regard 0.083, aspect 0.082, dimension 0.078, interpretation 0.07, meaning 0.069, nature 0.063, importance 0.062, consideration 0.061, focus 0.06, beginning 0.06, scope 0.06, continuation 0.058, relevance 0.057, emphasis 0.055, **backdrop** 0.054, **subject** 0.054, …

Context plays an central role in many statistical NLP problems. For example, the accuracy of part of speech (POS) taggers and word sense disambiguation systems depend on the *quality* and *quantity* of contextual information that these systems can extract from the training data. When predicting the POS of a word, for instance, the immediately preceding word is usually more important than the following word or the tenth previous word. A crucial part of training these systems lies in extracting from the data high-quality contextual information, in the sense of defining contexts that are both *accurate* and *correlated* with the information (the POS tags, chunks or word senses) the system is trying to extract.

The quality of contextual information is heavily dependent on the size of the training corpus: with less data available, extracting contextual information for any given phenomenon becomes less reliable. However, corpus size is no longer a limiting factor: whereas up to now researchers have typically worked with corpora of between one million and one hundred million words, it has become feasible to build much larger document collections; for example, Banko and Brill (2001) report on experiments with a one billion word corpus. However, dramatically increasing the corpus size is not without other practical consequences and limitations. For instance, Banko and Brill's experiments only used a small subset of the data available in their corpus.

Scaling context space involves balancing several competing factors, many of which can be interpreted in terms of the *quality*/*quantity* trade-off. For instance, although it is easy to collect vast quantities of text from the web, this text is often much noisier than newswire, e.g. Reuters Corpus Volume 1 (RCV1), edited newspaper text, e.g. ACQUAINT Corpus, or carefully selected samples, e.g. the British National Corpus (BNC). On the other hand, the breadth of topic coverage provided by the web or the BNC may produce better results than using news stories. Section 3.2 describes the corpora used in these experiments and Section 3.5.3 examines corpus influence on similarity systems.

The quality/quantity trade-off appears in the context *informativeness*, which is, in part, determined by the sophistication of the extraction algorithm. *Shallow processing*, such as POS tagging or chunking, identifies local syntactic relationships, while *deep processing*, such as full parsing, extracts syntactically richer information at the cost of increased complexity. Consequentially, extraction takes significantly more time and resources which results in much less text being processed in practice. Sections 3.3 and 3.4 describe various existing and new extraction processes compared in this experiment. Section 3.5.1 presents the results and Section 3.5.2 discusses the quality/quantity trade-off for similarity systems.

Finally, the extra information must be exploitable by the learning algorithms. However, the expanded space may make it infeasible to train some learners because of algorithmic efficiency or limited computational resources. Also, some algorithms cannot manage such large datasets effectively thus reducing rather than increasing the quality of the results.

Collecting reliable distributional evidence over informative contexts is crucial for exploiting the distributional hypothesis using vector-space models (Section 1.8). However, scaling context space can be a problem because these models often record every context seen in the text. Vector-space similarity is a good task to use to experiment with scaling training data. The naïve nearest-neighbour search is very simple, causing few interactions between the data and the nature of the chosen learning algorithm, making any conclusions drawn as robust as possible.

This chapter analyses a continuum of approaches to context extraction for vector-space similarity systems. These approaches differ in their linguistic sophistication, speed, reliability and the amount of information that they annotate each context with. The results in Section 3.5 establish some relationships between context informativeness and quality, algorithmic complexity and representation size and the performance of similarity systems and language systems in general.

3.1 Definitions

Formally, a *context relation* (or *context*) is a tuple (w, r, w') where w is a headword occurring in some *relation type* r, with another word w' in one or more the sentences. Each occurrence extracted from raw text is an *instance* of a context, that is, a context relation/instance is the type/token distinction. We refer to the tuple (r, w') as an *attribute* of w.

The *relation type* r labels the context with extra annotation describing the particular relationship between the two words. If there is no extra information to convey it can be empty. For instance, the relation type may convey syntactic information from grammatical relations or it may label the position of w' in a sliding window. The tuple (dog, direct-obj, walk) indicates that the term dog was the direct object of the verb walk. The context instances are extracted from the raw text, counted and stored in *attribute vectors*, which are lists of the attributes associated with a given headword and their raw frequencies. Notation for describing statistics over contexts is defined in Section 4.1.

3.2 Corpora

There is little research into the effect of corpus type, genre and size on performance of NLP systems; exceptions include studies in cross-domain parsing (Gildea, 2001; Hwa, 1999). However, it is commonly acknowledged that domain independence is a significant problem in NLP. Banko and Brill (2001) present learning curves for confusion set disambiguation on several different machine learning techniques. Early similarity systems used small (Hindle, 1990) or specialist (Grefenstette, 1994) corpora (Section 2.2.1) but with growing computing power more recent work by Lin (1998d) has used 300 million words of newspaper text.

The data used in this thesis involves two aggregated text collections. The *experimental corpus* consists of the British National Corpus and the Reuters Corpus Volume 1, and is used to compare context extractors, similarity measures and algorithms in this and the next two chapters. The *large-scale corpus* consists of the BNC, RCV1, and much of the English news holdings of the Linguistic Data Consortium (LDC). It contains over 2 billion words of text, part of which was collected and processed for Curran and Osborne (2002). It is used in the large-scale experiments and detailed evaluation in Chapter 6.

3.2.1 Experimental Corpus

The experimental corpus consists of two quite different corpora: the British National Corpus (BNC, Burnard, 1995) and the new Reuters Corpus Volume 1 (RCV1, Rose et al., 2002). The size of the two corpora are shown in Table 3.1. This is after the text has been reprocessed and includes punctuation in the word counts (unlike the quoted BNC numbers).

CORPUS	LABEL	DOCUMENTS	SENTENCES	WORDS
British National Corpus	BNC	4 124	5.6M	115M
Reuters Corpus Vol 1	RCV1	806 791	8.1M	207M

Table 3.1: *Experimental Corpus* statistics

The *British National Corpus* was collected by a consortium of publishers, industry and university partners. It consists of samples (of up to 45 000 words each) of both written and spoken British English. Approximately 10% (10 million words) is spoken and the remaining 90 million words text. The written portion has been collected from a wide range of domains (see Burnard, 1995, page 11) and a range of different formats including letters, books, periodicals, leaflets and text to be spoken (such as autocue text). It also covers a range of authors in gender, age and location. Some samples have been extracted from various sections of larger texts. The spoken component has also been designed in a similar way. These experiments have been restricted to the text component only because of problems parsing spoken text reliably.

The corpus has been marked up using SGML based on the *Text Encoding Initiative* guidelines (Sperberg-McQueen and Burnard, 2002) and includes sentence splitting, tokenization and POS tagging. The POS tagging uses the CLAWS4 tagset and tagger (Leech et al., 1994). The CLAWS4 tagger combines some common multi-word expressions such as according to and for the time being. Some higher level structures, such as lists are also marked up.

The *Reuters Corpus Volume 1* is a recently released archive of all of the English stories written by Reuters journalists between 20 August 1996 and 19 August 1997, made freely available to the research community. It consists of 806 791 news articles markedup with some meta-data using an XML schema. Unfortunately, the body text has not been marked up in any way. The text is much noisier than the heavily filtered BNC corpus and includes things like lists and tables rendered in text using whitespace and symbol characters. These are often difficult to identify automatically which adds considerable noise to the date. Also, the text has not been annotated with end of sentence markers or part of speech tags.

The text in both corpora has been retokenized using a `lex` (Lesk and Schmidt, 1975) grammar extending on the tokenizer described in (Grefenstette, 1994, pp. 149–150). This resulted in a slight increase in the number of words in the BNC because of tokenization errors in the original corpus, such as not splitting on slashes appropriately giving blood/mosquito and bites/toilet. The BNC sentence splitting was maintained and simple heuristics were used to split the RCV1 sentences based on paragraph markers, newlines and recognising acronyms.

For the scaling experiments, described in Section 3.5.2, the text is grouped into a range of corpus sizes. The written BNC and RCV1 were first randomly shuffled together to produce a single homogeneous corpus of approximately 300 million words (MWs). This is split into two 150MW corpora over which the main experimental results are averaged. We then created smaller corpora of size $\frac{1}{2}$ down to $\frac{1}{64}$th (2.34MW) of each 150MW corpus.

3.2.2 Large-Scale Corpus

The large-scale corpus consists of the BNC, RCV1, and most of the LDC's American and international newswire and newspaper text that has been collected since 1987: Continuous Speech Recognition III (CSR-III, Graff et al., 1995); North American News Text Corpus (NANTC, Graff, 1995); the NANTC supplement (NANTS, MacIntyre, 1998); and the ACQUAINT Corpus (Graff, 2002). The components and their sizes (including punctuation) are given in Table 3.2.

CORPUS	LABEL	DOCUMENTS	SENTENCES	WORDS
British National Corpus	BNC	4 124	6.2M	114M
Reuters Corpus Vol 1	RCV1	806 791	8.1M	207M
Continuous Speech Recognition-III	CSR-III	491 349	9.3M	226M
North American News Text Corpus	NANTC	930 367	23.2M	559M
North American News Text Supplement	NANTS	942 167	25.2M	507M
ACQUAINT Corpus	ACQUAINT	1 033 461	21.3M	491M

Table 3.2: *Large-Scale Corpus* statistics

The LDC has recently released the *English Gigaword* corpus (Graff, 2003) including most of the corpora listed above. I tokenized the text using the Grok-OpenNLP tokenizer (Morton, 2002) and split the sentences using MXTerminator (Reynar and Ratnaparkhi, 1997). Any sentences less than 3 words or more than 100 words long were rejected, along with sentences containing more than 5 numbers or more than 4 brackets, to reduce noise. The large-scale corpus is over 2 billion words, which makes the experiments in Chapter 6 currently the largest collection of text processed by statistical NLP tools for published research.

3.3 Existing Approaches

There are a wide range of methods in NLP, IR and data-mining that share the same basic vector-space approach to measuring similarity (Section 1.8). Where these methods often differ is the way in which "context" is defined. In these experiments we will only consider sententially and syntactically local context, unlike IR approaches such as Crouch (1988) and Sanderson and Croft (1999) which consider document level cooccurrence as context.

The context extractors described below cover a wide range of linguistic sophistication ranging from *none* (the sliding window methods), through *shallow methods* (CASS and SEXTANT) to more sophisticated *deep methods* (MINIPAR and RASP). The more sophisticated methods will produce more informative context relations by extracting relationships between syntactically related words and annotating them with extra structural information. However, the speed of each system is reduced dramatically as the sophistication increases.

The following example, from Grefenstette's MED corpus, will be used to compare extractors:

It was concluded that the carcinoembryonic antigens represent cellular constituents which are repressed during the course of differentiation of the normal digestive system epithelium and reappear in the corresponding malignant cells by a process of derepressive dedifferentiation.

Figure 3.1: Sample sentence for context extraction

3.3.1 Window Methods

Methods that define the context of the headword in terms of the neighbouring words within a limited distance (either words or characters) are called *window* methods. In these methods, a fixed-width sliding window with respect to the headword is used to collect words that often occur with the headword.

Window-based extractors have a very low complexity and so are very easy to implement and can run very quickly. They are also practically language independent once the text has been segmented. However, this implies that they do not leverage any extra linguistic information. For instance, when we are building a noun similarity system, not being able to distinguish between noun and verb terms is a significant disadvantage. It would be possible to add this information with a POS tagger, but this would reduce the simplicity, speed and language independence of the approach.

The important factors to consider for window methods are the geometry of the window and whether to consider every word within the window. There are several aspects to the geometry:

width how many words or characters does the window extend over.

symmetry whether the headword is placed in the centre of the window, i.e. does the window extend the same distance to the left and right.

boundaries whether the window is fixed regardless of boundaries, such as sentence and paragraph breaks, in the underlying text; e.g. does the window extend over sentences.

The simplest approach collects counts for every word in the window. However, another common approach is to filter the words in some way, either to eliminate high frequency but uninformative words such as function words, or to reduce the number of dimensions that must be dealt with in later processing. Finally, the window extractor may record the direction and/or position in the window using the relation type. Not recording the position, which is labelled with an asterisk in the experimental results, is a form of smoothing.

The context windows used in POS tagging and other sequence annotation tasks tends to be relatively local, such as the previous and next two or three words (see Daelemans, 1999). Normally, they do not extend beyond the sentence boundaries. Some work in vector-space similarity has also used such short lengths including lengths up to 10–20 words (e.g. McDonald, 2000). On the other hand, early experiments in word sense disambiguation used very large windows of up to 500 words (Yarowsky, 1992). Beeferman (1998) also used a 500 word window for *trigger* (collocation) extraction in a broadcast news corpus because it approximates average document length. However, as the number of words processed increases the cost of storing these contexts becomes prohibitive. Another factor is whether a word that appears so far away from the headword is informative, that is, correlated with the headword.

A practical problem with larger window models is that they may become too large to manipulate. For instance, in work on dimensionality reduction for vector-space models Schütze (1992a,b) uses a window of 100 words either side, but only considers the 1000 most frequent terms within the window. This fixes the context matrix to have rows of length 1000. Landauer and Dumais (1997) use a similar technique with *Latent Semantic Indexing* (Deerwester et al., 1990) but argue that a 500 *character* limit is more appropriate. Their reasoning is that a fixed character window will select either fewer longer (and thus more informative) words or more shorter (and thus less informative) words, extracting a consistent amount of contextual information for each headword.

MARKED	UNMARKED	DESCRIPTION
W(L_1R_1)	W(L_1R_1*)	first word to the left or right
W(L_1)	–	first word to the left
W($L_{1,2}$)	W($L_{1,2}*$)	first or second word to the left
W(L_{1-3})	W($L_{1-3}*$)	first, second or third word to the left
W(R_1)	–	first word to the right
W($R_{1,2}$)	W($R_{1,2}*$)	first or second word to the right

Table 3.3: Window context extractor geometries

Many window extractors employ a *stopword list* (or *stoplist*) containing uninformative and very frequent words, such as determiners and pronouns, which are filtered out of context relations. Eliminating stopwords significantly reduces the number of relations, but, because they are uninformative for judging similarity, this rarely impacts negatively on the quality of results. In fact, results often improve because large stopword counts can swamp other information. Jarmasz (2003) gives a list of stopwords he uses in similarity experiments and Grefenstette uses a stopword list (1994, page 151) in the *Webster's* dictionary evaluation.

The experiments in this thesis cover a range of window methods including those with and without the position and direction encoded using the relation type, and using a range (from 1 to 3 words) of window lengths. They also explore different lengths to the left and right to see which is most informative. The window geometries used are listed in Table 3.3. Extractors which do not distinguish between different directions or positions are identified with an asterisk, e.g. W(L_1R_1*) looks one word to the left and right but does not record the position in the window.

3.3.2 CASS

The CASS parser (Abney, 1991, 1996), part of Abney's SCOL system (1997), uses cascaded finite state transducers (FSTs) to produce a limited-depth parse of POS tagged text. CASS has been used in various NLP tasks including vector-space similarity for word sense disambiguation (Lee and Pereira, 1999), induction of selectional preferences (Abney and Light, 1999) and modelling lexical-semantic relations (Lapata, 2001).

The parser identifies *chunks* with 87.9% precision and 87.1% recall, and a per-word chunk accuracy of 92.1% (Abney, 1996). The parser distribution includes a large grammar for English (the `e8` demo grammar) and a tool (the `tuples` program) that extracts predicate-argument tuples out of the parse trees that CASS produces. I use the output of the C&C POS tagger (Curran and Clark, 2003a) as input to CASS. The experiments are based on SCOL version 1e.

RELATION	DESCRIPTION
subj	subject (*active frames only*)
obj	first object after the verb (*active frames only*) surface subject (*passive frames only*)
<*prep*>	head of the prepositional phrase labelled with the preposition *prep*
obj2	surface object (*passive frames only*)

Table 3.4: Some grammatical relations from CASS involving nouns

Any CASS grammatical relation (GR) that links a noun with any content word (nouns, verbs, adjectives) is a context relation. Inverse context relations are also created for noun-noun GRs; for instance, in interest rate, rate is modified by interest, so there is an inverse relation indicating interest modifies rate. The GR type is used as the relation type. Some of the most frequent GRs are shown in Table 3.4. Lee and Pereira (1999) only used the object relations and Lapata (2001) only used the object and subject relations.

The finite state parsing algorithm is very efficient. The times reported below include the POS tagging time. CASS is not capable of identifying indirect objects, so Joanis (2002, page 27) uses `tgrep` expressions to extract them. I use the default CASS output for consistency.

```
0 concluded :obj It
3 represent :obj constituents :subj antigens
10 repressed :during course :obj which
25 reappear :in cells :by process
```

Figure 3.2: CASS sample grammatical instances (from `tuples`)

3.3.3 SEXTANT

The *Semantic EXtraction from Text via Analysed Networks of Terms* (SEXTANT) system has been designed specifically for automatic thesaurus extraction (Grefenstette, 1994). It consists of a fast shallow NLP pipeline and a naïve grammatical relation extraction tool. The shallow pipeline consists of lexical and morphological analysis, POS tagging and chunking. The relation extraction tool makes five passes over the chunker output associating nouns with verbs, modifiers and prepositional phrases.

I have implemented several variants of SEXTANT that are described in detail in Section 3.4. Since the shallow pipeline and the grammatical relation extraction is fast, SEXTANT is very efficient, and has been used to process the 2 billion word collection used in Chapter 6.

One difficulty that Grefenstette (1994) describes is the interpretation of long noun compounds in SEXTANT. For instance, civil rights activist should be interpreted as ((civil rights) activist). Therefore, the extracted relations should be (civil rights) and (rights activist) rather than (civil activist). Unfortunately, SEXTANT extracts all three relations, and in general causes the two right most nouns in compounds to share all of the modifiers to their left as context. For frequent compound nouns, this made the nouns appear more similar than they were. For frequent long noun compounds, common in technical domains, this can be a significant problem. To overcome this problem Grefenstette does not allow nouns adjacent in frequent noun compounds to be similar. However, this eliminates a common form of synonym production; for instance, denim jeans can be abbreviated to denims, but Grefenstette's policy would not allow them to be similar (after morphological analysis).

3.3.4 MINIPAR

Lin (1998a) has used GRs extracted from newspaper text with MINIPAR to calculate semantic similarity, which in turn has been applied in many NLP applications (Section 1.4.2). The MINIPAR parser (Lin, 1998b) is a broad-coverage principle-based parser, a descendent of the PRINCIPAR parser (Lin, 1993, 1994). In an evaluation on the SUSANNE corpus (Sampson, 1995) MINIPAR achieves about 88% precision and 80% recall on dependency relationships (Lin, 1998b). Given the complexity of the parser, MINIPAR is quite efficient, at 300 words per second (Lin, 1998a). However, this is still significantly slower than CASS and SEXTANT.

RELATION	DESCRIPTION
appo	apposition
comp1	first complement
det	determiner
gen	genative marker
mod	the relationship between a word and its adjunct modifier
pnmod	post nominal modifier
pcomp-n	nominal complement of prepositions
post	post determiner
vrel	passive verb modifier of nouns
obj	object of verbs
obj2	second object of ditransitive verbs
subj	subject of verbs
s	surface subject

Table 3.5: Some grammatical relations from MINIPAR involving nouns

I have extracted context relations directly from the full parse tree using the pdemo program

distributed with MINIPAR. As with CASS, context relations were created for every GR that linked nouns with other content words and inverse relations were also created. Table 3.5 lists the MINIPAR grammatical relation types involving nouns (from the README file in the MINIPAR distribution). Padó and Lapata (2003) use chains of MINIPAR grammatical relations with a vector-space similarity model, which allows them to distinguish between several different types of lexical-semantic relationship.

MINIPAR is also the only extractor that identifies multi-word expressions, which means it has a minor advantage over the other approaches when it comes to the evaluation, since it has some chance of identifying the multi-word synonyms in the gold-standard thesauri which make up between 14–25% of the synonyms.

```
fin C:i:V conclude
conclude V:s:Subj it
conclude V:be:be be
conclude V:expletive:Subj it
conclude V:fc:C fin
fin C:c:COMP that
fin C:i:V represent
represent V:s:N antigen
antigen N:det:Det the
antigen N:mod:A carcinoembryonic
represent V:subj:N antigen
represent V:obj:N constituent
constituent N:mod:A cellular
constituent N:rel:C fin
fin C:whn:N which
fin C:i:V repress
repress V:be:be be
repress V:obj:N which
repress V:mod:Prep during
during Prep:pcomp-n:N course
course N:det:Det the
course N:mod:Prep of
of Prep:pcomp-n:N differentiation
differentiation N:mod:Prep of
of Prep:pcomp-n:N epithelium
epithelium N:det:Det the
epithelium N:mod:A normal
epithelium N:nn:N digestive system
digestive system N:lex-mod:(null) digestive
repress V:conj:V reappear
reappear V:subj:N which
reappear V:mod:Prep in
in Prep:pcomp-n:N cell
cell N:det:Det the
cell N:mod:A corresponding
cell N:mod:A malignant
cell N:mod:Prep by
by Prep:pcomp-n:N process
process N:det:Det a
process N:mod:Prep of
of Prep:pcomp-n:N dedifferentiation
dedifferentiation N:mod:A derepressive
```

Figure 3.3: MINIPAR sample grammatical instances (from pdemo)

3.3.5 RASP

The *Robust Accurate Statistical Parsing* project (RASP) parser (Briscoe and Carroll, 2002) uses a statistical model over the possible state transitions of an underlying LR parser with a manually constructed phrase structure grammar. RASP achieves an F-score of 76.5% on a manually annotated 500 sentence subset of the SUSANNE corpus (Sampson, 1995) using the grammatical relation-based evaluation proposed by Carroll et al. (1998).

McCarthy et al. (2003) have used RASP to extract a thesaurus of simplex and phrasal verbs which was applied to determining compositionality. Weeds and Weir (2003) have used RASP GRs for vector-space semantic similarity comparing the work of Lin (1998d) and Lee (1999) in terms of precision and recall. John Carroll has kindly supplied me with the RASP GRs from the written portion of the British National Corpus for these experiments. The RASP GRs used as context relations in this thesis are shown in Table 3.6. Once again, these are the GRs which link nouns with other content words and again inverse context relations are also generated.

RELATION	DESCRIPTION
mod	relation between head and modifier
ncmod	non-clausal modifiers (including PP, adjectival and nominal modification)
detmod	relation between noun and determiner
ncsubj	non-clausal subjects
obj	most general object relation
dobj	direct object relation, first non-clausal complement not introduced by preposition
iobj	indirect object relation, non-clausal complement introduced by preposition
obj2	second non-clausal complement in ditransitive constructions
xcomp	predicate and clausal complement with no overt subject
conj	conj used to annotate the type of conjunction and heads of conjuncts

Table 3.6: Some grammatical relations from RASP involving nouns

```
(|ncsubj| |conclude+ed:3_VVN| |It:1_PPH1| |obj|)
(|clausal| |conclude+ed:3_VVN| |represent:8_VV0|)
(|clausal| |conclude+ed:3_VVN| |reappear:26_VV0|)
(|ncsubj| |reappear:26_VV0| |antigen+s:7_NN2| _)
(|iobj| |in:27_II| |reappear:26_VV0| |cell+s:31_NN2|)
(|iobj| |by:32_II| |reappear:26_VV0| |process:34_NN1|)
(|ncsubj| |derepressive:36_VVG| |antigen+s:7_NN2| _)
(|dobj| |derepressive:36_VVG| |dedifferentiation:37_NN1| _)
(|ncsubj| |represent:8_VV0| |antigen+s:7_NN2| _)
(|dobj| |represent:8_VV0| |constituent+s:10_NN2| _)
(|clausal| |represent:8_VV0| |be+:12_VBR|)
(|ncsubj| |be+:12_VBR| |which:11_DDQ| _)
(|xcomp| _ |be+:12_VBR| |repressed:13_JJ|)
(|ncmod| _ |antigen+s:7_NN2| |carcinoembryonic:6_JJ|)
(|detmod| _ |antigen+s:7_NN2| |the:5_AT|)
(|ncmod| _ |constituent+s:10_NN2| |cellular:9_JJ|)
(|ncmod| _ |epithelium:24_NN1| |system:23_NN1|)
(|ncmod| _ |epithelium:24_NN1| |digestive:22_JJ|)
(|ncmod| _ |epithelium:24_NN1| |normal:21_JJ|)
(|detmod| _ |epithelium:24_NN1| |the:20_AT|)
```

Figure 3.4: RASP sample grammatical relations (abridged)

3.4 Approach

My approach is based on Grefenstette's SEXTANT system introduced in Section 3.3.3. Except for the window methods, SEXTANT is the simplest context extractor and is extremely fast. It uses naïve grammatical relation processing over shallow phrase chunks in place of the manually developed grammars used by the parsing approaches. The efficiency of the SEXTANT approach makes the extraction of grammatical relations from over 2 billion words of raw text feasible. Finally, it was the only context extractor not made freely available in source code or executable form, but is instead described in detail in Grefenstette (1994). Reimplementing it completes the survey of approaches to context-space similarity systems in the literature.

There are three versions of my SEXTANT implementation, each using different shallow NLP tools which vary in their sophistication, complexity and speed. SEXTANT(NB) uses simple Naïve Bayes tagging/chunking models, SEXTANT(LT) uses the *Text Tokenisation Toolkit* (Grover et al., 2000), and SEXTANT(MX) uses the C&C maximum entropy tools (Curran and Clark, 2003a,b). They demonstrate the sensitivity of SEXTANT to the quality of these components, which are described below.

3.4.1 Lexical Analysis

SEXTANT uses a lexical analyser and sentence splitter generated from a `lex` grammar (Lesk and Schmidt, 1975), reproduced in Grefenstette (1994, pp 149–150). This grammar identifies contractions (e.g. 'd and 'll), genitive markers ('s), abbreviations (such as month names) and some acronym forms. Lexical analysis is followed by simple name recognition which concatenates titlecase words into a single term when they do not directly follow a period. The lexical analysis used in my experiments is described in Sections 3.2.1 and 3.2.2.

3.4.2 Part of Speech Tagging

Grefenstette (1994) assigns a set of possible POS tags from the CLARIT dictionary (Evans et al., 1991) which occurs as part of morphological normalisation. The CMU POS tagger, a trigram tagger based on de Marcken (1990) and trained on the Brown Corpus (Francis and Kucera, 1982), is used to disambiguate the set of POS tags.

In my reimplementations, SEXTANT(NB) uses a very simple Naïve Bayes POS tagger with the same feature set as Ratnaparkhi (1996). This tagger makes local classification decisions rather than maximising the probability over the sequence using Viterbi or beam search. This is very simple to implement and is extremely fast. SEXTANT(LT) uses the LT-POS tagger from the Language Technology Group at the University of Edinburgh (Grover et al., 2000). LT-POS is the slowest of the POS taggers. SEXTANT(MX) uses a maximum entropy POS tagger developed jointly with Stephen Clark (Curran and Clark, 2003a). It has been designed to be very efficient, tagging at around 100 000 words per second. The only similar performing tool is the *Trigrams 'n' Tags* tagger (Brants, 2000) which uses a much simpler statistical model. All three taggers have been trained on the Penn Treebank (Marcus et al., 1994), so the remaining components are designed to handle the Penn POS tag set (Santorini, 1990).

3.4.3 Phrase Chunking

Grefenstette (1994) uses a simple transition table algorithm to recognise noun phrase (NP) and verb phrase (VP) chunks. The `CanBegin` table contains POS tags allowed to start an NP or VP. The `CanContinue` table contains pairs of POS tags across which the phrase may continue. The `CanEnd` table contains POS tags allowed to terminate phrases. The algorithm scans for a `CanBegin` POS tag, then collects the longest chain of `CanContinue` pairs, and finally backtracks until a `CanEnd` tag is found. Grefenstette states that the tables are designed to produce the longest possible NPs including prepositional phrases (PPs) and conjunctions.

As above, SEXTANT(NB) uses a Naïve Bayes classifier with word and POS features, SEXTANT(LT) uses the rule-based LT-CHUNK, and SEXTANT(MX) uses a maximum entropy chunker which uses the same features as the C&C Named Entity recogniser (Curran and Clark, 2003b). The Naïve Bayes and maximum entropy chunkers are trained on the entire Penn Treebank (Marcus et al., 1994) chunks extracted using the CoNLL-2000 script (Buchholz, 2000). The Penn Treebank separates PPs and conjunctions from NPs so these chunks are concatenated to match Grefenstette's table-based results.

3.4.4 Morphological Analysis

Grefenstette (1994) uses the CLARIT morphological normaliser (Evans et al., 1991) before POS tagging. My implementations use `morpha`, the Sussex morphological analyser (Minnen et al.,

[*it*]$_{NP}$ [*be conclude*]$_{VP}$ *that* [*the carcinoembryonic antigen*]$_{NP}$ [*represent*]$_{VP}$ [*cellular constituent*]$_{NP}$ *which* [*be repress*]$_{VP}$ [*during the course of differentiation of the normal digestive system epithelium*]$_{NP}$ *and* [*reappear*]$_{VP}$ [*in the correspond malignant cell by a process of derepressive dedifferentiation*]$_{NP}$.

Figure 3.5: Chunked and morphologically analysed sample sentence

2000, 2001), which is implemented using `lex` grammars for both affix splitting and generation. This analyser is also used internally by the RASP parser. `morpha` has wide coverage – nearly 100% against the CELEX lexical database (Minnen et al., 2001) – and is very efficient, analysing over 80 000 words per second (Minnen et al., 2000).

Unlike Grefenstette, the analysis is performed after POS tagging since `morpha` can use POS tag information. `morpha` often maintains sense distinctions between singular and plural nouns; for instance: spectacles is not reduced to spectacle, but fails to do so in other cases: glasses is converted to glass. This inconsistency is problematic when using morphological analysis to smooth vector-space models. The benefit of morphological smoothing of context relations is described in Section 3.5.4.

3.4.5 Grammatical Relation Extraction

After the raw text has been POS tagged and chunked, the grammatical relation extraction algorithm is run over the chunks. This consists of five passes over each sentence that first identify noun and verb phrase heads and then collect grammatical relations between each common noun and its modifiers and verbs.

Passes 1 and 2 associate adjectival, nominal and PP modifiers with the nouns they modify and also identifies the head within the NP. VP heads and their voice (active, passive or attributive) are then identified. Passes 3–5 associate verbs with their subjects and objects. A global list of grammatical relations generated by each pass is maintained across the passes. The global list is used to determine if a word is already attached. Once all five passes have been completed this association list contains all of the noun-modifier/verb pairs which have been extracted from the sentence. The grammatical relations extracted by SEXTANT are shown in Table 3.7. As with the previous parsing context extractors, inverse context relations are also created for noun-noun grammatical relations (nn and nnprep). Figure 3.6 shows the grammatical relations extracted for the sample sentence.

RELATION	DESCRIPTION
adj	relation between a noun and an adjectival modifier
dobj	relation between a verb and a direct object
iobj	relation between a verb and an indirect object
nn	relation between a noun and a noun modifier
nnprep	relation between a noun and the head of a PP modifier
subj	relation between a verb and a subject

Table 3.7: Grammatical relations from SEXTANT

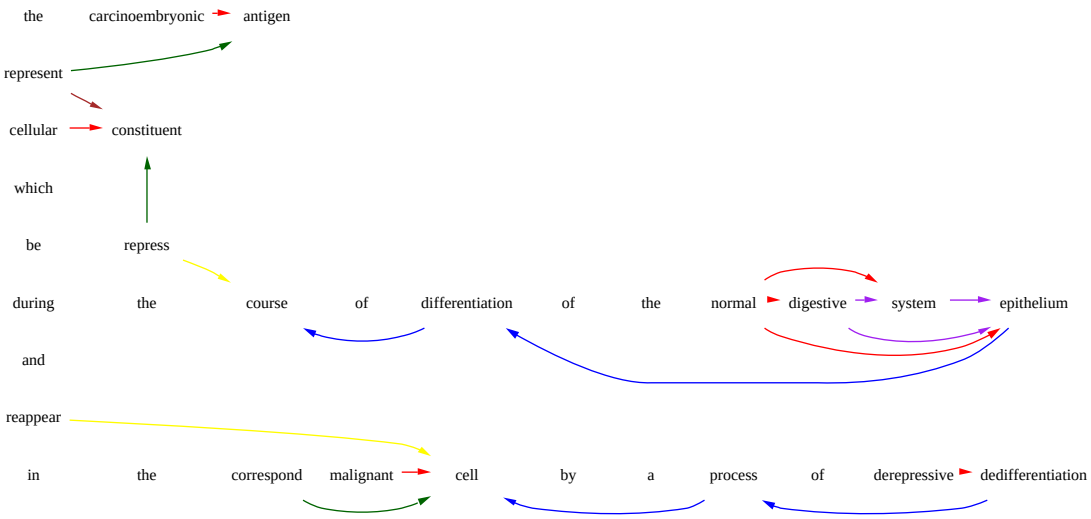

Figure 3.6: SEXTANT sample grammatical relations coloured as follows: **adj**, **nn**, **nnprep**, **subj**, **dobj** and **iobj**

Pass 1: Noun Pre-modifiers

This pass scans NPs, left to right, creating adjectival (adj) and nominal (nn) pre-modifier GRs with every noun to the pre-modifier's right, up to a preposition or the phrase end. This corresponds to assuming right-branching noun compounds. For example, normal forms an adj GR with digestive, system and epithelium in the sample sentence. Within each NP only the NP and PP heads remain unattached.

Pass 2: Noun Post-modifiers

This pass scans NPs, right to left, creating post-modifier GRs between the unattached heads of NPs and PPs. If a preposition is encountered between the noun heads, a prepositional noun (nnprep) GR is created, otherwise an appositional noun (nn) GR is created. This corresponds to assuming right-branching PP-attachment. For example, dedifferentiation modifies process,

which in turn modifies cell. After this phrase only the NP head remains unattached.

Tense Determination

The rightmost verb in each VP is considered the head. A VP is initially categorized as active. If the head verb is a form of *be* then the VP becomes attributive. Otherwise, the algorithm scans the VP from right to left: if an auxiliary verb form of be is encountered the VP becomes passive; if a progressive verb (except being) is encountered the VP becomes active.

Only the noun heads on either side of VPs remain unattached. The remaining three passes attach these to the verb heads as either subjects or objects depending on the voice of the VP.

Pass 3: Verb Pre-Attachment

This pass scans sentences, right to left, associating the first NP head to the left of the VP with its head. If the VP is active, a subject (subj) relation is created; otherwise, a direct object (dobj) relation is created. For example, antigen is the subject of represent.

Pass 4: Verb Post-Attachment

This pass scans sentences, left to right, associating the first NP or PP head to the right of the VP with its head. If the VP was classed as active and the phrase is an NP then a direct object (dobj) relation is created. If the VP was classed as passive and the phrase is an NP then a subject (subj) relation is created. If the following phrase is a PP then an indirect object (iobj) relation is created. The interaction between the head verb and the preposition determine whether the noun is an indirect object of a ditransitive verb or alternatively the head of a PP that is modifying the verb. However, SEXTANT always attaches the PP to the previous phrase.

Pass 5: Verb Progressive Participles

The final step of the process is to attach progressive verbs to subjects and objects (without concern for whether they are already attached). Progressive verbs can function as nouns, verbs and adjectives and once again a naïve approximation to the correct attachment is made. Any progressive verb which appears after a determiner or quantifier is considered a noun. Otherwise, it is considered a verb and passes 3 and 4 are repeated to attach subjects and direct objects. This pass is dependent on the way the chunker includes progressive participles.

Finally, SEXTANT collapses the nn, nnprep and adj relations together into a single broad modifier grammatical relation. Grefenstette (1994, page 46) claims this extractor has a grammatical relation accuracy of 75% after manually checking 60 sentences.

SYSTEM	SPACE MB	RELS. M	ATTRS. M	TERMS k	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –	TIME –
RASP	267	43.22	12.44	185	1956	70	53	45	2.08	6.0 d
SEXTANT(LT)	178	20.37	7.84	97	1835	70	53	45	2.04	3.2 d
MINIPAR	376	68.41	16.22	828	1921	65	50	44	2.01	1.9 d
CASS	191	38.30	9.31	191	1517	53	43	35	1.63	1.6 hr
SEXTANT(NB)	269	30.25	11.92	268	1856	73	51	44	2.05	1.4 hr
SEXTANT(MX)	313	38.78	16.11	400	1910	71	51	44	2.08	1.1 hr
W(L_1)	124	79.81	7.50	422	1660	64	46	38	1.84	6.7 m
W($L_{1,2}$)	348	155.41	18.75	463	1705	64	50	41	1.91	7.0 m
W($L_{1,2}*$)	269	155.41	16.03	463	1679	69	47	41	1.91	7.0 m
W(L_{1-3})	582	226.40	31.34	467	1623	69	47	41	1.87	7.5 m
W($L_{1-3}*$)	401	226.40	23.65	467	1603	60	45	39	1.77	7.5 m
W(L_1R_1)	278	159.62	14.97	452	1775	73	50	41	2.00	7.0 m
W(L_1R_1*)	224	159.62	13.35	452	1700	63	49	40	1.91	7.0 m
W(R_1)	124	79.81	7.49	371	1277	44	28	24	1.23	6.7 m
W($R_{1,2}$)	348	155.41	18.79	438	1490	47	39	32	1.47	7.0 m

Table 3.8: Thesaurus quality results for different context extractors

3.5 Results

There are four sets of results related to context extraction. The first results compare context extractors on the written portion of the British National Corpus (BNC). Section 3.5.2 investigate the impact of corpus size on similarity systems and the trade-off between corpus size, running time and representation size using the experimental corpus (Section 3.2.1). Section 3.5.3 considers the impact of corpus type by comparing results from the BNC and RCV1 text. The remaining sections investigate the benefit of smoothing and filtering the context representation. The 70 word experimental test set (Section 2.2.2) is used for all of these experiments. Similarity has been calculated using the JACCARD measure with the TTEST weighting function, which is found to be the best semantic similarity measure function in the next chapter.

3.5.1 Context Extractors

Table 3.8 summarises the representation size and performance of each context extractor applied to the written portion of the BNC, which was used because the RASP GRs were supplied by John Carroll and SEXTANT(LT) took too long to process the RCV1 corpus.

RASP performs significantly better than the other context extractors using the direct match evaluation but MINIPAR and SEXTANT(NB) also produce quite similar results over the other evaluation metrics. Amongst the simpler methods, W(L_1R_1) and W($L_{1,2}$) give reasonable results. Depending on the components, the shallow methods vary quite considerably in performance. Of these the state-of-the-art maximum entropy SEXTANT(MX) performs the best. Overall, the more sophisticated parsers outperform the shallow parsing approaches which significantly outperform the majority of window-based approaches.

The first thing to note is the time spent extracting contextual information: RASP, SEXTANT(LT) and MINIPAR take significantly longer to run (in days) than the other extractors and the window methods run extremely quickly (in minutes). SEXTANT(MX), which has been designed for speed, runs 40 times faster than MINIPAR and over 120 times faster than RASP, but performs almost as well. These ratios are only approximate because RASP was run on different (but comparable) hardware. On the other hand, SEXTANT(LT) was one of the slowest systems even though it was also a very shallow approach, clearly implementation efficiency is important.

Also, MINIPAR extracts many more headwords and relations with a much larger representation than SEXTANT, whereas RASP extracts more relations for a smaller number of headwords. This is partly because MINIPAR extracts more types of relations from the parse tree than SEXTANT and RASP and partly because it extracts extra multi-word expressions. The larger window methods have low correlation between the headword and context and so extract a massive context representation, but the results are over 10% worse than the syntactic extractors.

Given a medium-sized corpus and a reasonable amount of time, it is clear that RASP or MINIPAR will produce the best results. However, the choice is no longer obvious when the quantity of raw text available is effectively unlimited.

3.5.2 Corpus Size

Similarity systems need large quantities of text to reliably extract contextual information. In light of the amount of raw text now freely available in news corpora (Section 3.2.2) and on the web, we must reconsider the limiting factors of the previous results. Table 3.9 shows what happens to thesaurus quality as we decrease the size of the corpus to $\frac{1}{64}$th of its original size (2.3MWS) for SEXTANT(NB). Halving the corpus results in a significant reduction for most of the measures. All five evaluation measures show the same log-linear dependence on the size of the corpus. Figure 3.7 shows the same trend for Inverse Rank evaluation of the MINIPAR extractor with a log-linear fitting the data points.

CORPUS MWS	SPACE MB	RELS. M	ATTRS. M	TERMS k	DIRECT AVG	P(1) %	P(5) %	P(10) %	INVR –
150.0	274	53.07	12.08	268.94	23.75	64.5	47.0	39.0	1.85
75.0	166	26.54	7.38	181.73	22.60	58.0	43.5	36.0	1.73
37.5	98	13.27	4.36	120.48	21.75	54.0	41.0	34.5	1.62
18.8	56	6.63	2.54	82.33	20.45	47.0	36.5	31.0	1.46
9.4	32	3.32	1.44	55.55	18.50	40.0	32.5	27.5	1.29
4.7	18	1.66	0.82	37.95	16.65	34.0	29.5	23.5	1.13
2.3	10	0.83	0.46	25.97	14.60	27.5	25.0	19.5	0.93

Table 3.9: Average SEXTANT(NB) results for different corpus sizes

We can use the same curve fitting to estimate thesaurus quality on larger corpora for three of the best extractors: SEXTANT(NB), MINIPAR and W(L_1R_1). Figure 3.8 does this with the direct match evaluation. The estimate indicates that MINIPAR will continue to be the best performer on direct matching. We then plot the direct match scores for the 300MW corpus to see how accurate our predictions are. The SEXTANT(NB) system performs almost exactly as predicted and the other two slightly under-perform their predicted scores, thus the fitting is accurate enough to make reasonable predictions.

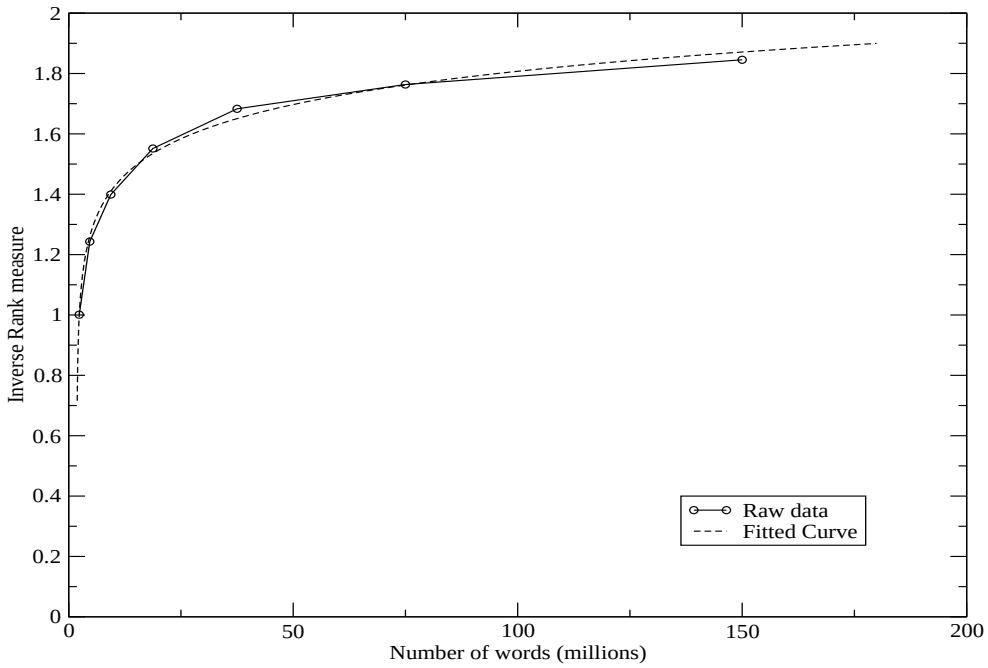

Figure 3.7: MINIPAR INVR scores versus corpus size

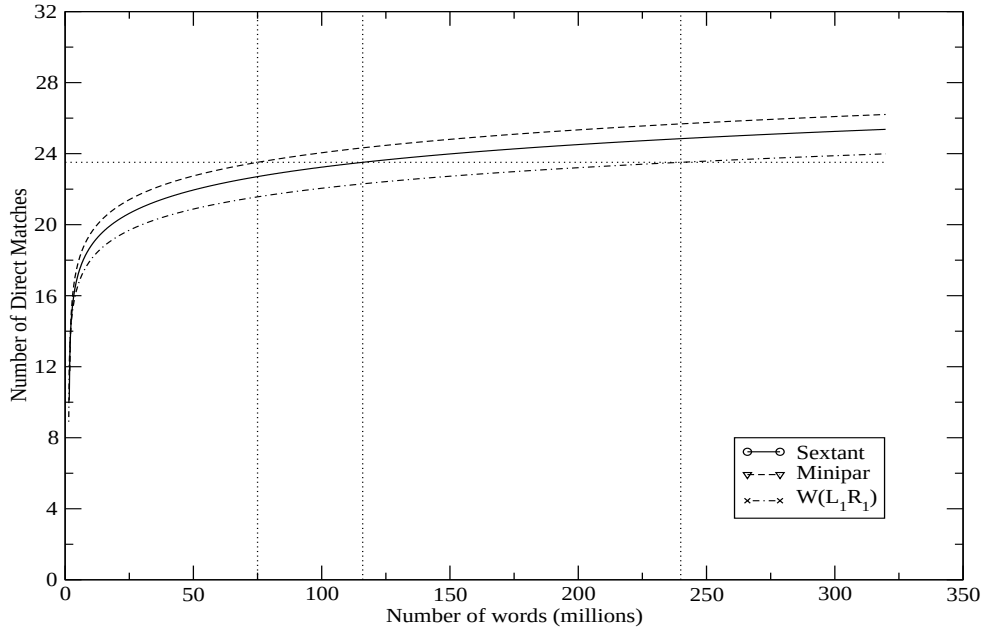

Figure 3.8: DIRECT matches versus corpus size

Using Figure 3.8 and the timing data in Table 3.8 it is possible to make "engineering" decisions regarding the trade-off between extractor complexity, relation quality and speed. For instance, if we fix the total time and computational resources at an arbitrary point, e.g. the point where MINIPAR can process 75 MWs, we get an average direct match score of 23.5. However, we can get the same resultant accuracy by using SEXTANT(NB) on a corpus of 116 MWs or W(L_1R_1) on a corpus of 240 MWs. From Figure 3.8, extracting contexts from corpora of these sizes would take MINIPAR 37 hours, SEXTANT(NB) 2 hours and W(L_1R_1) 12 minutes.

However, there is an almost linear relationship between the amount of raw text consumed and the size of the resulting model, in terms of the number of unique relations and the number of headwords. Interpolation on Figure 3.9 predicts that the extraction would result in 10M unique relations from MINIPAR and SEXTANT(NB) and 19M from W(L_1R_1). Figure 3.10 indicates that extraction would result in 550k MINIPAR headwords, 200k SEXTANT(NB) headwords and 600k W(L_1R_1) headwords. The window methods and MINIPAR suffer from the greatest representation inflation as the raw text is consumed.

These results suggest that accurate, efficient shallow context extractors, such as SEXTANT(MX), are the most successful approach when large quantities of text are available.

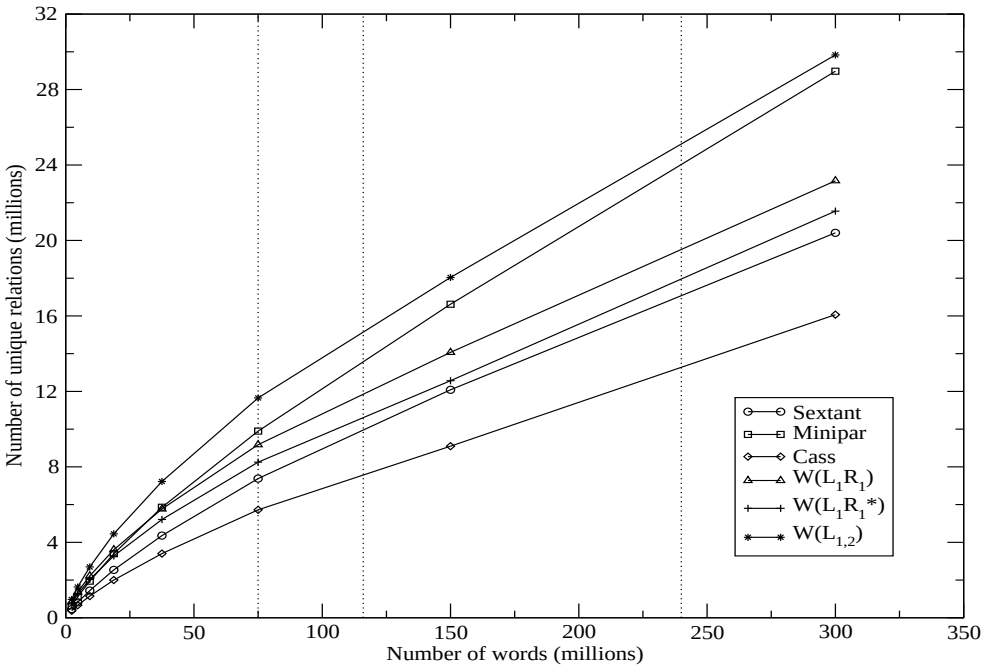

Figure 3.9: Representation size versus corpus size

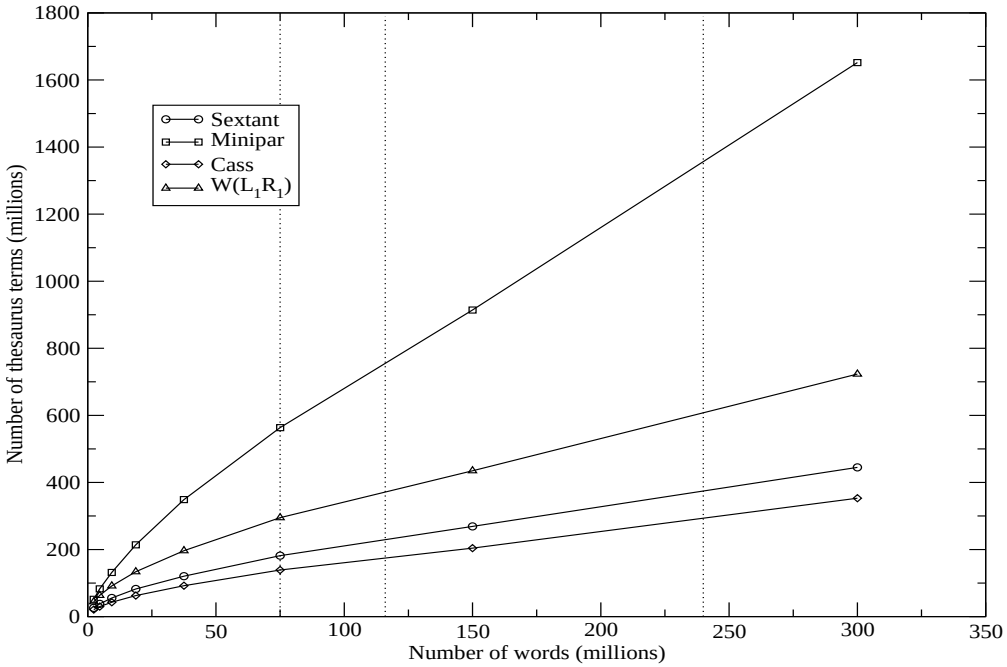

Figure 3.10: Thesaurus terms versus corpus size

SYSTEM	CORPUS	SPACE MB	RELS. M	ATTRS. M	TERMS k	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –
MINIPAR	BNC	376	68.41	16.22	828	1921	65.0	50.0	44.0	2.01
MINIPAR	RCV1	422	117.61	17.02	1001	1472	58.0	43.0	37.0	1.68
SEXTANT(NB)	BNC	269	30.25	11.92	268	1856	73.0	51.0	44.0	2.05
SEXTANT(NB)	RCV1	279	75.89	12.25	269	1468	56.0	43.0	34.0	1.66
W(L_1R_1)	BNC	278	159.62	14.97	452	1775	73.0	50.0	41.0	2.00
W(L_1R_1)	RCV1	245	262.87	13.17	417	1356	51.0	39.0	33.0	1.52

Table 3.10: Results on BNC and RCV1 for different context extractors

MORPH.	SPACE MB	ATTRS. M	TERMS k	DIRECT AVG	P(1) %	P(5) %	P(10) %	INVR –
none	345	14.70	298	20.33	32.5	36.9	33.6	1.37
attributes	302	13.17	298	20.65	32.0	37.6	32.5	1.36
both	274	12.08	269	23.74	64.5	47.0	39.0	1.86

Table 3.11: Effect of morphological analysis on SEXTANT(NB) thesaurus quality

3.5.3 Corpus Type

Table 3.10 gives the results for MINIPAR, SEXTANT(NB) and W(L_1R_1) on the BNC and RCV1 text collections. The performance from all of the systems is significantly better on the BNC than the RCV1 corpus. There are a number of possible explanations for this including that the BNC has been more heavily edited and so is a much cleaner corpus; also it contains a wide range of genres and, perhaps more importantly, topics.

Unfortunately, BNC and RCV1 differ in noise and topic coverage so it is not possible to draw a stronger conclusion. Clearly corpus type has a very large impact on performance. These results illuminate another aspect of the *quality*/*quantity* trade-off. Assembling a very large corpus of freely available raw text will not guarantee an improvement in performance. Creating a noisy corpus with wide topic coverage will allow the dominant factor in these results to be identified.

3.5.4 Smoothing

Since MINIPAR and RASP perform morphological analysis on the context relations we have added an existing morphological analyser (Minnen et al., 2000) to the other extractors. Table 3.11 shows the improvement gained by morphological analysis of the attributes and relations for the SEXTANT(NB) 150MW corpus.

SYSTEM	SPACE MB	RELS. M	ATTRS. M	TERMS k	DIRECT AVG	P(1) %	P(5) %	P(10) %	INVR –
300M	431	80.33	20.41	445	25.30	61.0	47.0	39.0	1.87
150M	274	53.07	12.08	269	23.75	64.5	47.0	39.0	1.85
FIXED	244	61.17	10.74	265	24.35	65.0	46.5	38.5	1.86
LEXICON	410	78.69	18.09	264	25.25	62.0	47.0	40.0	1.87
>1	149	67.97	6.63	171	24.20	66.0	45.0	38.0	1.85
>2	88	62.57	3.93	109	23.20	66.0	46.0	36.0	1.82

Table 3.12: Thesaurus quality with relation filtering

The morphological analysis of the attributes does not significantly affect performance but it does reduce the representation size. However, when both headwords and attributes are processed, improvement in results is very large, as is the reduction in the representation size and the number of context relations. The reduction in the number of terms is a result of coalescing the plural nouns with their corresponding singular nouns, which greatly reduces the data sparseness problems. The morphological analysis makes a significant impact on the data-sparseness problem, unlike the minimal improvement for PP-attachment (Collins and Brooks, 1995). The rest of the experiments use morphological analysis of both the headwords and attributes.

3.5.5 Filtering

The context representation is very large even for the most constrained context extractor. This section considers some methods for limiting the size of the context representation. Table 3.12 shows the results of performing various kinds of filtering on the representation size.

The FIXED and LEXICON filters run over the full 300MW corpus, but have size limits based on the 150MW corpus. The FIXED filter does not allow any object/attribute pairs to be added that were not extracted from the 150MW corpus. The LEXICON filter does not allow any objects to be added that were not extracted from the 150MW corpus. The FIXED and LEXICON filters show that counting over larger corpora does produce marginally better results, that is, getting more reliable counts for the same contexts does slightly improve performance.

The >1 and >2 filters prune relations with a frequency of less than or equal to one or two. The >1 and >2 filters show that the many relations that occur infrequently do not contribute significantly to the vector comparisons and hence do not impact on the final results, even though they dramatically increase the representation size.

3.6 Future Work

The context experiments in this chapter leave several open problems to be explored. Firstly, there are still context extractors missing from these experiments. for example, the very large window methods described in Section 3.3.1 that use only the 1000 most frequent attributes. How does this limit impact on performance for computationally-intensive approaches like Latent Semantic Analysis (LSA). There are also many combinations of grammatical relations from the parsing extractors which are worth exploring individually. This chapter has only discussed using all of the GRs that are associated with nouns for each extractor. There are a number of larger problems described below that build on this work.

3.6.1 Multi-word Terms

Most of the context extractors only handle single word terms. However, around 25% of terms in manually created thesauri are multi-word (Section 2.2.3). The treatment of multi-word terms has not been adequately treated in most NLP tasks. Few POS taggers use knowledge of compound nouns or phrasal verbs to improve their accuracy. The first problem is identifying multi-word expressions and the second is incorporating them into the shallow pipeline. Adding multi-word terms will significantly increase the representation size. However, they should improve the attribute quality by removing highly correlated contexts (e.g. (rate, nn, interest) and splitting up very high frequency attributes (e.g. verb-particle will split (object, get)).

3.6.2 Topic Specific Corpora

There is an increasing interest in extracting technical and specialised terms, usage and vocabulary from topic specific corpora. There are two motivations for doing this. Firstly, lexical resources for many of these domains is scarce while raw text is usually in abundance, so automatic extraction methods are particularly attractive. I would like to extract thesauri for domains such as bioinformatics, which generate vast amounts of text and already have lexical resources available for evaluation.

Secondly, comparing extracted synonyms may provide an avenue for comparing the vocabulary of particular specialised domains with everyday usage. Additions and omissions would then indicate differences in language usage between the domains. Direct comparison of the attribute

vectors may also highlight differing usages in different domains. If the attribute vectors of a term from two different corpora were quite different then it is likely that the term has a different meaning in each corpus.

3.6.3 Creating a Thesaurus from the Web

Finally, I would like to construct a thesaurus from webpages spidered from the web. Firstly, this would demonstrate the efficiency of the SEXTANT(MX) shallow pipeline and the parallel implementation described in Section 5.3. It could also be used to address the noise or coverage question posed by the corpus type results in Section 3.5.3. There are two components that would need to be added to the similarity system. The first component is a web spider for collecting randomly distributed web pages. The second component is new tokenization and text processing that takes into account the HTML tags. This component will be crucial in extracting text which is relatively noise-free. Using document clustering techniques from IR or document collections generated from domain-specific queries it may be possible to build topic specific thesauri from large general text collections such as the web. These topic specific thesauri could then be compared and perhaps merged together into a single thesaurus with topic markers.

3.7 Summary

This chapter has introduced and compared a wide range of approaches to extracting contextual information for measuring semantic similarity. The performance of these approaches was correlated with the sophistication of the linguistic processing involved. Unfortunately, the best systems are therefore also the least scalable. Until recently, large enough quantities of text were not available to make efficiency an issue. However, my results in Section 3.5.2 demonstrate that once we have effectively unlimited amounts of raw text, shallow systems which are linguistically informed but very efficient can prove to be the most effective.

It is a phenomenon common to many NLP tasks that the quality or accuracy of a system increases log-linearly with the size of the corpus. Banko and Brill (2001) also found this trend for the task of confusion set disambiguation on corpora of up to one billion words. They demonstrated behaviour of different learning algorithms with very simple contexts on extremely large corpora. We have demonstrated the behaviour of a simple learning algorithm on much more complicated contextual information on very large corpora.

My experiments suggest that the existing methodology of evaluating systems on small corpora without reference to the execution time and representation size ignores important aspects of the evaluation of NLP tools.

These experiments show that efficiently implementing and optimising the NLP tools used for context extraction is of crucial importance, since the increased corpus sizes make execution speed an important evaluation factor when deciding between different learning algorithms for different tasks and corpora. These results also motivate further research into improving the asymptotic complexity of the learning algorithms used in NLP systems. In the new paradigm, it could well be that far simpler but scalable learning algorithms significantly outperform existing systems.

Chapter 4

Similarity

> **similarity**: **resemblance** 0.122, **parallel** 0.083, contrast 0.061, flaw 0.060, discrepancy 0.060, difference 0.056, **affinity** 0.052, aspect 0.052, **correlation** 0.052, variation 0.052, contradiction 0.051, distinction 0.050, divergence 0.049, commonality 0.049, disparity 0.048, characteristic 0.048, shortcoming 0.048, significance 0.046, clue 0.046, hallmark 0.045, ...

Once an accurate and informative contextual representation of each headword has been extracted from raw text, it is compiled into a vector-space representation by counting the number of times each context occurs. Headwords are then compared using the distributional hypothesis that similar words appear in similar contexts, i.e. they have similar context vectors. With a context space defined, measuring semantic similarity involves devising a function for measuring the similarity between context vectors that best captures our notion of semantic similarity.

This chapter begins by factoring the existing similarity measures into two components: measures and weights. Section 4.1 defines the notation to describe them. The *measure* functions, which calculate the overall similarity between the two weighted vectors, are described with their motivation in Section 4.2.

The *weight* functions transform the raw counts for each context instance into more comparable values by incorporating a measure of the informativeness of the attribute and its frequency. Intuitively, weight functions model the importance of an attribute. Section 4.3 describes the existing weight functions and in the process motivates an analogy between weight functions and collocation extraction statistics. This insight leads to new weight functions that significantly outperform the state-of-the-art using the evaluation described in Chapter 2.

There are two types of similarity measures: *distance* or *dissimilarity* measures, for instance the L_2 NORM distance, which increase as the distance between the vectors increases; and *similarity* measures, for instance the COSINE measure, which decrease as the distance between the vectors increases. I will use the term *similarity measure* loosely for all similarity functions, whether they are similarity or dissimilarity measures.

There are also other properties that may be important depending on the application; for instance, whether the similarity function is *symmetric*, $\text{sim}(a,b) \equiv \text{sim}(b,a)$, and whether it satisfies the *triangle inequality*, $\text{sim}(a,b) + \text{sim}(b,c) \geq \text{sim}(a,c)$. These properties are important for clustering and search applications which sometimes rely on these assumptions for their correctness. For my evaluation methodology, which only relies on ranking, such properties are not important; whether the function calculates similarity or dissimilarity simply just changes whether ranking must be in ascending or descending order. Other work, e.g. Lee (2001), has used a negative exponential to convert distance measures into similarity measures.

Lee (1999) considers these formal properties in her analysis of several different similarity measures. Lin (1998d) describes and compares several similarity functions. Weeds and Weir (2003) compare the performance of the similarity measures proposed by Lee and Lin in terms of precision and recall. Strehl (2002) gives a detailed comparison of measure functions and their impact on clustering.

Grefenstette (1994) breaks the weight function down into two further factors: a *global* weight g and a *local* weight l. The global weight is a function of the headword and attribute in the relation and involves frequency counts over all extracted contexts. The local weight function is based directly on the context instance frequency. The weight function is constrained to be in the range 0–1. Pantel and Lin (2002a) incorporate a relation frequency-based correction function that can also be considered as a local weight.

The work in this thesis does not explicitly consider separating local and global weight functions. Also, my implementation does not restrict the weight function to the range 0–1, although most of the successful weight functions are restricted to this range. Section 4.4 describes some interesting results when the weight functions are allowed negative values.

Some measure functions are designed to compare frequency distributions, for instance, the information theoretic measures proposed by Lee (1999). In these cases the weight function is either the relative frequency or is a normalisation (to a total probability of one) of some other previously applied weight function.

4.1 Definitions

The context extractor returns a series of context relations with their instance frequencies. These relations can be represented in nested form $(w,(r,w'))$, which distinguishes the attribute, but can easily be flattened to give (w,r,w'). Computationally, nested relations are represented as sparse vectors of attributes and frequencies for each headword.

From this representation we can calculate a large range of values including the headword, attribute and relation frequencies (both token and type). These counts are not the same as the number of times the headword or attribute occurs in the corpus because a single attribute can appear in the overlapping context of several headwords, and a single headword may have several attributes within each context. Also, not every instance of a headword in the corpus will result in context instances being produced by the extractor. Hence the true instance frequency of headwords and attributes is currently lost in the relation extraction process.

I describe the functions evaluated in this chapter using an extension of the notation used by Lin (1998a), where an asterisk indicates a set of values ranging over all existing values of that component of the relation tuple. In this notation, everything is defined in terms of the existence of context instances, that is context relations with a non-zero frequency. The set of attributes for a given headword w on a given corpus is defined as:

$$(w,*,*) \equiv \{(r,w') \mid \exists (w,r,w')\} \tag{4.1}$$

For convenience, I have extended the notation to weighted attribute vectors by defining a generic weighting function for each relation $\mathrm{wgt}(w,r,w')$. This is the place holder for the weight functions described in Section 4.3.

A subscripted asterisk indicates that the variables are bound together:

$$\sum \mathrm{wgt}(w_m,*_r,*_{w'}) \times \mathrm{wgt}(w_n,*_r,*_{w'}) \tag{4.2}$$

which is a notational abbreviation of:

$$\sum_{(r,w') \in (w_m,*,*) \cap (w_n,*,*)} \mathrm{wgt}(w_m,r,w') \times \mathrm{wgt}(w_n,r,w')$$

For frequency counts used in defining weight functions there is a similar notation:

$$f(w,*,*) \equiv \sum_{(r,w') \in (w,*,*)} f(w,r,w') \tag{4.3}$$

$$p(w,*,*) \equiv \frac{f(w,*,*)}{f(*,*,*)} \quad (4.4)$$

$$n(w,*,*) \equiv |(w,*,*)| \quad (4.5)$$

$$N_w \equiv |\{w\,|n(w,*,*) > 0\}| \quad (4.6)$$

Here $f(w,*,*)$ is the total *instance* or *token* frequency of the contexts that w appears in; $n(w,*,*)$ is the total *type* frequency, i.e. the number of attributes that w appears with. Using this notation, we can define the token and type frequency of each context, headword and attribute, and within each attribute the word and relation type frequencies. These values represent all that is available from the relation extraction output by simple counting. All of the measure and weight functions are defined in terms of these fundamental values.

4.2 Measures

Measure functions perform the high-level comparison of weighted vector-space representations of each headword. Table 4.1 lists the measure functions which are described below and evaluated in Section 4.4. These measure functions cover several different types including simple distance metrics like L-norms (Manhattan and Euclidean distance), Information Retrieval inspired set measures, weighted versions of these developed by Grefenstette (1994) and others, other measures used in the literature and finally distributional methods which compare the relative frequency distributions based on information theoretic principles. I have also created my own extensions to the set based measures using similar principles to Grefenstette (described in Section 4.2.3). Alternative generalisations are marked with a dagger. An extensive but slightly dated study of distance measures is given in Anderberg (1973).

4.2.1 Geometric Distances

The L_1, L_2 and the L_∞ norms (also called *Minkowski* distances) are well known measures of distance derived from a coordinate geometry perspective of distance. The norm number n indicates the power in the following general form:

$$L_n(w_1,w_2) = \sqrt[n]{\sum(\text{wgt}(w_1,*_r,*_{w'}) - \text{wgt}(w_2,*_r,*_{w'}))^n} \quad (4.7)$$

The L_1 norm is also called the *Manhattan* or *Levenshtein* distance:

$$L_1(w_1,w_2) = \sum|\text{wgt}(w_1,*_r,*_{w'}) - \text{wgt}(w_2,*_r,*_{w'})| \quad (4.8)$$

L$_1$ NORM	$\sum \lvert \text{wgt}(w_1,*_r,*'_w) - \text{wgt}(w_2,*_r,*'_w) \rvert$	L$_2$ NORM	$\sqrt{\sum (\text{wgt}(w_1,*_r,*'_w) - \text{wgt}(w_2,*_r,*'_w))^2}$
SETCOSINE	$\frac{\lvert (w_1,*,*) \cap (w_2,*,*) \rvert}{\sqrt{\lvert (w_1,*,*) \rvert \times \lvert (w_2,*,*) \rvert}}$	COSINE	$\frac{\sum \text{wgt}(w_1,*_r,*_{w'}) \times \text{wgt}(w_2,*_r,*_{w'})}{\sqrt{\sum \text{wgt}(w_1,*,*)^2 \times \sum \text{wgt}(w_2,*,*)^2}}$
SETDICE	$\frac{2\lvert (w_1,*,*) \cap (w_2,*,*) \rvert}{\lvert (w_1,*,*) \rvert + \lvert (w_2,*,*) \rvert}$	DICE	$\frac{\sum \text{wgt}(w_1,*_r,*_{w'}) \times \text{wgt}(w_2,*_r,*_{w'})}{\sum \text{wgt}(w_1,*_r,*_{w'}) + \text{wgt}(w_2,*_r,*_{w'})}$
DICE†	$\frac{2\sum \min(\text{wgt}(w_1,*_r,*_{w'}), \text{wgt}(w_2,*_r,*_{w'}))}{\sum \text{wgt}(w_1,*_r,*_{w'}) + \text{wgt}(w_2,*_r,*_{w'})}$	SETJACCARD	$\frac{\lvert (w_1,*,*) \cap (w_2,*,*) \rvert}{\lvert (w_1,*,*) \cup (w_2,*,*) \rvert}$
JACCARD	$\frac{\sum \min(\text{wgt}(w_1,*_r,*_{w'}), \text{wgt}(w_2,*_r,*_{w'}))}{\sum \max(\text{wgt}(w_1,*_r,*_{w'}), \text{wgt}(w_2,*_r,*_{w'}))}$	JACCARD†	$\frac{\sum \text{wgt}(w_1,*_r,*_{w'}) \times \text{wgt}(w_2,*_r,*_{w'})}{\sum \text{wgt}(w_1,*_r,*_{w'}) + \text{wgt}(w_2,*_r,*_{w'})}$
LIN	$\frac{\sum \text{wgt}(w_1,*_r,*_{w'}) + \text{wgt}(w_2,*_r,*_{w'})}{\sum \text{wgt}(w_1,*,*) + \sum \text{wgt}(w_2,*,*)}$	α-SKEW	*see Section 4.2.5*
JS-DIV	*see Section 4.2.5*		

Table 4.1: Measure functions evaluated

and measures the component-wise absolute difference between two vectors. Lee (1999) quotes a bounding relationship between the L_1 norm and the KL-divergence (see Section 4.2.5). The L_2 norm is also called the *Euclidean* distance:

$$L_2(w_1,w_2) = \|\text{wgt}(w_1,*_r,*_{w'}) - \text{wgt}(w_2,*_r,*_{w'})\| \quad (4.9)$$

$$= \sqrt{\sum (\text{wgt}(w_1,*_r,*_{w'}) - \text{wgt}(w_2,*_r,*_{w'}))^2} \quad (4.10)$$

Lee (1999) quotes Kaufman and Rousseeuw (1990) who suggest that the L_2 norm is extremely sensitive to the effects of outliers in the vector and prefer the L_1 norm. The final norm is the L_∞ norm, which is equivalent to taking the maximum distance between the corresponding relation weights of the two terms.

Finally, many methods simply combine the weights of the corresponding context relations. This is particularly common with mutual information weighted scores, for example, Hindle (1990) and Luk (1995) (see Section 4.3.4).

4.2.2 Information Retrieval

The measure functions prefixed with SET- in Table 4.1 use the set theoretic model from early experiments in IR (van Rijsbergen, 1979). These measures include the *Dice*, *Jaccard*, *cosine* and *overlap* measures which are summarised in Manning and Schütze (1999, page 299). These methods have been extended to incorporate weightings for each set member. The *cosine* measure, originally taken from linear algebra, extends naturally to weighted vectors and has become the standard measure for weighted vectors in IR.

The *overlap* measure counts the number of attributes the two headwords have in common as a fraction of the number of attributes in the smaller headword, i.e. the one with fewer attributes by type. For objects obj_m and obj_n the overlap measure is:

$$\frac{2|(w_1,*,*) \cap (w_2,*,*)|}{min(|(w_1,*,*)|,|(w_2,*,*)|)} \tag{4.11}$$

The *Dice* measure (Dice, 1945) is twice the ratio between the number of shared attributes and the total number of attributes for each headword, i.e. including the common attributes twice. The constant ensures the function ranges between 0 and 1. Dice has been used in many NLP and IR applications including compiling multi-word translation lexicons (Smadja et al., 1996).

$$\frac{2|(w_1,*,*) \cap (w_2,*,*)|}{|(w_1,*,*)| + |(w_2,*,*)|} \tag{4.12}$$

The *Jaccard* measure, also called the *Tanimoto* measure (Tanimoto, 1958), compares the number of common attributes with the number of unique attributes for a pair of headwords:

$$\frac{|(w_1,*,*) \cap (w_2,*,*)|}{|(w_1,*,*) \cup (w_2,*,*)|} \tag{4.13}$$

Grefenstette (1994) uses a weighted generalisation of Jaccard (Section 4.2.3).

Witten et al. (1999) motivate the use of the *cosine* measure in IR over the dot product and the L_1 and higher order norms. The *dot* or *inner product* of two document vectors developed naturally from generalising coordinate wise matching. Unfortunately it does not account for the length of each vector, so it always favours longer vectors. However, the norm distances (Section 4.2.1) discriminate too strongly against vectors with significantly different lengths, such as documents and queries in an IR context, or common and rare words in NLP. The cosine measure overcomes these problems by considering the difference in *direction* of two vectors in context space as opposed to the distance. This has a well understood geometric interpretation starting from the inner product between two vectors $\vec{w}_1$ and $\vec{w}_2$:

$$\vec{w}_1 \cdot \vec{w}_2 = ||\vec{w}_1||\,||\vec{w}_2|| \cos\theta \tag{4.14}$$

which can be transformed giving the angle (the cosine of the angle) between the two vectors:

$$\cos\theta = \frac{\vec{w}_1 \cdot \vec{w}_2}{||\vec{w}_1||\,||\vec{w}_2||} \tag{4.15}$$

$$= \frac{\sum \text{wgt}(w_1,*_r,*_{w'})\,\text{wgt}(w_2,*_r,*_{w'})}{\sqrt{\sum \text{wgt}(w_1,*,*)^2 \sum \text{wgt}(w_2,*,*)^2}} \tag{4.16}$$

4.2.3 Set Generalisations

Grefenstette (1994) generalises the Jaccard similarity measure to non-binary value (fuzzy sets) semantics, by relaxing the binary membership test, so that each attribute is represented by a real value in the range 0–1. This means that intersection, union and set cardinality must be reformulated. Grefenstette's generalisation replaces intersection with the minimum weight, and union with a maximum weight. Set cardinality is generalised to summing over the union of the attributes of the headwords:

$$\frac{\sum \min(\text{wgt}(w_1,*,*),\text{wgt}(w_2,*,*))}{\sum \max(\text{wgt}(w_1,*,*),\text{wgt}(w_2,*,*))} \tag{4.17}$$

By constraining the weights to either 0 or 1, it is clear that the weighted measure reduces to the binary Jaccard measure. There are also alternative generalisations for Jaccard and other set measures. For example, the overlap metric can be generalised as the sum of the maximum of weights for w_1 and w_2 on the numerator, and the sum of the minimum weights on the denominator. The Dice measure can be extended in a similar way. It turns out that the generalisation for Dice and Jaccard can be equivalent depending on the method under consideration. Table 4.1 shows the different generalisations used in this thesis for Jaccard and Dice. Alternate generalisations are marked with a dagger. There are other possible generalisations of the Jaccard function that I have not considered here (e.g., Strehl, 2002, page 94). Dagan et al. (1993) use a form of Jaccard which separates the left and right contexts which is equivalent to using a window extractor with the relation type equal to left or right.

4.2.4 Information Theory

Lin (1998d) proposes his own similarity metric based on three intuitions:

- The similarity between objects A and B is related to what they have in common (called their *commonality*). The more commonality they share the more similar they are.
- The similarity between objects A and B is inversely related to the differences between them. The more differences they have, the less similar they are.
- The maximum similarity between objects A and B should only be reached when they are identical, no matter how much commonality they share.

He then presents a series of information theoretic (and other) assumptions to constrain his definition of a similarity measure. The information theory used is the information measure

$I(X)$ for an event X, defined as the negative log probability (Cover and Thomas, 1991):

$$I(X) = -\log P(X) \tag{4.18}$$

The 6 assumptions which define Lin's similarity measure are:

Assumption 1: Commonality is defined as $I(\text{common}(A,B))$ where $common(A,B)$ is a common proposition or event. In our case, $\text{common}(A,B)$ refers to common attributes between headwords A and B.

Assumption 2: Difference is defined as $I(\text{desc}(A,B)) - I(\text{common}(A,B))$ where $desc(A,B)$ is a proposition that describes "what A and B are". In our case, $\text{desc}(A,B)$ is the attributes representing each headword.

Assumption 3: Similarity is only a function of commonality and difference.

Assumption 4: The similarity of identical objects is one.

Assumption 5: The similarity of objects with no commonality is zero.

Assumption 6: Overall similarity between objects is a weighted average of their similarity from "different" perspectives. For instance, if there are two sources of features, similarity should be calculated by combining the individual similarities using a weighted average.

Using these assumptions Lin derives the equation of similarity as:

$$\text{sim}(A,B) = \frac{\log P(\text{common}(A,B))}{\log P(\text{desc}(A,B))} \tag{4.19}$$

Lin (1998d) then goes on to use this similarity measure for three tasks: similarity between ordered ordinal values based on their distribution; string similarity (compared with edit distance and trigram similarity); and for word similarity using grammatical relations from MINIPAR (Section 3.3.4) using the equation:

$$sim(w_1,w_2) = \frac{2I(F(w_1) \cap F(w_2))}{I(F(w_1)) + I(F(w_2))} \tag{4.20}$$

where $F(w)$ returns the set of features (which I call *attributes*) for the headword w.

Any generalisation of the intersection here will lose any extra information about the joint probability of the feature set. However, in this context, words w_1 and w_2 are assumed to be independent, so there is no such information. By factoring out the information measure, we

are left with a function similar to a partially generalised Dice. Again, there is the problem of interpreting the intersection:

$$sim(w_1, w_2) = \frac{2\,\mathrm{wgt}(F(w_1) \cap F(w_2))}{\mathrm{wgt}(F(w_1)) + \mathrm{wgt}(F(w_2))} \tag{4.21}$$

Taking the product of the two weight functions will lead to the generalised Dice measure. An alternative that I consider here, as does Lin (1998a), is to consider the sum of the two weights, and remove the constant so the assumptions are still satisfied:

$$\frac{\sum \mathrm{wgt}(w_1, *_r, *_{w'}) + \mathrm{wgt}(w_2, *_r, *_{w'})}{\sum \mathrm{wgt}(w_1, *, *) + \sum \mathrm{wgt}(w_2, *, *)} \tag{4.22}$$

4.2.5 Distributional Measures

Pereira et al. (1993) consider the task of vector-based similarity as one of comparing the conditional distributions of the headwords $p = P(*|w_1)$ and $q = P(*|w_2)$. Their approach uses information theoretic measures of *distributional similarity* as measures of semantic similarity. The $P(*|w)$ distributions are either estimated directly as relative frequencies $\frac{f(x,w)}{f(x)}$ or after smoothing has been applied to the raw counts.

The basis of these distributional measures is the *Kullback-Leibler divergence* (KL-divergence) or *relative entropy* (Cover and Thomas, 1991, page 18):

$$D(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \tag{4.23}$$

where $0 \log 0$ is defined to be zero using limiting arguments. The KL-divergence can be interpreted as the expected value of the *loss of information* $I(q(x)) - I(p(x))$ of modelling the source distribution $p(x)$ with another distribution $q(x)$. In this sense, if one distribution can be used to encode another without much loss of coding efficiency they are similar.

The KL-divergence is non-negative and equal to zero iff $p(x) \equiv q(x)\,\forall x$. It is not symmetrical (i.e. $D(p||q) \not\equiv D(q||p)$), but this is not a major difficulty since we can easily take the sum of both KL-divergences (which Kullback called the *divergence*). Lee (1997) gives several different motivations for the use of the KL-divergence as a measure of distributional similarity.

Although the KL-divergence has many theoretical benefits, it is hard to implement in practice because it is undefined for the case where $q(x) = 0$ and $p(x) \neq 0$. For semantic similarity, the distributions are very sparse making this is a significant problem. There are two alternatives: use of smoothing on the distributions $p(x)$ and $q(x)$ or modify the divergence in some way to

handle this problem. Lee (1997) considers both approaches, using a back-off smoothing, with weight $\alpha(x)$, to the marginal $p(y)$ for the KL-divergence:

$$p_{\mathrm{BO}}(y|x) = \begin{cases} \frac{f(x,y)}{f(x)} & f(x,y) > 0 \\ \alpha(x)p(y) & otherwise \end{cases} \tag{4.24}$$

A significant disadvantage of this approach is that the calculation becomes very expensive because the zeros can no longer be ignored (p.c. Stephen Clark). I will not consider using back-off, but instead use the modifications to the KL-divergence that Lee (1997, 1999) proposes.

The first of these is the *total divergence to the mean*, also called the *Jensen-Shannon* (JS) divergence, which involves comparing both distributions to the mean of the two distributions:

$$A(p,q) = D(p||\frac{p+q}{2}) + D(q||\frac{p+q}{2}) \tag{4.25}$$

This overcomes the problem of zeros in either the p or q distribution and at the same time makes the measure symmetrical. Also, $A(p,q)$ still maintains the property that only identical distributions have a score of zero. Lee (1999) gives an algebraic manipulation of 4.25 which only requires calculation over the shared attributes, giving some performance improvement over the naïve approach. She also demonstrates that $A(p,q)$ has a maximum value of $2\log 2$. Lee (1997) compares divergence with other measures graphically suggesting that they are less susceptible to sampling error because their values deviate less for small changes in the parameters.

An alternative to the JS-divergence is to only add a weighted amount of the second distribution to the first, which leads to the *α-skew divergence* (Lee, 1999):

$$s_\alpha(p,q) = D(p||\alpha p + (1-\alpha)q) \tag{4.26}$$

For $\alpha = 1$ the α-skew divergence is the KL-divergence and for $\alpha = \frac{1}{2}$ the α-skew divergence is twice the JS-divergence. Commonly used values for α are 0.1 and 0.01.

4.3 Weights

The context relation weight function is designed to assign higher value to contexts that are more indicative of the meaning of that word. These weight functions can incorporate frequency counts for any component(s) of the relation tuple. Table 4.2 lists the weight functions considered in this thesis. The weight functions include simple frequency functions; approaches from information retrieval; and from existing systems (Grefenstette, 1994; Lin, 1998a,d).

IDENTITY	1.0	FREQ	$f(w,r,w')$
RELFREQ	$\frac{f(w,r,w')}{f(w,*,*)}$	TF-IDF	$\frac{f(w,r,w')}{n(*,r,w')}$
TF-IDF†	$\frac{\log_2(f(w,r,w')+1)}{\log_2(1+\frac{N_{(r,w')}}{n(*,r,w')})}$	GREF94	$\frac{\log_2(f(w,r,w')+1)}{\log_2(n(*,r,w')+1)}$
CHI2	*see Section 4.3.5*	LR	*see Section 4.3.5*
LIN98A	$\log(\frac{f(w,r,w')f(*,r,*)}{f(*,r,w')f(w,r,*)})$	LIN98B	$-\log(\frac{n(*,r,w')}{N_w})$
DICE	$\frac{2p(w,r,w')}{p(w,*,*)+p(*,r,w')}$		
MI	$\log(\frac{p(w,r,w')}{p(w,*,*)p(*,r,w')})$	TTEST	$\frac{p(w,r,w')-p(*,r,w')p(w,*,*)}{\sqrt{p(*,r,w')p(w,*,*)}}$

Table 4.2: Weight functions compared in this thesis

My proposed weight functions are motivated by the intuition that highly predictive attributes are strong collocations with their headwords. This in itself is not a new concept, *mutual information* having been successfully used as a weighting function by a number of systems in the past (Hindle, 1990; Lin, 1998c; Luk, 1995). However, this is the first research to connect weighting with collocational strength and test various weight functions systematically. I have implemented most of the approaches in the *Collocations* chapter of Manning and Schütze (1999), including the t-test, χ^2-test and likelihood ratio.

I have also experimented with limiting these functions to a positive range, which has been used in the past (Hindle, 1990; Lin, 1998c), and adding extra frequency weighting. In the weight function naming convention, the $^{\pm}$ suffix indicates an unrestricted range and the LOG suffix indicates that an extra $\log_2(f(w,r,w')+1)$ factor has been added to promote the influence of higher frequency attributes. The † suffix indicates an alternative formula.

4.3.1 Simple Functions

The simple weight functions include using the value 1 if a relation exists regardless of its frequency and zero otherwise (IDENTITY); using the raw frequency directly (FREQ) or using the relative frequency (RELFREQ). The distributional methods, such as the α-skew divergence, are only properly defined with the relative frequency weight function. However, it is possible to consider alternative weight functions by renormalising the vector after applying the alternative weight function.

4.3.2 Information Retrieval

The standard IR term weighting functions are based on the term frequency-inverse document frequency (TF-IDF) principle. The *term frequency* is the number of times the term appears in a particular document, or in our case the context instance frequency $f(w,r,w')$. Large term frequencies indicate the term is representative of the document (in this case, the meaning of the headword). The *document frequency* is the number of documents the term appears in, or in our case the attribute frequency $n(*,r,w')$. Large document frequencies indicate the term does not discriminate well between documents (meanings). For instance, it might be a determiner. TF-IDF balances these two competing factors by taking the ratio.

Witten et al. (1999, pp. 183–185) describe various ways of encoding the TF-IDF principle. The term frequency can be either taken directly $f(w,r,w')$ or using a logarithm to reduce the impact of high frequencies $\log_2(1+f(w,r,w'))$. Here we have followed Grefenstette's convention of adding one to the frequency so that $f(w,r,w')=1$ gives a weight of one after the logarithm. The inverse document frequency can be used directly $\frac{1}{n(*,r,w')}$ or again reduced using a logarithm:

$$\log_2\left(1+\frac{N_w}{n(*,r,w')}\right) \tag{4.27}$$

Witten et al. also describe several other variations for TF-IDF.

4.3.3 Grefenstette's Approach

In developing my implementation of SEXTANT, a number of inconsistencies were discovered between the description in *Explorations in Automatic Thesaurus Discovery* (EATD, Grefenstette, 1994) of the local and global weight functions and the quoted examples and results. With Grefenstette's assistance I was able to identify his original weighting functions. Making a clear distinction between attributes and relations clarifies the weighting function descriptions.

In particular, the global weight function (EATD, page 48) does not satisfy the 0–1 range constraint, does not match the experimental results (EATD, Figure 3.14, page 52) and the results obtained with this formula are not as good as those quoted (EATD, Figure 3.12, page 51) using Grefenstette's original data. With Grefenstette's assistance and access to his original SEXTANT implementation I have inferred the global weight function used. Grefenstette's source code contains the several different formula which could be selected.

The global function, based on the description in EATD with corrections from the source code is an entropy-based global measure (p.c. Grefenstette):

$$g(w,r,w') = 1 + \sum p(w|r,w') \log_2(p(w|r,w')) \tag{4.28}$$

where $p(w|r,w')$ is $\frac{f(w,r,w')}{f(*,r,w')}$. The local weighting function is a log-frequency measure:

$$l(w,r,w') = \log_2(1 + f(w,r,w')) \tag{4.29}$$

These functions resolve some of the inconsistencies in EATD. However, the best performance on my dataset was produced by a different weight function from Grefenstette's source code, which was another variation of TF-IDF:

$$\frac{\log_2(f(w,r,w')+1)}{\log_2(n(*,r,w')+1)} \tag{4.30}$$

4.3.4 Mutual Information

Perhaps the most widely-used weight function in both vector-space similarity and wider NLP tasks is *mutual information* (MI, Fano, 1963), which is often defined in NLP as:

$$I(x,y) \quad = \quad \log \frac{p(x,y)}{p(x)p(y)} \tag{4.31}$$

However this is *pointwise* mutual information, i.e. between two random events $x \in X$ and $y \in Y$. The full definition of mutual information between two random variables X and Y is:

$$I(X;Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)} \tag{4.32}$$

The mutual information can be interpreted as the KL-divergence (defined above) between the joint distribution $p(x,y)$ and the product or independent distribution $p(x)p(y)$.

Church and Hanks (1989, 1990) use the term *association ratio* rather than (pointwise) mutual information because for word tasks with order encoded in the frequency count ($f(x,y) \not\equiv f(y,x)$), the calculation does not satisfy the symmetrical property of information ($I(x,y) \equiv I(y,x)$). The second reason is that Church and Hanks use a window method for extracting frequency counts can so $f(x,y)$ can be larger than $f(x)$ or $f(y)$. They employ mutual information to identify strong collocations. It is now recognised as the standard approach to this task (Manning and Schütze, 1999).

Hindle (1990) uses pointwise mutual information as a weight function between headwords and attributes (in Hindle's case the subject and object relations between the nouns and their verbs) for vector-space similarity experiments:

$$I(w,r,w') = \log_2 \frac{p(w,r,w')}{p(w,*,*)\,p(*,r,w')} \tag{4.33}$$

$$= \log_2 \frac{f(w,r,w')\,f(*,*,*)}{f(w,*,*)\,f(*,r,w')} \tag{4.34}$$

Hindle claims that MI is better than cosine because it is roughly proportional to the number of contexts in common, and better than inner product because it is guaranteed that the noun will be most similar to itself. In fact, Hindle's weighting is a bit more complicated than pointwise mutual information. He uses the smallest absolute mutual information value if the weights are both positive or both negative, otherwise the similarity score for that particular relation is zero. It is common to restrict the range of the mutual information score to non-negative values e.g. Lin (1998d) and Dagan et al. (1993).

Lin (1998a) uses a slightly different calculation of the mutual information for a relation than the earlier work of Hindle, based on different dependence assumptions in the product estimate:

$$I(w,r,w') = \log_2 \frac{p(w,r,w')}{p(w)p(r\,|\,w)p(w'\,|\,w)} \tag{4.35}$$

$$= \log_2 \frac{f(w,r,w')\,f(*,r,*)}{f(w,r,*)\,f(*,r,w')} \tag{4.36}$$

Brown et al. (1992) use mutual information to determine which clusters to merge in their cluster based n-gram language modelling. Dagan et al. (1993) use mutual information for estimating cooccurrence probabilities. Luk (1995) uses mutual information to score cooccurrences in definition concepts as part of a word sense disambiguation system. Turney (2001) uses mutual information with cooccurrence probabilities for selecting the correct word in vocabulary tests.

4.3.5 New Approach

The previous section has shown that a wide range of systems have used mutual information to weight similarity terms by their significance. The success of mutual information in both collocation identification and vector-space similarity suggests there are parallels between these tasks. My hypothesis is that *strong correlates are very informative for semantic similarity* because they occur frequently enough to be reliable and their correlation with specific headwords makes them indicative of the nature of the headword. I have tested this by implementing the

t-test, χ^2-test and likelihood ratio methods described in Manning and Schütze (1999, chap. 5) for extracting collocations.

The t-test and the χ^2-test are standard hypothesis testing techniques. The standard approach is to define a *null hypothesis* that contradicts what we wish to demonstrate and then reject it using the statistical test. For collocation extraction, the *null hypothesis* is that there is no relationship or dependence between the two words, that is the product distribution $p(x,y) = p(x)p(y)$ accurately models the relationship between two words. To reject this we compare the product distribution with the observed joint distribution using a statistical test.

For instance, the t-test compares a value x against a normal distribution defined by its mean μ, sample variance s^2 and sample size N:

$$\tau = \frac{x-\mu}{s}\sqrt{N} \tag{4.37}$$

In the context of calculating association strength within relations, that is between headwords and attributes, this becomes:

$$\frac{p(w,r,w') - p(*,r,w')p(w,*,*)}{\sqrt{p(*,r,w')p(w,*,*)}} \tag{4.38}$$

The χ^2-test for collocation extraction, uses a 2 by 2 *contingency table* which counts the events involving the headword and the attribute. The four cells store the frequency of the headword and attribute cooccurring (O_{wa}), the headword occurring without the attribute ($O_{w\bar{a}}$), the attribute occurring without the headword ($O_{\bar{w}a}$), and neither of them occurring ($O_{\bar{w}\bar{a}}$). For context relation weighting the χ^2-test becomes:

$$\frac{N(O_{wa}O_{\bar{w}\bar{a}} - O_{w\bar{a}}O_{\bar{w}a})^2}{(O_{wa}+O_{w\bar{a}})(O_{wa}+O_{\bar{w}a})(O_{w\bar{a}}+O_{\bar{w}\bar{a}})(O_{\bar{w}a}+O_{\bar{w}\bar{a}})} \tag{4.39}$$

where N is the total number of contexts $f(*,*,*)$ and the contingency cells are:

$$O_{wa} = f(w,r,w') \tag{4.40}$$

$$O_{w\bar{a}} = f(w,*,*) - O_{wa} \tag{4.41}$$

$$O_{\bar{w}a} = p(*,r,w') - O_{wa} \tag{4.42}$$

$$O_{\bar{w}\bar{a}} = N - O_{w\bar{a}} - O_{\bar{w}a} + O_{wa} \tag{4.43}$$

MEASURE	WEIGHT	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –
SETCOSINE	TTEST	1276	14	15	15	0.76
SETDICE	TTEST	1496	63	44	34	1.69
SETJACCARD	TTEST	1458	59	43	34	1.63
COSINE	TTEST	1276	14	15	15	0.76
DICE	TTEST	1536	19	20	20	0.97
DICE†	TTEST	1916	76	52	45	2.10
JACCARD	TTEST	1916	76	52	45	2.10
JACCARD†	TTEST	1745	40	30	28	1.36
LIN	TTEST	1826	60	46	40	1.85
JS-DIV	RELFREQ	1619	66	46	35	1.76
α-SKEW	RELFREQ	1456	51	40	30	1.53

Table 4.3: Evaluation of measure functions

4.4 Results

For computational practicality, I make the simplifying assumption that the performance of measure and weight functions are independent of each other. I have run experiments over a range of measure-weight combinations which suggest that this is a reasonable approximation. Therefore, I have evaluated the weight functions using the DICE† measure, and the measure functions using the TTEST weight because they produced the best results in my previous experiments. The exception to this is the divergence measures, which require the RELFREQ weight.

Table 4.3 presents the results of evaluating the measure functions. The best performance across all measures was shared by JACCARD and DICE†, which produced identical results for the 70 test nouns. DICE† is slightly faster to compute and is to be preferred, although for historical reasons JACCARD has been used in later experiments. The next best system, which performed almost 5% worse on DIRECT, was Lin's measure, also another variant of the DICE-JACCARD measure functions. On the other evaluation measures, particularly the precision measures, DICE† and JACCARD have produced outstanding results. The combination of measuring the common and unique attributes that these measures encode (Lin, 1998d) performs best for semantic similarity experiments. The JS-divergence is the best of the rest, significantly outperforming the remaining measures. Surprisingly, the α-skew divergence performs badly, but this might be improved by experimenting with the value of α.

Table 4.4 presents the results of evaluating the weight functions. Here TTEST significantly outperformed the other weight functions, which supports the intuition that good context relations are strong collocates of the headword. Lin's information theoretic measure LIN98A and Hindle's mutual information MI measure are the next best performing weights, adding further

WEIGHT	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –
IDENTITY	1228	46	34	29	1.33
FREQ	1227	63	38	28	1.51
RELFREQ	1614	64	49	36	1.79
TF-IDF	1509	46	39	33	1.53
TF-IDF†	1228	59	38	29	1.47
GREF94	1258	54	38	29	1.46
LIN98A	1735	73	50	42	1.96
LIN98B	1271	47	34	30	1.37
MI	1736	66	49	42	1.92
CHI2	1623	33	27	26	1.24
DICE	1480	61	45	34	1.70
TTEST	1916	76	52	45	2.10
LR	1510	53	39	32	1.58

Table 4.4: Evaluation of bounded weight functions

WEIGHT	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –
DICELOG	1498	67	45	35	1.73
TTESTLOG	1865	70	49	41	1.99
MILOG	1841	71	52	43	2.05

Table 4.5: Evaluation of frequency logarithm weighted measure functions

WEIGHT	DIRECT COUNT	P(1) %	P(5) %	P(10) %	INVR –
$MI^{\pm}$	1511	59	44	39	1.74
$MILOG^{\pm}$	1566	61	46	41	1.84
$TTEST^{\pm}$	1670	67	50	43	1.96
$TTESTLOG^{\pm}$	1532	63	50	42	1.89

Table 4.6: Evaluation of unbounded weight functions

support for the collocation hypothesis. It is surprising that the other collocation extractors did not perform as well, since TTEST is not popular for collocation extraction because of its behaviour on low frequency counts. Clearly, this behaviour is beneficial for semantic similarity.

The results for the frequency logarithm weighted evaluation functions are shown in Table 4.5. The performance of the DICE and MI weight functions improve with this added frequency weighting which suggest that DICE and MI do not take frequency into account enough. On the other hand, the performance of TTEST is reduced suggesting frequency is already contributing.

One difficulty with weight functions involving logarithms or differences is that the score is sometimes negative which often has a detrimental effect on the overall performance. The results in Table 4.6 show that weight functions that are not bounded below by zero do not

perform as well on thesaurus extraction. However, unbounded weights do produce interesting and unexpected results: they tend to return misspellings of the headword and its synonyms, abbreviations and lower frequency synonyms.

For instance, TTEST$^{\pm}$ returned Co, Co. and PLC for company, but they do not appear in the synonym lists extracted with TTEST. The unbounded weight functions also extracted more hyponyms, for example, corporation names for company, including Kodak and Exxon. Finally unbounded weights tended to promote synonyms from minority senses because frequent senses get demoted by negative weights. For example, TTEST$^{\pm}$ returned writings, painting, fieldwork, essay and masterpiece as the best synonyms for work, whereas TTEST returned study, research, job, activity and life. The TTEST$^{\pm}$ function is negative when the joint probability is less than the expected value from the product distribution (Equation 4.38). These results suggest this occurs more often for more frequent synonyms, and so rare synonyms get a higher relative rank when the weight function is unbounded.

4.5 Summary

This chapter has presented a systematic study of the semantic similarity measures described in the literature. It begins by factoring similarity measures into a weight function that assesses the informativeness of contextual information and a measure function that compares weighted context vectors. It extends notation introduced by Lin (1998a) to conveniently describe the measure and weight functions. The evaluation of a range of measure functions taken from geometry, IR and existing similarity systems in this chapter has shown the DICE† and JACCARD measures to be superior by a significant margin. DICE† is the preferred choice because it is slightly more efficient to compute.

This chapter also proposes new weight functions inspired by the observation that informative attributes are strong collocates with their headwords because strong collocations are relatively frequent and highly correlated. Testing this intuition, I implemented the collocation extraction statistics described in the *Collocations* chapter of Manning and Schütze (1999). The evaluation shows that the TTEST weight function, based on the t-test significantly outperforms every weight functions, from IR and existing similarity systems, including MI. However, good results using MI and the other collocation extraction functions suggests this intuition might be true.

However, the list of measure and weight functions is still incomplete. I intend to add other measure and weight functions, and also test many more weight-measure function combinations.

Chapter 5

Methods

> **method**: **technique** 0.169, **procedure** 0.095, **means** 0.086, **approach** 0.081, **strategy** 0.074, tool 0.071, concept 0.062, **practice** 0.061, formula 0.059, **tactic** 0.059, **technology** 0.058, mechanism 0.058, **form** 0.054, alternative 0.052, **standard** 0.051, **way** 0.050, **guideline** 0.049, **methodology** 0.048, model 0.047, **process** 0.047, ...

This chapter covers three different algorithms and implementation techniques for improving vector-space similarity systems. The first section describes the use of ensembles for improving the quality of similarity results, and corresponds to part of Curran (2002). The second section improves the algorithmic complexity of the naïve nearest-neighbour algorithm, and corresponds to part of Curran and Moens (2002a). The third section describes the large-scale experiments on over 2 billion words of text, using my efficient SEXTANT(MX) implementation with the best performing measure and weighting functions found in the previous two chapters. It also describes the implementation techniques required to perform these large-scale experiments using a parallelized version of the nearest-neighbour algorithm which runs on a Beowulf cluster.

5.1 Ensembles

Ensemble learning is a machine learning technique that combines the output of several different classifiers with the goal of improving classification performance. The classifiers within the ensemble may differ in several ways, such as the learning algorithm or knowledge representation

used, or data they were trained on. Ensemble learning has been successfully applied to numerous NLP tasks, including POS tagging (Brill and Wu, 1998; van Halteren et al., 1998), chunking (Tjong Kim Sang, 2000; Tjong Kim Sang et al., 2000), word sense disambiguation (Pederson, 2000) and statistical parsing (Henderson and Brill, 1999). Dietterich (2000) presents a broad introduction to ensemble methods.

Ensemble methods overcome learner bias by averaging the bias over different systems. For an ensemble to be more effective than its constituents, the individual classifiers must have better than 50% *accuracy* and must produce *diverse* erroneous classifications (Dietterich, 2000). Brill and Wu (1998) call this complementary disagreement *complementarity*. Although ensembles are often effective on problems with small training sets, recent work suggests this may not be true as dataset size increases. Banko and Brill (2001) found that for confusion set disambiguation with corpora larger than 100 million words, the best individual classifiers outperformed ensemble methods.

One limitation of their results is the simplicity of their task and methods used to examine the efficacy of ensemble methods. However, the task was constrained by the ambitious use of one billion words of training material. Disambiguation is relatively simple because confusion sets are rarely larger than four elements. The individual methods must be inexpensive because of the computational burden of the huge training set. They must perform limited processing of the training corpus and can only consider a fairly narrow context surrounding each instance. Finally, because confusion set disambiguation only uses local context, these experiments ignored the majority of the one billion words of text.

This section explores the value of ensemble methods for the more complex task of computing semantic similarity, training on corpora of up to 300 million words. The increased complexity leads to results contradicting Banko and Brill (2001), which are then explored further using ensembles of different contextual complexity. This work emphasises the link between contextual complexity and the problems of representation sparseness and noise as corpus size increases, which in turn impacts on learner bias and ensemble efficacy.

5.1.1 Existing Approaches

Hearst and Grefenstette (1992) have proposed a combination of the results of their respective similarity systems to produce a hyponym hierarchy. Although this is strictly not an ensemble

method it does use the combined information from their two different systems to make a final decision. The results from these two methods are very different, so each system brings a lot of new information to the combination. In particular, Hearst and Grefenstette (1992) find a significant improvement in recall, which is a major problem for hyponym extraction systems.

Turney et al. (2003) combine several different similarity systems including Latent Semantic Analysis (Landauer and Dumais, 1997), pointwise mutual information (Turney, 2001), thesaurus-based similarity and similarity calculated using cooccurrence scores from *Google*. They implement a committee of these approaches using several mixture models which describe how the probability distribution from each system is weighted. In each mixture model, each system is associated with a weight which is optimized on training data using a simple hill-climbing algorithm.

Littman et al. (2002) implement an open architecture for solving crossword puzzles where several independent programs contribute candidate answers for each clue, which are then merged and tested by a central solver. This problem is quite similar to the vocabulary tests used by Turney et al. (2003) for evaluation.

5.1.2 Approach

The experiments in this section have been conducted using ensembles consisting of up to six similarity systems each using a different context extractor described in Chapter 3. The six context extractors are: CASS, MINIPAR, SEXTANT(NB), W($L_{1,2}$), W(L_1R_1) and W(L_1R_1*). The similarity systems use the same TTEST weight and JACCARD measure function. They cover a wide range in performance, as summarised in the top half of Table 5.1. Each ensemble member returns the usual 200 synonyms and scores for each of the 70 headwords in the experimental test set.

I have built ensembles from all six context extractors, labelled with an asterisk in the results (e.g. MEAN(∗)), and the top three performing extractors, MINIPAR, SEXTANT(NB) and W(L_1R_1), (e.g. MEAN(3)), based on the results in Table 5.1.

Ensemble voting methods for this task are interesting because the output of the component systems consists of an ordered set of extracted synonyms rather than a single class label or a probability distribution. To test for subtle ranking effects I have implemented three different methods of combination:

MEAN: the arithmetic mean rank of each headword over the ensemble.

HARMONIC: the harmonic mean rank of each term.

MIXTURE: ranking based on the mean score for each term.

For the arithmetic and harmonic mean voting methods, the system calculates the mean *ranking* for each synonym over all of the ensemble members, and then reranks them using the mean rank. The arithmetic and harmonic means are compared because they behave very differently when the values being combined vary considerably. For the mixture method, the system calculates the mean *score* for each synonym over all of the ensemble members, and then reranks them using the mean score. The individual member scores are not normalised because each extractor uses the same similarity measure and weight function. Ties are arbitrarily broken.

The ensemble assigns a rank of 201 and similarity score of zero to words that did not appear in the list of 200 synonyms returned by each ensemble member. These boundaries for unseen synonyms were chosen to be slightly worse than the rank and score of the last extracted synonym. The values of the boundary parameters could have a considerable impact on the results since they determine how much of an influence words can have that are not returned by all of the ensemble members. However, I have not attempted to experiment with these parameters.

5.1.3 Calculating Disagreement

To measure the complementary disagreement between individual ensemble members, *a* and *b*, I have calculated both Brill and Wu's *complementarity C* and the *Spearman rank-order correlation* R_s (Press et al., 1992) to compare their output:

$$C(A,B) = (1 - \frac{|\text{errors}(A) \cap \text{errors}(B)|}{|\text{errors}(A)|}) * 100\% \quad (5.1)$$

$$R_s(A,B) = \frac{\sum_i (\text{r}(A_i) - \overline{\text{r}(A)})(\text{r}(B_i) - \overline{\text{r}(B)})}{\sqrt{\sum_i (\text{r}(A_i) - \overline{\text{r}(A)})^2} \sqrt{\sum_i (\text{r}(B_i) - \overline{\text{r}(B)})^2}} \quad (5.2)$$

where A and B are synonym lists produced by the two members and $\text{r}(X_s)$ is the rank of synonym s in synonym list X and $\overline{\text{r}(X)}$ is the mean rank of the synonyms in X. The Spearman rank-order correlation coefficient is the linear correlation coefficient between the rankings of elements of A and B. R_s is a useful non-parametric comparison for when the ranking is more

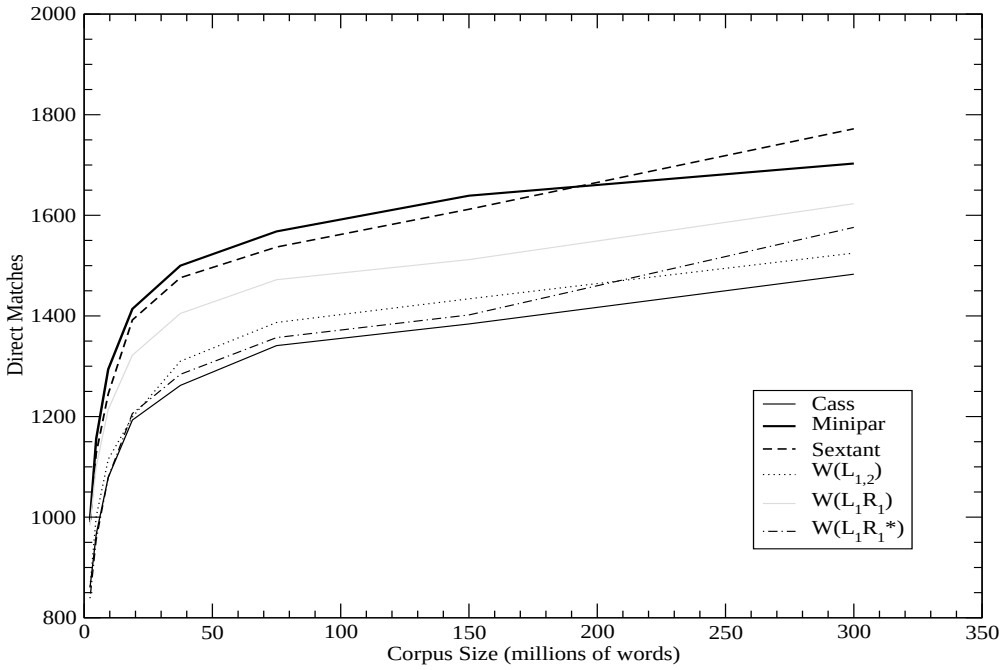

Figure 5.1: Individual performance to 300MW using the DIRECT evaluation

relevant than the scores assigned to the individual items. For the results shown in Table 5.2, the average over all pairs in the ensemble is quoted.

5.1.4 Results

Figure 5.1 shows the performance trends for the individual extractors on corpora ranging from 2.3 million up to 300 million words. The best individual context extractors are SEXTANT(NB), MINIPAR and W(L_1R_1), with SEXTANT(NB) outperforming MINIPAR beyond approximately 200 million words. These three extractors are combined to form the top-three ensemble. CASS and the other window methods perform significantly worse than SEXTANT(NB) and MINIPAR. Interestingly, the window extractor without positional information W(L_1R_1*) performs almost as well as the window extractor with positional information W(L_1R_1) on larger corpora, suggesting that position information is not as useful with large corpora, perhaps because the left and right set of words for each headword becomes relatively disjoint.

Figure 5.2 plots the learning curve over the range of corpus sizes for the best three individual methods and the full ensembles. Although the ensembles clearly dominate the individual ex-

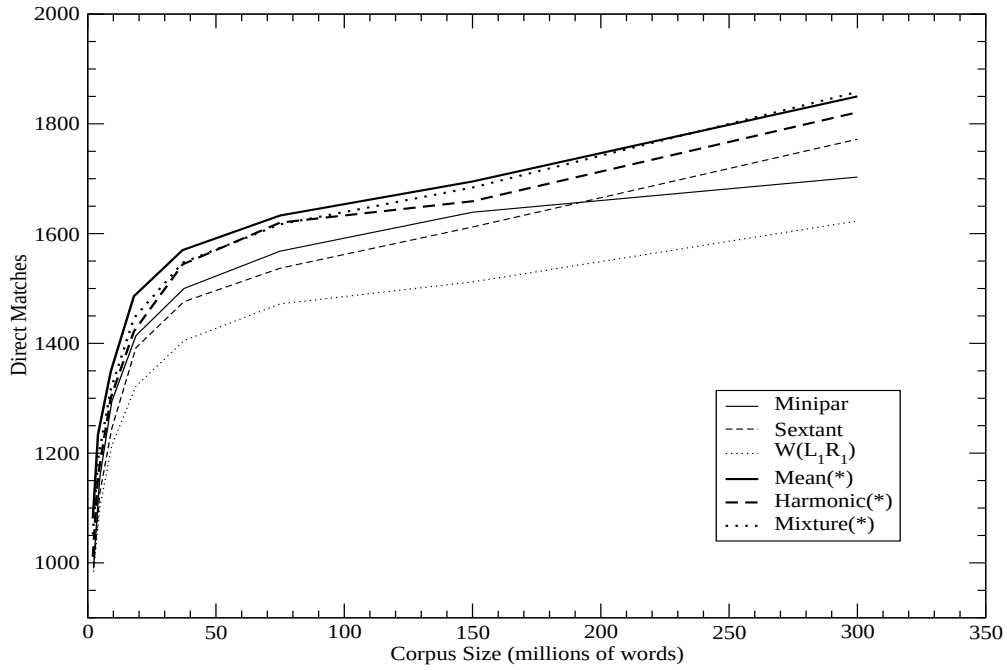

Figure 5.2: Ensemble performance to 300MWs using the DIRECT evaluation

tractors for the entire learning curve, these plots indicate that ensemble methods are of more value, at least in percentage terms, for smaller training sets. If this trend were to continue, then we would eventually expect no benefit from using an ensemble, as suggested by Banko and Brill (2001). However, the trend shown does not give a clear indication either way as to whether the individual extractors will eventually asymptote to the ensemble methods.

Table 5.1 presents the final results for all the individual extractors and the six ensembles on the experimental corpus. At 300 million words, all of the ensemble methods outperform the individual extractors which contradicts the results obtained by Banko and Brill (2001) for confusion set disambiguation. The best performing ensembles, MIXTURE(∗) and MEAN(∗), combine the results from all of the individual extractors. MIXTURE(∗) performs nearly 10% better on the DIRECT evaluation than SEXTANT(NB), the most competitive individual context extractor at 300MWs. Table 5.1 also shows that full ensembles, combining all six individual extractors, outperform ensembles combining only the top three extractors. This seems rather surprising given that the other individual extractors seem to perform significantly worse than the top three.

It is interesting to see how the weaker methods still contribute to ensemble performance. For

System	DIRECT	P(1)	P(5)	P(10)	INVR
CASS	1483	50%	41%	33%	1.58
MINIPAR	1703	59%	48%	40%	1.86
SEXTANT	1772	61%	47%	39%	1.87
W($L_{1,2}$)	1525	54%	43%	37%	1.68
W(L_1R_1)	1623	57%	46%	38%	1.76
W(L_1R_1*)	1576	63%	44%	38%	1.78
MEAN($*$)	1850	66%	50%	43%	2.00
MEAN(3)	1802	63%	50%	44%	1.98
HARMONIC($*$)	1821	64%	51%	43%	2.00
HARMONIC(3)	1796	63%	51%	43%	1.96
MIXTURE($*$)	1858	64%	52%	44%	2.03
MIXTURE(3)	1794	63%	51%	44%	1.99

Table 5.1: Individual and ensemble performance at 300MW

thesaurus extraction, there is no clear concept of *accuracy greater than 50%* since it is not a simple classification task. So, although most of the evaluation results are significantly less than 50%, this does not represent a failure of a necessary condition of ensemble improvement.

Considering the complementarity and rank-order correlation coefficients for the constituents of the different ensembles proves to be more informative. Table 5.2 shows these values for the smallest and largest corpora and Table 5.3 shows the pairwise complementarity for the ensemble constituents. The Spearman rank-order correlation ranges between -1 for strong anti-correlations through to 1 for high correlation. In these experiments, the average Spearman rank-order correlation is not sensitive enough to compare disagreement within our ensembles, because the values are very similar for every ensemble. However, the average complementarity, which is a percentage, clearly shows the convergence of the ensemble members with increasing corpus size, which partially explains the reduced efficacy of ensemble methods for large corpora. Since the top-three ensembles suffer this to a greater degree, they perform significantly worse at 300 million words. Further, the full ensembles can average the individual biases better since they sum over a larger number of ensemble methods with different biases.

To evaluate an ensemble's ability to reduce the data sparseness and noise problems suffered by different context models, I have constructed ensembles based on context extractors with different levels of complexity and constraints. Table 5.4 shows the performance on the experimental corpus for the three syntactic extractors, the top three performing extractors and their corresponding mean rank ensembles. For these more sophisticated context extractors, the en-

Ensemble	R_s	C
Ensemble(∗) on 2.3M words	0.467	69.2%
Ensemble(3) on 2.3M words	0.470	69.8%
Ensemble(∗) on 300M words	0.481	54.1%
Ensemble(3) on 300M words	0.466	51.2%

Table 5.2: Agreement between ensemble members on small and large corpora

System	CASS	MINI	SEXT	W($L_{1,2}$)	W(L_1R_1)	W(L_1R_1*)
CASS	0%	58%	59%	65%	63%	69%
MINI	57%	0%	47%	57%	54%	60%
SEXT	58%	47%	0%	54%	53%	58%
W($L_{1,2}$)	65%	58%	55%	0%	40%	43%
W(L_1R_1)	63%	54%	54%	39%	0%	33%
W(L_1R_1*)	69%	60%	58%	43%	33%	0%

Table 5.3: Pairwise complementarity for extractors

sembles continue to outperform individual learners, since the context representations are still reasonably sparse. The average complementarity is greater than 50%.

Table 5.5 shows the performance on the experimental corpus for a range of window-based extractors and their corresponding mean rank ensembles. Most of the individual learners perform poorly because the extracted contexts are only weakly correlated with the headwords. Although the ensemble performs better than most individuals, they fail to outperform the best individual on DIRECT evaluation. Since the average complementarity for these ensembles is similar to the methods above, we must conclude that it is a result of the individual methods themselves. In this case, the most correlated context extractor, e.g. W(L_1R_1) in the centre ensemble of Ta-

System	DIRECT	P(1)	P(5)	P(10)	INVR
CASS	1483	50%	41%	33%	1.58
MINIPAR	1703	59%	48%	40%	1.86
SEXTANT	1772	61%	47%	39%	1.87
MEAN(P)	1803	60%	48%	42%	1.89
W(L_1R_1)	1623	57%	46%	38%	1.76
MINIPAR	1703	59%	48%	40%	1.86
SEXTANT	1772	61%	47%	39%	1.87
MEAN(3)	1802	63%	50%	44%	1.98

Table 5.4: Complex ensembles perform better than best individuals

System	DIRECT	P(1)	P(5)	P(10)	INVR
W(L_1)	1566	59%	42%	35%	1.70
W(L_2)	1235	44%	36%	31%	1.38
W(R_1)	1198	44%	28%	24%	1.19
W(R_2)	1200	49%	30%	24%	1.25
MEAN($D_{1\|2}$)	1447	54%	46%	37%	1.74
W($L_{1,2}$)	1525	54%	43%	37%	1.68
W(L_1R_1)	1623	57%	46%	38%	1.76
W($R_{1,2}$)	1348	53%	32%	29%	1.40
MEAN($D_{1,2}$)	1550	63%	46%	39%	1.81
W($L_{1,2}*$)	1500	50%	41%	36%	1.60
W(L_1R_1*)	1576	63%	44%	38%	1.78
W($R_{1,2}*$)	1270	46%	29%	27%	1.28
MEAN($D_{1,2}*$)	1499	64%	46%	39%	1.82

Table 5.5: Simple ensembles perform worse than best individuals

ble 5.5, extracts a relatively noise-free representation which performs better than averaging the bias of the other very noisy ensemble members.

5.2 Efficiency

Vector-space approaches to similarity rely heavily on extracting large contextual representations for each headword to minimise data sparseness and noise. This large contextual representation must be extracted from an even larger quantity of raw text. As Chapter 3 demonstrates, performance improves significantly as the corpus size increases. NLP is entering an era where virtually unlimited amounts of text are available, but the computational resources and techniques required to utilise it are not. Under these conditions, Chapter 3 examines the computational trade-offs to be considered between the quality and quantity of contextual information.

However, extracting a large, high-quality contextual representation is not the only computational challenge, because comparing these representations is not scalable. All of the similarity measures can be computed in $O(m)$ steps, where m is the number of attributes for each headword. If there are n headwords and the similarity measure is computed in a pairwise fashion, the total time complexity is $O(n^2m + n^2 \log n)$ which is very expensive.

In Chapter 3, Figure 3.9 shows that the total number of contexts grows almost linearly with

corpus size (which relates to m); Figure 3.10 shows that the number of headwords also increases linearly, but at a much slower rate. However, this does mean the vector-space nearest-neighbour algorithm is effectively cubic in corpus size. Clearly, this expansion needs to be bounded in some way without a significant loss in quality of results.

5.2.1 Existing Approaches

Grefenstette (1994, page 59) stores attribute vectors as linked lists with bit signatures which allow for efficient checks for shared attributes. On Grefenstette's small scale experiments, bit vectors are reasonably effective at reducing the execution time because many vectors do not share any attributes. A fundamental problem is that because attributes are not randomly distributed, some attributes are extremely common, e.g. (obj, get), and may occur with many headwords, making the bit signature ineffective.

Also, as the corpus size increases, the number of attributes increases, and so does the probability of sharing at least one attribute or the bit signatures returning false positives for a much larger number of attributes. The only way to solve the latter problem is to increase the size of the bit signature, which already takes up a considerable space overhead. Unfortunately, memory usage is quite significant, as all of the relations and their frequencies must be kept in memory for every object giving a space complexity of $O(nm)$.

One method that I have already implemented which considerably improves performance involves pre-computing and caching attribute weights for the cases where the attribute does not exist in the other vector. Calculating a similarity measure involves moving along each element of the two attribute vectors. My implementation stores the cumulative sum of the remaining attributes at each element in the vector. So when the shorter sparse vector is exhausted rather than running along summing the remaining elements of the longer vector, the cached cumulative sum can be used. This has quite a significant improvement on performance but at the cost of $O(nm)$ additional memory.

The previous methods reduce the complexity constants by relatively small factors but have no impact on either the vocabulary size n or the number of attributes m. What is required is at least a much larger reduction in the complexity coefficients or even better a reduction or bounding on the factors m and n. One way this can be achieved is to eliminate low frequency headwords because they have very little contextual information. This can significantly reduce

the number of headwords n but this impacts on the recall and precision of the system – usually the recall drops and the precision is increased. Grefenstette (1994) does this by only comparing headwords which have a frequency greater than 10. Some cutoff experimental results are given in Section 5.2.2. Other work, such as (Lee, 1999), only considers the 1000 most frequent common nouns. The experiments on merging morphological variants in Section 3.5.4 are also a form of headword reduction by smoothing rather than filtering.

Clustering (Brown et al., 1992; Pereira et al., 1993) using methods such as *k-means* also reduces the number of similarity comparisons that need to be performed, because each comparison is to a small number of attribute vectors that summarise each cluster. k, the number of clusters, is usually much smaller than the number of headwords n. Algorithms used by Memory-Based Learners (MBL), such as the IGTrees (Daelemans et al., 1997), impose an ordering over the features to efficiently search for a reasonable match. Vector-space models of semantic similarity could be reformulated in terms of IGTrees over a very large number of classes consisting of every headword in the representation.

Another approach is to reduce the number of attributes m, that is, the dimensionality of the context space. Landauer and Dumais (1997) use Latent Semantic Analysis (Deerwester et al., 1990) and Schütze (1992a,b) uses Single Value Decomposition to significantly reduce the dimensionality of context spaces. These methods have the added advantage of combining almost all of the information in the original dimensions in the new smaller dimensions, thereby smoothing as well as reducing the number of dimensions. However, these methods themselves are computationally intensive. For instance, Schütze (1992a,b) only uses the 1000 most frequent words as context because computing the SVD on a larger matrix is very expensive. The same problem is true of LSA as well. This means these methods are important for their smoothing properties, but the dimensionality reduction itself is as expensive as performing the nearest-neighbour comparisons.

There are other methods that fit into this category that have not been used for vector-space semantic similarity such as Principle Component Analysis (PCA). What is needed are dimensionality reduction techniques that do not need to operate on the entire matrix, but instead can make local decisions based on a single attribute vector. In signal processing, these methods include Fourier and Wavelet analysis, but it is not clear how these methods could be applied to our attribute vectors, where there is no connection between the i-th and $(i+1)$-th elements of the attribute vector.

5.2.2 Minimum Cutoffs

Introducing a minimum cutoff that ignores low frequency headwords can eliminate many unnecessary comparisons against potential synonyms with very little informative contextual information. Figure 5.3 presents both the performance of the system using direct match evaluation (left axis) and execution times (right axis) for increasing cutoffs. This test was performed using JACCARD and the TTEST and LIN98A weight functions. The first feature of note is that as the minimum cutoff is increased to 30, the direct match results improve for TTEST, which is probably a result of the TTEST's weakness on low frequency counts.

The trade-off between speed, precision and recall needs to be investigated. Initially, the execution time is rapidly reduced by small increments of the minimum cutoff. This is because Zipf's law (Zipf, 1949) applies to headwords and their relations, and so small increments of the cutoff eliminate many headwords from the tail of the distribution. There are only 29 737 headwords when the cutoff is 30, 88 926 headwords when the cutoff is 5, and 246 067 without a cutoff, and because the extraction algorithm is $O(n^2m)$; this results in significant efficiency gains. Since extracting only 70 headwords takes about 43 minutes with a minimum cutoff of 5, the efficiency/performance trade-off is particularly important from the perspective of implementing a practical extraction system.

Even with a minimum cutoff of 30 as a reasonable compromise between speed and accuracy, extracting a thesaurus for 70 headwords takes approximately 20 minutes. If we want to extract a complete thesaurus for 29 737 headwords left after the cutoff has been applied, it would take approximately one full week of processing. Given that the size of the training corpus is much much larger in the next section, which would increase both the number of attributes for each headword and the total number of headwords above the minimum cutoff, this is not nearly fast enough. The problem is that the time complexity of thesaurus extraction is not practically scalable to significantly larger corpora.

Although the minimum cutoff helps by reducing n to a reasonably small value, it does not constrain m in any way. In fact, using a cutoff increases the average value of m across the headwords because it removes low frequency headwords with few attributes. For instance, the frequent company appears in 11 360 grammatical relations, with a total frequency of 69 240 occurrences, whereas the infrequent pants appears in only 401 relations with a total frequency of 655 occurrences.

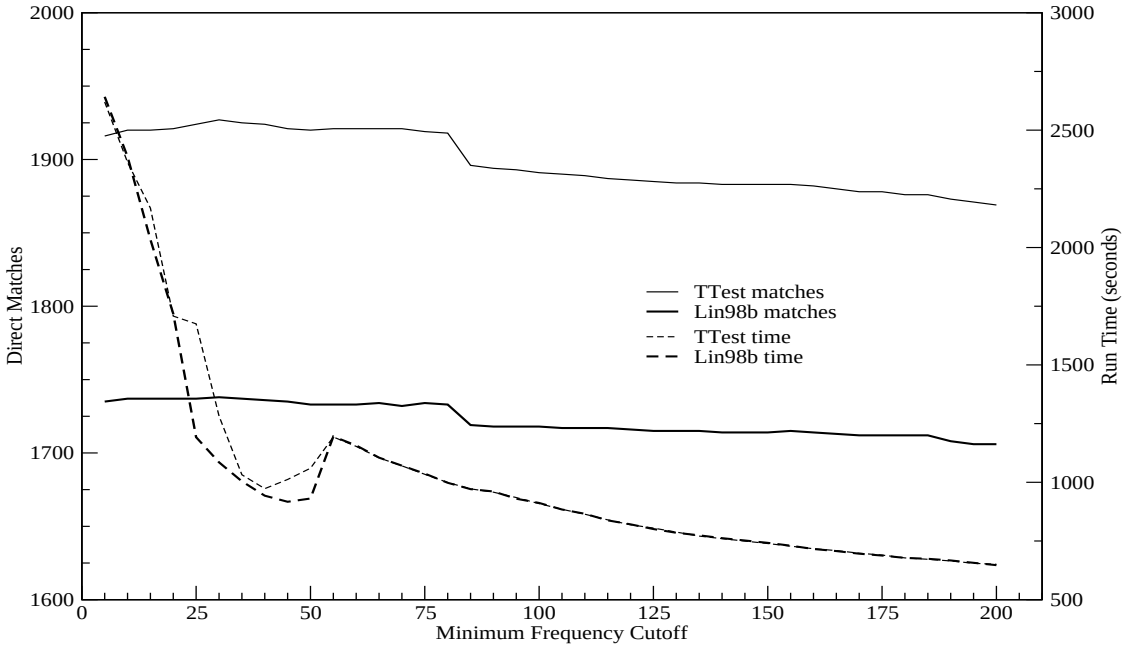

Figure 5.3: Performance and execution time against minimum cutoff

The problem is that for every comparison, the algorithm must examine almost the entire length of both attribute vectors. Grefenstette (1994) uses bit signatures to test for shared attributes, but because of the high frequency of the most common attributes, this does not skip many comparisons. Our system keeps track of the sum of the remaining vector which is a significant optimization, but comes at the cost of increased representation size. However, what is needed is some algorithmic reduction that bounds the number of full $O(m)$ vector comparisons performed.

5.2.3 Canonical Attributes

My approach attempts to deal with the very large vocabulary and very large feature vectors without significant loss of information while at the same time reducing the time complexity of the algorithm so that it can scale to massive text collections.

The first requirement is that any dimensionality reduction or other preprocessing acts on each individual attribute vector without considering other vectors. There are two reasons for this:

- to ensure the time complexity of the preprocessing is not greater than the time taken to actually perform the comparisons;
- to allow easy parallelisation of the algorithm by splitting the headwords across multiple processes. This is needed for the very large-scale experiments in Section 5.3.

One way of bounding the complexity is to perform an approximate comparison first. If the approximation returns a positive result, then the algorithm performs the full comparison. This is done by introducing another, much shorter vector of *canonical attributes*, with a bounded length k. If our approximate comparison returns at most p positive results for each term, then the time complexity becomes $O(n^2k + npm)$, which, since k is constant, is $O(n^2 + npm)$. So as long as the system uses an approximation function and vector such that $p \ll n$, the system will run much faster and be much more scalable in m, the number of attributes. However, $p \ll n$ implies that we are discarding a very large number of potential matches and so there will be a performance penalty. This trade-off is governed by the number of the canonical attributes and how representative they are of the full attribute vector, and thus the headword itself. It is also dependent on the functions used to compare the canonical attribute vectors.

A strong constraint on synonyms is that they usually share key verbs (and sometimes modifiers) that they associate with. For instance, clothing is almost invariably associated with the verb wear. Once the words are grouped into coarse "topic" categories using the verb context relations, the similarity measure can be pairwise computed on each group. If there are k categories the time complexity is greatly reduced to $O(k(\frac{n}{k})^2m + n\frac{n}{k}\log\frac{n}{k})$.

This idea of pre-clustering can be used repeatedly to form a hierarchy of clusters. The headwords in each cluster are then compared using the similarity measure. Another alternative is to compare key verbs first before other relations, and if the key verbs score well, to then compare the rest of the attribute vectors. This could reduce m quite considerably on average and is a similar idea to ordering the attributes in Memory-Based Learning described above.

The canonical vector must contain attributes that best describe the headword in a bounded number of entries. The obvious first choice is the most strongly weighted attributes from the full vector. Figure 5.4 shows some of the most strongly weighted attributes for pants with their frequencies and weights. However, these attributes, although strongly correlated with pants, are in fact too specific and idiomatic to be a good summary, because there are very few other words with similar canonical attributes. For example, (adjective, smarty) only appears with two other headwords (bun and number) in the entire corpus. The heuristic is so aggressive that too

RELATION	COUNT	SCORE
(adjective, smarty)	3	0.0524
(direct-obj, pee)	3	0.0443
(noun-mod, loon)	5	0.0437
(direct-obj, wet)	14	0.0370
(direct-obj, scare)	10	0.0263
(adjective, jogging)	5	0.0246
(indirect-obj, piss)	4	0.0215
(noun-mod, ski)	14	0.0201

Figure 5.4: The top weighted attributes of pants using TTEST

RELATION	COUNT	SCORE
(direct-obj, wet)	14	0.0370
(direct-obj, scare)	10	0.0263
(direct-obj, wear)	17	0.0071
(direct-obj, keep)	7	0.0016
(direct-obj, get)	5	0.0004

Figure 5.5: Canonical attributes for pants

few positive approximate matches result.

To alleviate this problem I have filtered the attributes so that only strongly weighted subject, direct-obj and indirect-obj relations are included in the canonical vectors. This is because in general they constrain the headwords more and partake in fewer idiomatic collocations with the headwords. So the general principle is the most descriptive verb relations constrain the search for possible synonyms, and the other modifiers provide finer grain distinctions used to rank possible synonyms. Figure 5.5 shows the 5 canonical attributes for pants. This canonical vector is a better general description of the headword pants, since similar headwords are likely to appear as the direct object of wear, even though it still contains the idiomatic attributes (direct-obj, wet) and (direct-obj, scare).

One final difficulty this example shows is that attributes like (direct-obj, get) are not informative. We know this because (direct-obj, get) appears with 8769 different headwords, which means the algorithm may perform a large number of unnecessary full comparisons since (direct-obj, get) could be a canonical attribute for many headwords. To avoid this problem a maximum cutoff is applied on the number of headwords the attribute appears with.

With limited experimentation, I have found that TTESTLOG is the best weight function for selecting canonical attributes. This may be because the extra $\log_2(f(w,r,w')+1)$ factor encodes the desired bias towards relatively frequent canonical attributes. If a canonical attribute is

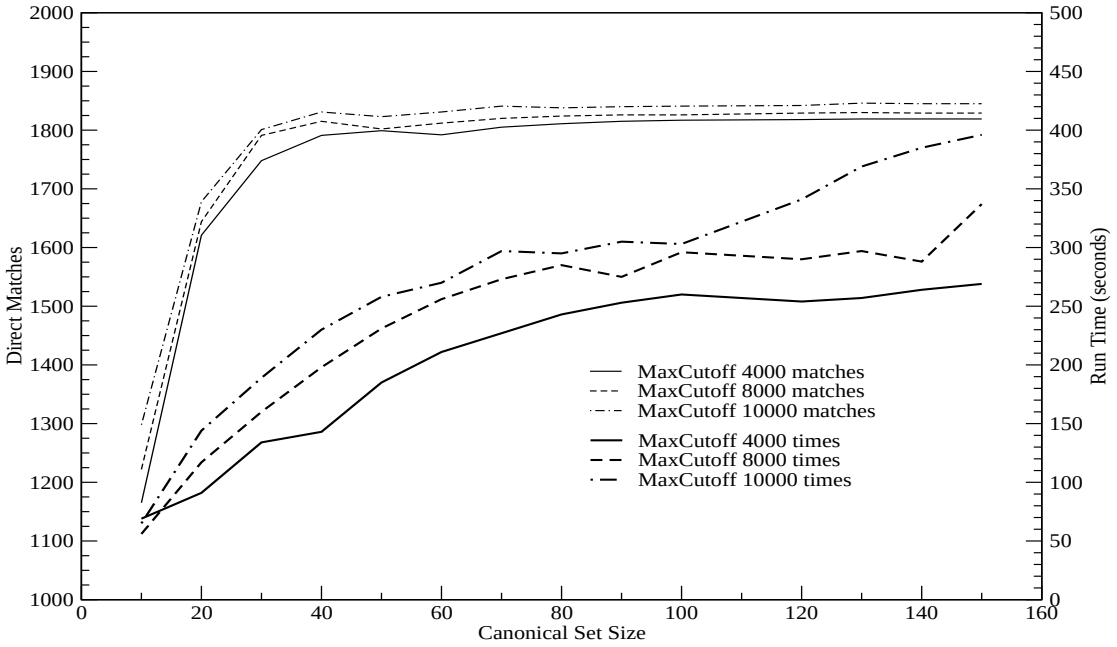

Figure 5.6: Performance against canonical set size

shared by the two headwords, then our algorithm performs the full comparison.

Figure 5.6 shows system performance and speed, as canonical vector size is increased, with the maximum cutoff at 4000, 8000, and 10 000. As an example, with a maximum cutoff of 10 000 and a canonical vector size of 70, the total DIRECT score of 1841 represents a 3.9% performance penalty over full extraction, for an 89% reduction in execution time.

5.3 Large-Scale Experiments

This section describes the very large-scale experiments using the 2 billion word corpus that is introduced in Section 3.2.2. As far as I am aware this is the first work to perform experiments using shallow NLP tools on a corpus of this size. Although there are experiments that have used the web as a corpus, they only estimate cooccurrence relative frequencies using search engine hit counts (Keller et al., 2002; Modjeska et al., 2003). Experiments by Banko and Brill (2001) have used a 1 billion word corpus for ensemble experiments (as described in Section 5.1). However, these experiments only considered selected examples (for confusion sets) from the

STATISTIC	VALUE
number of words in corpus	2 107 784 102
number of sentences in corpus	92 778 662
corpus space	10.6 GB
ALL results	
number of headwords	541 722
number of relations (by type)	52 408 792
number of relations (by token)	553 633 914
number of attributes (by type)	1 567 145
representation space	0.99 GB
CUTOFF(5) results	
number of headwords	68 957
number of relations (by type)	10 516 988
number of relations (by token)	488 548 702
number of attributes (by type)	224 786
representation space	0.19 GB

Table 5.6: Relation statistics over the large-scale corpus

corpus and thus did not process the entire corpus.

I first ran the entire SEXTANT(MX) extraction pipeline: tokenization, tagging, chunking and relation extraction over the entire 2 billion words of text. This is only feasible because each component has been designed to run very quickly and these components use relatively shallow methods. These issues are discussed in Section 3.4. The statistics for the extracted relations are given in Table 5.6. The problem is that the ALL data no longer fits in memory on commodity hardware. And, even if it could fit, it would be extremely slow because of the significantly increased vocabulary size. To overcome this problem I have written a parallelized version of my similarity system which runs on a 64-node Beowulf cluster. The ALL results use the parallel implementation with all of the extracted data on 10 nodes of the cluster.

The CUTOFF(5) results use an attribute frequency cutoff of five or more. This cutoff was chosen to make the dataset fit on a single machine for comparison. Notice that using a cutoff of five dramatically reduces the number of relations and attributes by type, but has a much less severe impact on the number of relations by token. The space required to store the representation is also reduced by a factor of about 5. Finally, the number of headwords is drastically reduced.

5.3.1 Parallel Algorithm

The nearest-neighbour algorithm for finding the most similar synonyms simply iterates over the vocabulary comparing each vector with the vector of the headword in question. Parallelising

this task is relatively straightforward. It involves splitting the vocabulary into separate parts and running the loop in parallel with each part on a separate process. After each process has found the most similar words in its own part of the vocabulary, the results are combined, i.e. the root process sorts the combined results and returns the top *n*. For this reason, high performance computing people would call this task *embarassingly parallel*.

The first question is how to split the words across the machine. The simplest approach would be to take the first *m* words for the first machine and so on, but this suffers from the problem of uneven loading of the task across processes. In such a situation, many processes might be waiting for the most heavily loaded process to catch up and so the efficiency of the parallisation is limited by the unbalanced load.

In our case, the load is unbalanced because the frequency of words (and thus the size of their vectors) is not consistent across the vocabulary. A more complicated approach would be to count the number of headwords in each context vector and try to balance each process so it contains a similar number of words and total length of the vectors. In practice, it suffices to split the words by sending the *n*-th word to the *n*-th process (using the modulus operator), so each machine gets a distribution over the vocabulary and hopefully the word frequency. I have found this results in quite well balanced loads for these experiments.

However, there are more problems than simply splitting the vocabulary. The first problem is that the attribute vector being compared against must be transmitted to each process. For a fixed set of headwords this simply involves including the headwords in the data distributed to each process, but for an online algorithm, the process containing the word must transmit the vector to all other processes before they can start comparing it against their vocabulary part.

The second problem is that even relatively simple weighting functions use properties, such as the attribute frequency, which must be summed over all relations which are distributed between the processes. However, most of these global properties can be calculated once in advance and stored in each process. For instance, storing the total frequency of each attribute is insignificant compared with storing many large context vectors.

5.3.2 Implementation

For my parallel large-scale experiments I have modified my code to run on a large Beowulf cluster using a message-passing approach. This approach involves each node having its own memory (inaccessible to other nodes, as opposed to shared memory parallelism) and communi-

MEASURE	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
ALL	2021	77	69	58	48	38	2.23
CUTOFF(5)	1381	81	71	55	44	36	2.18

Table 5.7: Results from the 2 billion word corpus on the 70 experimental word set

cating and synchronising by sending messages between the nodes. This model of communication matches a cluster configuration very well but can also be efficiently implemented on shared memory machines. This is not true in reverse for shared memory parallelism. The Message Passing Interface MPI library (Gropp et al., 1996) is the emerging standard for implementing message-passing algorithms portably in high performance computing. I have used this library to communicate between the nodes.

The process begins with splitting the data file into node sized chunks and transferring one chunk to each node of the cluster. I currently use 10 nodes, each calculating similarity over representations of approximately 100MB. This runs faster than the usual single CPU runs which typically have representations of around 200MB. I create a separate common data file containing the relations for the test headwords, and another file containing global attribute counts. These files are also distributed to each node. The MPI library then starts processes on each node in the cluster. Each node iterates over the list of words it has for each word in the testfile under control of the root node. Answers are sent back to the root node, which collates and reranks the results to produce the standard output format.

5.3.3 Results

Table 5.7 shows the performance of the two datasets on the 70 word experimental evaluation. The ALL results significantly outperform the best individual context extractor (RASP). However, the CUTOFF(5) results are very poor indicating just how important all of the contextual information is, even those events that only occur 4 times in 2 billion words.

The results for the large 300 word detailed evaluation and also the results on the application-based evaluation are given in the next chapter.

5.4 Summary

This chapter has demonstrated three different algorithms for similarity systems. The first was the application of ensemble learning to similarity systems. The second was a new approximation algorithm for efficient synonym extraction. The third was a parallel implementation of my similarity system which has allowed the calculation of semantic similarity from over 2 billion words of raw text.

This chapter also demonstrated the effectiveness of ensemble methods for synonym extraction and investigates the performance of ensemble extractors on corpora ranging up to 300 million words in size. Contrary to work reported by Banko and Brill (2001), the ensemble methods continue to outperform the best individual systems for very large corpora.

The poorly constrained window methods, where contextual correlation is often low, outperformed the ensembles, which parallels results from Banko and Brill (2001). This suggests that large training sets ameliorate the predominantly noise-induced bias of the best individual learner better than amortising the bias over many similar ensemble constituents. Noise is reduced as occurrence counts stabilise with larger corpora, improving individual classifier performance, which in turn causes ensemble constituents to converge, reducing complementarity. This reduces the efficacy of classifier combination and contributes to individual classifiers outperforming the ensemble methods.

For more complex, constrained methods the same principles apply. Since the correlation between context and target is much stronger, there is less noise in the representation. However, the added constraints reduce the number of contextual relations extracted from each sentence, leading to data sparseness. These factors combine so that ensemble methods continued to outperform the best individual methods.

This chapter has also investigate the speed/performance trade-off using minimum frequency cutoffs to ignore headwords which are unlikely synonyms because of their limited contextual support. This has lead to the proposal of a new approximate comparison algorithm based on *canonical attributes* and a process of coarse- and fine-grained comparisons. This approximation algorithm is dramatically faster than simple pairwise comparison, with only a small performance penalty, which means that complete thesaurus extraction on large corpora is now feasible. Further, the canonical vector parameters allow for control of the speed/performance trade-off. These experiments show that large-scale thesaurus extraction is practical, and al-

though results are not yet comparable with manually constructed thesauri, may now be accurate enough to be useful for some NLP tasks.

Finally, this chapter has introduced a parallel implemention of my similarity system that allows semantic similarity to be calculated using contextual information extracted from over 2 billion words of raw text. The output of this system is analysed in the next chapter using the detailed evaluation and the large evaluation set.

Chapter 6

Results

> **result**: **consequence** 0.065, **outcome** 0.062, **effect** 0.056, **finding** 0.055, evidence 0.054, response 0.048, possibility 0.042, kind 0.041, impact 0.041, datum 0.041, **reason** 0.041, extent 0.041, report 0.040, example 0.040, series 0.040, aspect 0.040, account 0.039, amount 0.038, degree 0.038, basis 0.038, …

This chapter is devoted to analysing the output of the large-scale similarity system and demonstrating its use in an application. The large-scale system and an evaluation on the experimental test set were presented in Section 5.3. The detailed error analysis in Section 6.1 compares performance against a number of variables that were introduced in Section 2.3 on a larger set of words controlled for frequency. These frequency brackets are useful for estimating the performance of the system in practice. The analysis also uses WORDNET to classify the system output according to the types of lexical-semantic relations that are considered to be synonyms by the system. In particular, it is important to know to what degree the system confuses synonyms with antonyms, hyponyms and meronyms.

After describing the analysis of my similarity system, I demonstrate a relatively simple application of semantic similarity. This involves repeating the recent experiments by Ciaramita and Johnson (2003) in categorising previously unseen words using *supersenses* defined in terms of the WORDNET lexicographer files. This task represents the first step in automatically inserting words in WORDNET. My approach significantly improves on the existing work. As mentioned in Chapter 1, automatic methods can either be used directly or as an assistant to a lexicographer to add words to WORDNET.

HEADWORD	DIRECT COUNT	DIRECT MAX	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –	INVR MAX
company	31	378	100	100	60	50	45	2.78	6.51
interest	64	746	100	100	100	80	60	3.69	7.19
problem	30	283	100	100	80	60	30	2.53	6.22
change	32	576	100	100	60	60	50	2.76	6.93
idea	61	454	100	100	100	90	60	3.71	6.70
radio	26	177	100	100	60	40	45	2.20	5.76
star	32	588	100	100	80	50	35	2.62	6.96
knowledge	28	167	100	100	100	80	60	3.24	5.70
pants	12	241	100	100	40	40	20	1.91	6.06
tightness	6	152	0	0	20	10	10	0.61	5.60
Average (over 300)	26	316	68	68	55	45	35	2.08	5.92

Table 6.1: Performance on the 300 word evaluation set

6.1 Analysis

This section presents a detailed analysis of the large-scale results using the 300 word evaluation set (see Appendix A). This evaluation was introduced in Section 2.3. The evaluation methodology used to compare the systems described in the previous three chapters needed to be easy to compare and draw conclusions from. For this reason, a small number of evaluation measures were reported so that a direct comparison was possible. Now that a single set of results is being evaluated it is possible to perform a more detailed analysis.

Table 6.1 presents the results from the large-scale similarity system for some example headwords and the average over the 300 word evaluation set. The two columns labelled MAX give the maximum possible score for the previous column where it is not simply 100%. For example, company could have a maximum DIRECT score of 378 synonyms from the gold-standard. The average performance, at the bottom of Table 6.1, is significantly lower than the average scores in Table 5.7 indicating how much harder the larger test set is. The main reason for this is that it contains a larger portion of lower frequency words, including a number of words that are very rare with less than 5 occurrences per million.

The next two sections analyse the results summarised in Table 6.1 in more detail. Section 6.1.1 considers how performance varies with a number of parameters, including the relative frequency of the word and the amount of contextual information describing it. This is important for estimating the reliability of the system on unseen words. Section 6.1.2 examines the distribution of synonyms and errors across the results using WORDNET. This information is very useful for determining if the system is accurate enough which is often application dependent

REL. FREQ. OPM	NUM	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
TOP 100	111	32.3	67	66	55	46	39	2.20
$50 < f_r \leq 100$	39	29.3	67	62	57	47	37	2.14
$20 < f_r \leq 50$	30	27.9	80	70	62	50	39	2.28
$10 < f_r \leq 20$	30	23.1	70	72	60	47	36	2.14
$5 < f_r \leq 10$	30	20.9	63	62	51	43	31	1.88
$2 < f_r \leq 5$	30	16.4	67	63	52	36	26	1.76
$1 < f_r \leq 2$	15	17.7	73	60	48	43	30	1.85
$0 < f_r \leq 1$	15	18.6	60	60	51	41	31	1.81

Table 6.2: Performance compared with relative frequency of the headword

NUMBER OF ATTRIBUTES	NUM	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
$50\text{k} < n \leq 100\text{k}$	13	35.8	69	65	54	52	45	2.39
$10\text{k} < n \leq 50\text{k}$	177	30.2	71	66	57	47	38	2.20
$5\text{k} < n \leq 10\text{k}$	35	21.9	60	64	51	38	31	1.84
$1\text{k} < n \leq 5\text{k}$	64	18.7	70	67	57	45	31	1.97
$0 < n \leq 1\text{k}$	11	14.8	36	36	35	31	23	1.27

Table 6.3: Performance compared with the number of extracted attributes

(see Section 2.1.5). For instance, confusing antonyms with synonyms may be perfectly acceptable in a smoothing application, where the focus is purely on distributional similarity, but in an information retrieval or extraction context, extracting antonyms could be worse than not retrieving any synonyms.

6.1.1 Performance Breakdown

The evaluation of the results against headword relative frequency, in occurrences per million, is shown in Table 6.2. The second column indicates how many examples there are in each bin. The process of selecting these frequency ranges is described in Section 2.2.2. Although there is a noticeable drop in the DIRECT and INVR measures as the relative frequency decreases, the precision measures remain relatively stable. For instance, the top ranking synonym is still correct 60% of the time when the headword only occurs once every million words, demonstrating that even for quite rare words the system can extract reasonable synonyms. Also, the large drop in the DIRECT and INVR measures is partly caused by rarer words having fewer synonyms in the gold-standards, an effect which has less impact on the precision measures.

Table 6.3 shows the performance against the number of attributes, and Table 6.4 the number of contexts, extracted for each headword from the 2 billion words. These results clearly

NUMBER OF CONTEXTS	NUM	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
$1\text{M} < n \leq 10\text{M}$	60	34.6	70	65	56	49	41	2.30
$100\text{k} < n \leq 1\text{M}$	123	28.6	69	64	56	46	37	2.13
$50\text{k} < n \leq 100\text{k}$	23	23.2	70	74	61	43	33	2.10
$10\text{k} < n \leq 50\text{k}$	51	20.8	75	71	57	42	31	2.01
$5\text{k} < n \leq 10\text{k}$	22	18.9	64	61	51	42	31	1.84
$0 < n \leq 5\text{k}$	21	15.9	48	50	42	39	27	1.56

Table 6.4: Performance compared with the number of extracted contexts

NUMBER OF SENSES	NUM	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
$5 < n \leq 8$	11	49.8	64	73	65	61	52	2.73
$3 < n \leq 5$	52	38.0	69	69	59	48	41	2.36
$2 < n \leq 3$	59	30.5	71	62	54	46	37	2.15
$1 < n \leq 2$	65	25.7	82	75	61	50	37	2.29
$0 < n \leq 1$	113	17.2	59	58	49	39	29	1.74

Table 6.5: Performance compared with polysemy of the headword

demonstrate that to obtain reasonable results a large amount of contextual information must be extracted. There is a very large drop in performance across all measures if the number of attributes drops below 1000 or the number of context instances drops below 5000. The general trend is that the precision measures are reasonably stable above the 1000 attribute threshold, while the DIRECT and INVR measures show a steady decrease in performance with decreasing numbers of attributes and contexts. These tables also show how large the contextual representation becomes for some words, with up to 100 000 attributes summarising up to 10 000 000 context instances extracted from the 2 billion word corpus. Also, a large percentage of the words have over 10 000 attributes.

Table 6.5 shows the evaluation of the results against polysemy, i.e. the number of senses for the headword in WORDNET. The DIRECT and INVR measures tend to increase with increasing numbers of senses while the precision measures are fairly stable. An exception are headwords with one sense which perform significantly worse, and two senses, which perform significantly better. Single sense headwords tend to be rare, while highly ambiguous headwords tend to be frequent, which conflates the trends. Intuitively, fewer senses will produce better results because the context vector only represents one meaning. This is probably why the two sense headwords are performing the best because they have few senses but are not as rare as many of the one sense headwords. More experiments are needed with a large enough test set to control for both number of senses and relative frequency.

WORDNET ROOT	NUM	DIRECT COUNT	P(1) %	P(2) %	P(5) %	P(10) %	P(20) %	INVR –
abstraction	149	30.4	70	64	56	46	37	2.16
activity	85	34.5	68	67	60	50	41	2.32
entity	184	27.4	67	65	54	45	35	2.07
event	33	36.0	82	73	61	52	44	2.49
group	72	32.8	67	69	56	49	40	2.24
phenomenon	25	36.8	72	72	64	49	42	2.43
possession	22	34.0	64	64	54	45	36	2.15
psych. feature	89	36.0	75	74	64	53	43	2.49
state	56	33.3	73	69	57	49	40	2.31

Table 6.6: Performance compared with WORDNET root(s) of the headword

Table 6.6 shows the evaluation of the results against the WORDNET unique beginner(s), or roots, the headwords appears under. When a headword appears under multiple roots its evaluation is added to all of the roots. These results suggest that entity and abstraction words are the hardest to find synonyms for, whilst event, phenomenon and psychological feature words are slightly easier. Semantic properties do not appear to impact heavily on the results, and if there is any influence it is significantly less than the influence of relative frequency.

6.1.2 Error Analysis

In this section I consider the types of errors produced by my similarity system. What constitutes an "error" has been discussed in Section 2.3. Basically, the approach is to extract the top 10 synonyms for a headword and then look at the lexical relations that WORDNET uses to describe the relationship between the headword and the synonym. The WORDNET relations considered are synonym, antonym, hyponym, hypernym, meronym and holonym.

RELATION	SYNONYM
hyponym	subsidiary
hypernym	unit
sister	firm, industry, business
sister	bank, giant, maker
sister	manufacturer

Table 6.7: Lexical-semantic relations from WORDNET for the synonyms of **company**

I have also created another category called sisters which covers words that appear in any other subtree of the parent of the headword. That is, you go up the hierarchy to the parent and then down any other subtree. I also count synonyms which do not appear in WORDNET. The relations found for the top ten synonyms of company are given in Table 6.7.

HEADWORD	SYN	ANT	MER	HOL	HYPO	HYPE	SIS	MISS	ERR
company	0	0	0	0	1	1	8	0	0
interest	2	0	0	0	1	0	4	0	3
problem	1	0	0	0	0	2	0	0	7
change	1	0	0	0	9	0	0	0	0
idea	1	0	0	0	6	1	0	0	2
radio	0	0	0	0	0	1	6	0	3
star	0	0	0	0	0	3	3	1	3
knowledge	0	0	0	0	8	0	0	0	2
pants	1	0	0	0	2	0	6	1	0
tightness	0	0	0	0	0	0	1	0	9
Total (over 300)	177	15	26	34	384	215	673	92	1374
Percentage (over 300)	6	1	1	1	13	7	23	3	46

Table 6.8: Types of errors in the 300 word results

The relation distribution for the top ten synonyms over the 300 word results are shown in Table 6.8. The bottom row of the table gives the percentage distribution over the relations. The error column, ERR, shows the number of synonyms that do not appear in any of the other relations. Approximately 40% of extracted synonyms fall into classes that are reasonably compatible with the headword (every class except antonyms, missing and error). The 6% of WORDNET synonyms found is expected because synonym distinctions are very fine-grained in WORDNET. The largest relation is the sister relation with 23% of the extracted synonyms, which gives an indication of the level of sense distinction which the similarity system is capable of identifying. Also, of the top ten extracted synonyms, approximately 3% do not appear in WORDNET showing that gold-standard coverage is still a significant problem.

6.2 Supersenses

This section presents experiments in classifying previously unseen common nouns with WORDNET *supersenses*. It forms an application-based evaluation of the large-scale similarity system described in the previous chapter and evaluated in the previous section. The task involves repeating the experiments and evaluation used by Ciaramita and Johnson (2003) with a similarity-based approach rather than their classification-based approach.

The supersense tagger implemented by Ciaramita and Johnson (2003) is a multi-class perceptron classifier (Crammer and Singer, 2001), which uses the standard collocation, spelling and syntactic features common in WSD and named entity recognition systems.

WORDNET *supersenses*, as defined by Ciaramita and Johnson (2003), are the broad semantic classes created by lexicographers as the initial step of inserting words into the WORDNET hierarchy. These are called *lexicographer files* or *lexfiles*. For the noun hierarchy, there are 25 lexfiles (and a file containing the list of the top level nodes in the hierarchy called Tops).

Lexfiles form a set of coarse-grained sense distinctions included with WORDNET. For example, the word company appears with the following supersenses in WORDNET 1.7.1: group, which covers company in the social, financial and troupe senses (amongst others), and state, which covers companionship. The names and descriptions of the noun lexfiles, taken from the `lexnames` manpage distributed with WORDNET, are shown in Table 6.9. There are also 15 lexfiles for verbs, 3 for adjectives and 1 for adverbs.

LEXFILE	DESCRIPTION
act	acts or actions
animal	animals
artifact	man-made objects
attribute	attributes of people and objects
body	body parts
cognition	cognitive processes and contents
communication	communicative processes and contents
event	natural events
feeling	feelings and emotions
food	foods and drinks
group	groupings of people or objects
location	spatial position
motive	goals
object	natural objects (not man-made)
person	people
phenomenon	natural phenomena
plant	plants
possession	possession and transfer of possession
process	natural processes
quantity	quantities and units of measure
relation	relations between people or things or ideas
shape	two and three dimensional shapes
substance	substances
time	time and temporal relations

Table 6.9: 25 lexicographer files for nouns in WORDNET 1.7.1

.

Some of these lexfiles map directly to the top level nodes in the noun hierarchy, called *unique beginners*, while others are grouped together as hyponyms of a unique beginner (Fellbaum, 1998, page 30). For example, abstraction subsumes the lexical files attribute, quantity, relation,

communication and time. There are 11 unique beginners in the WORDNET noun hierarchy which could also be used as supersenses. Ciaramita (2002) has produced a mini-WORDNET by manually reducing the WORDNET hierarchy to 106 broad categories. Ciaramita et al. (2003) describe how the lexfiles can be used as root nodes in a two level hierarchy with all of the WORDNET senses appearing directly underneath.

Other alternative sets of supersenses could be created by an arbitrary cut somewhere through the WORDNET hierarchy near the top, or by using topics from a thesaurus such as Roget's or the Macquarie thesaurus. Their topic distinctions are much less fine-grained than WORDNET senses, which have been criticised as being too difficult to distinguish even for experts. Further, Ciaramita et al. (2003) state that most of the key distinctions between senses of a word are still maintained with supersenses.

Supersense tagging can provide automated or semi-automated assistance to lexicographers adding words to the WORDNET hierarchy. Once this task is solved successfully, it may be possible to insert words directly into the fine-grained distinctions of the hierarchy itself. Clearly, this is the ultimate goal, to be able to insert new terms into the existing hierarchy, and extend the hierarchy where necessary.

Supersense tagging is also of interest for the many applications that need coarse-grained senses, for instance information extraction and question answering. Ciaramita and Johnson (2003) suggest that supersense tagging is a similar task to named entity recognition, which also has a very small set of options with similar granularity for labelling previously unseen terms (e.g. location, person and organisation). In these ways, it is also similar to the bootstrapping techniques for learning members of a particular class (Riloff and Shepherd, 1997).

Ciaramita and Johnson (2003) have analysed the BLLIP corpus, finding that common nouns that do not appear in WORDNET 1.6 have a relative frequency of 0.0054, and so on average an unknown word appears every 8 sentences. Clearly it is important to be able to interpret these words in some way.

6.2.1 Previous Work

There has been a considerable amount of work addressing the issue of structurally (and statistically) manipulating the hierarchy of the English WORDNET and the construction of new wordnets using the concept structure from English.

In terms of adding information to the existing English WORDNET, Beeferman (1998) adds several different types of information including phonetic and rhyming similarity from the CMU pronunciation dictionary using edit distance and anagram. More importantly, he also adds collocation pairs extracted from a large corpus (160 million words of broadcast news) using mutual information. The cooccurrence window was 500 words (which was designed to approximate average document length). Over 350 000 trigger pairs were added. The result (and the regular expression language that describes paths over the network) he terms *lexical FreeNet*.

Caraballo and Charniak (1999) have examined the issue of determining the specificity of nouns from raw text. They find that simple frequency counts are the most effective way of determining the parent-child relationship ordering. Raw frequency achieves 83% accuracy over types of vehicle, food and occupation. The other measure Caraballo and Charniak found to be successful was measuring the entropy of the conditional distribution of surrounding words given the noun. This is a necessary step for determining hyponymy relationships between words and building a noun hierarchy (Caraballo, 1999). However, it is clear that these methods cannot extend to abstract types. For instance, entity is less frequent than many concepts it subsumes. This suggests it will only be possible to add words to an existing abstract categories rather than create categories right up to the unique beginners.

Hearst and Schütze (1993) flatten the structure of WORDNET into 726 categories using an algorithm which attempts to minimise the variance between the size of each category. They use these categories to label paragraphs with topics, effectively repeating Yarowsky's (1992) word sense disambiguation experiments using the WORDNET-derived categories rather than Roget's thesaurus. They then use Schütze's (1992a) WordSpace system to add topical links, such as between ball, racquet and game (the *tennis problem*) that are currently not supported by WORDNET. Further, they also use the same context-space techniques to label previously unseen words using the most common class assigned to the top 20 synonyms for that word. These unseen words are common nouns, proper names, and existing words with previously unseen senses.

Widdows (2003) uses a similar technique to Hearst and Schütze to insert words into the WORDNET hierarchy. He first extracts synonyms for the unknown word using vector-space similarity measures with Latent Semantic Analysis and then searches for a location in the hierarchy nearest to these synonyms. The same technique as Hearst and Schütze (1993) and Widdows (2003) is used in my approach to supersense tagging.

There are very few links between the noun and verb hierarchies, topically related words and morphologically related terms in WORDNET. Harabagiu et al. (1999) set out to improve this situation by augmenting WORDNET with links between concepts in the gloss of each synset. This involves first disambiguating the words in the glosses with respect to WORDNET senses for each word. Their motivation for doing this is the construction of a knowledge base for common sense reasoning, in particular for inference in question answering.

Apart from augmenting WORDNET with different kinds of knowledge, there has been considerable work trying to align WORDNET with other lexical resources, including Levin's verb classes (Green et al., 2001), topic categories from Roget's thesaurus (Mandala et al., 1999; Nastase and Szpakowicz, 2001) and information from LDOCE (Kwong, 1998). Stevenson (2002) uses three different similarity metrics to align WORDNET with CIDE+ to augment WORDNET with related term links. Even the Dewey decimal system has been combined with WORDNET (Cavaglià, 1999).

There have also been efforts to augment WORDNET with domain-specific ontologies (O'Sullivan et al., 1995) and also to prune senses and synonym relation links based on evidence from domain-specific corpora (Basili et al., 1998; Turcato et al., 2000). The concept structure of the English WORDNET has been used to automatically create wordnets for other languages with the help of bilingual dictionaries for Catalan (Benítez et al., 1998; Farreres et al., 1998), Spanish (Atserias et al., 1997; Farreres et al., 1998) and Korean (Lee et al., 2000). These approaches must first disambiguate the entries in the MRD against the senses in the English WORDNET core structure.

6.2.2 Evaluation

The supersense tagging task has a very natural evaluation: inserting the extra common nouns that have been added to a new version of WORDNET. Ciaramita and Johnson (2003) use the words that have been added to WORDNET 1.7.1 since the WORDNET 1.6 release. They compare this evaluation with a standard cross-validation approach that uses a small percentage of the words from their WORDNET 1.6 training set for evaluation. Their results suggest that the WORDNET 1.7.1 test set is significantly harder because of the large number of abstract category nouns, for instance, communication and cognition, that appear in the 1.7.1 data, which are rather difficult to classify correctly.

WORDNET 1.6		WORDNET 1.7.1	
NOUN	SUPERSENSE	NOUN	SUPERSENSE
stock index	communication	week	time
fast food	food	buyout	act
bottler	group	insurer	group
subcompact	artifact	partner	person
advancer	person	health	state
cash flow	possession	income	possession
downside	cognition	contender	person
discounter	artifact	cartel	group
trade-off	act	lender	person
billionaire	person	planner	artifact

Figure 6.1: Example nouns and their supersenses

My evaluation will use exactly the same test sets as Ciaramita and Johnson (2003). The WORDNET 1.7.1 test set consists of 744 previously unseen nouns, the majority of which (over 90%) have only one sense. The WORDNET 1.6 test set consists of several cross-validation sets of 755 nouns randomly selected from the BLLIP training set used by Ciaramita and Johnson (2003). Massimiliano Ciaramita has kindly supplied me with the WORDNET 1.7.1 test set and one cross-validation run of the WORDNET 1.6 test set. All of my experiments are performed on the WORDNET 1.6 test set with one final run on the WORDNET 1.7.1 test set. Some examples from the test sets are given in Figure 6.1 with their supersenses.

6.2.3 Approach

My approach uses semantic similarity with a hand-crafted fall-back for unseen words. The similarity approach uses the supersenses of extracted synonyms to choose the correct supersense. This is the approach used by Hearst and Schütze (1993) and Widdows (2003). The synonyms are extracted using the large-scale similarity system described in Section 5.3. This method is used if there is sufficient information about the unknown noun to extract reasonably reliable synonyms. The fall-back method is a simple hand-coded classifier which examines the unknown noun and makes a guess based on morphological analysis. These rules were created by looking at the suffixes of rare nouns in WORDNET 1.6. The supersense guessing rules are given in Table 6.10. If none of the rules match, then the default artifact is assigned.

The problem now becomes how to convert the ranked list of extracted synonyms into a single supersense selection. Each synonym has one or more supersense tags taken from WORDNET 1.6 to match the training data used by Ciaramita and Johnson (2003). For this there are a

SUFFIX	EXAMPLE	SUPERSENSE
-ness	remoteness	attribute
-tion, -ment	annulment	act
-ist, -man	statesman	person
-ing, -ion	bowling	act
-ity	viscosity	attribute
-ics, -ism	electronics	cognition
-ene, -ane, -ine	arsine	substance
-er, -or, -ic, -ee, -an	mariner	person
-gy	entomology	cognition

Table 6.10: Hand-coded rules for supersense guessing

number of parameters to consider:

- how many synonyms to use;
- how to weight each synonym's contribution;
- whether unreliable synonyms should be filtered out;
- how to deal with polysemous synonyms.

The experiments described below consider a range of options for these parameters. In fact, these experiments are so quick to run I have been able to exhaustively test many combinations of these parameters. For the number of synonyms to use I have considered a wide range, from just the top scoring synonym through to 200 extracted synonyms.

There are several ways to weight each synonym's contribution. The simplest approach would be to give each synonym the same weight. Another approach is to use the scores returned by the similarity system. Finally, the weights can use the ranking of the extracted synonyms. Again these options have been considered below. A related question is whether to use all of the extracted synonyms, or perhaps filter out synonyms for which a small amount of contextual information has been extracted, and so might be unreliable.

The final issue is how to deal with polysemy. Does each sense get a single count when it appears or is it distributed evenly between senses like Resnik (1995)? Another alternative is to only consider synonyms with a single supersense in WORDNET.

One disadvantage of the similarity approach is that it requires full synonym extraction, which compares the unknown word against a large number of words when, in fact, we want to calculate the similarity to a small number of supersenses. This inefficiency could be reduced significantly if we consider only very high frequency words, but even this is still expensive.

6.2.4 Results

I have used the WORDNET 1.6 test set to experiment with different parameter settings and have kept the WORDNET 1.7.1 test set as a final comparison of best results with Ciaramita and Johnson (2003). The synonyms were extracted from the 2 billion word corpus using SEXTANT(NB) with the JACCARD measure and TTEST weight function. The experiments were performed by considering all possible combinations of the parameters described below.

The following weighting options were considered for each supersense: the initial weight for a supersense could either be a constant (IDENTITY) or the similarity score (SCORE) calculated by the synonym extractor. The initial weight could then be divided by the number of supersenses to share out the weight (SHARED). The weight could also be divided by the rank (RANK) to penalise supersenses further down the list.

The best performance on the WORDNET 1.6 test set was achieved by using the SCORE weight function, without any sharing or ranking penalties.

Synonyms are filtered before contributing to the vote with their supersense(s). This filtering involves checking that the synonym's frequency and number of attributes are large enough to ensure the synonym is reliable. I have experimented with a wide range of minimum cutoffs for the frequency and number of attributes. The best performance on the WORDNET 1.6 data was achieved by using cutoffs of 5 for both the frequency and the number of attributes. However, many systems performed almost as well with only one of the two cutoffs set.

The next question is how many synonyms are considered and whether that number applies before or after the filtering has occurred. For instance, if this number applies before filtering, then fewer than 50 synonyms may contribute to the supersense vote because they have been filtered out. All of the top performing systems used 50 synonyms applied after the filtering process has occurred.

The final consideration regarding the synonym list is whether highly polysemous nouns should be filtered out. In fact, using a filter which removed synonyms with more than one or two senses turned out to make little difference.

Finally, the decision needs to be made between using the similarity measure or the guessing rules. This is determined by looking at the frequency and number of attributes for the unknown word. Not surprisingly, the similarity system works better than the guessing rules if it has any

information at all, so the frequency and number of attributes cutoffs were 0 or 5 for the top performing systems.

The accuracy of the best performing system(s) was 68.2% with several other combinations of the parameters described above performing almost as well. This accuracy should be compared against the results of Ciaramita and Johnson (2003) who get 53.4% as their best accuracy. On the WORDNET 1.7.1 test set, the performance of the best system (from the above experiments) is 62.8%, which significantly outperforms Ciaramita and Johnson (2003) who get an accuracy of 52.9%.

6.2.5 Future Work

An alternative approach worth exploring is to create vectors for the supersense categories themselves. This has the advantage of producing a much smaller number of vectors, but the question then becomes how to construct such vectors. One solution would be to take the intersection between vectors that fit into a particular class (i.e. to find the common contexts that these words appear in). However, given the sparseness of the data this would not leave very large vectors. Another solution is to sum the vectors but this could potentially produce very large vectors which may not match well against the smaller vectors. A final solution would be to consider a large set of the canonical attributes defined in the previous chapter for approximate matching.

Also, I would like to move onto the more difficult task of insertion into the hierarchy itself and compare against the results from Widdows (2003) using latent semantic indexing. Here the issue of how to combine vectors is even more interesting since there is the additional structure of the WORDNET inheritance hierarchy and the small synonym sets that can be used for more fine-grained combination of vectors.

6.3 Summary

This chapter has analysed the results of the large-scale system described in the previous chapter and applied these results to the task of supersense tagging.

The analysis showed that for the DIRECT and INVR measures there is a strong dependence between relative frequency of the headword and synonym quality. On the precision measures the dependence was not as significant. Similar results were found for the number of extracted

contexts and attributes. An important result was identifying a minimum number of extracted contexts and attributes required to achieve reasonable results. The analysis also looked at performance against semantic properties such as the number of senses of the headword and the broad semantic category it belonged to.

The application of semantic similarity to supersense tagging follows similar work by Hearst and Schütze (1993) and Widdows (2003). To classify a previously unseen word my approach extracts synonyms and uses their supersenses as an indication of the supersense of the unseen word. Using this approach I have significantly outperformed the existing work of Ciaramita and Johnson (2003).

Chapter 7

Conclusion

> **conclusion**: finding 0.067, outcome 0.057, **interpretation** 0.044, **assertion** 0.043, assessment 0.043, **explanation** 0.041, **judgment** 0.039, **assumption** 0.039, **decision** 0.039, recommendation 0.037, **verdict** 0.037, **completion** 0.036, **inference** 0.036, suggestion 0.036, **result** 0.035, answer 0.035, **view** 0.035, comment 0.034, testimony 0.034, argument 0.034, ...

This thesis explores a statistical approach to lexical semantics, in particular, the measurement of semantic similarity, or synonymy, using vector-space models of context.

I have presented an exhaustive systematic analysis of existing vector-space approaches using a methodology designed to separate synonymy from other related properties. This analysis has inspired new measures of similarity that emphasise *semantic* rather than *distributional* similarity, which results in a significant improvement over the state-of-the-art.

I have also developed techniques for improving similarity calculations. The first is an ensemble of learners approach which improves performance over individual methods. The second is a novel approximation algorithm which bounds the time complexity of the nearest-neighbour calculations making this approach feasible for large collections. The third is a parallel implementation of my similarity system using message-passing which allows the calculates similarity on a Beowulf cluster. This large-scale system is used to compute similarity over a 2 billion word corpus, currently the largest quantity of text to be analysed using shallow statistical techniques.

A final experiment involved applying this large-scale system to supersense tagging, assigning broad semantic classes taken from the WORDNET lexicographer files, to previously unseen common nouns. My results significantly outperform existing approaches.

Chapter 1 introduces semantic similarity, describing the theoretical and practical problems of defining synonymy and other lexical-semantic relations. It discussed Roget's thesaurus, WORDNET, and other manually constructed resources and their ongoing contribution to NLP research and applications. However, the cost and complexity of manual development, and the problems with manually developed resources, are presented motivating computational approaches to semantic similarity. The chapter concludes with an overview of the context-space model of semantic similarity which forms the basis of this work. It also describes the partitioning of my experiments into context extraction, similarity measures and algorithmic methods.

Chapter 2 surveys existing evaluation techniques for semantic similarity, finding them unsuitable for a systematic comparison of similarity systems. This motivates the experimental methodology which compares ranked lists of extracted synonyms with a gold-standard created by unifying several thesauri. This ensures semantic similarity is evaluated rather than other properties such as distributional similarity or syntactic substitutability. This chapter also introduces the detailed error analysis which is applied to the large-scale results in Chapter 6.

Chapter 3 argues that shallow methods are better suited to extracting contextual information for semantic similarity because they can process larger volumes of text than is feasible with complex methods. The chapter begins by evaluating several context extractors ranging from simple window-based methods through to wide-coverage parsers. The results demonstrate that shallow methods can perform almost as well as more linguistically sophisticated approaches. However, shallow methods are often several orders of magnitude faster. The results also show that similarity systems improve with increasing corpus size leading me to advocate shallow methods for semantic similarity. Other results demonstrated that smoothing and filtering the context relations can also improve performance whilst reducing the size of the representation.

Chapter 4 hypothesises that the most informative context relations for measuring similarity are strong collocations, proposing new measures based on statistical collocation extraction techniques. The chapter begins by factoring similarity measures into two components: functions for weighting context relations (*weights*) and functions for comparing weighted vectors (*measures*). The DICE† measure and my new TTEST weight, based on the t-test for collocation extraction, significantly outperform the state-of-the-art techniques. The combination of shallow context extraction with the DICE† and TTEST similarity measure forms my similarity system which is used for the remaining results.

Chapter 5 describes an ensemble of similarity systems and proposes two algorithms for practi-

cal synonym extraction. The ensemble combines several context extractors from Chapter 3 using three proposed voting techniques. The results obtained contradict those of Banko and Brill (2001) which led to further analysis. Chapter 5 then discusses the inefficiency of the nearest-neighbour algorithm for semantic similarity which makes extracting synonyms for even moderately large vocabularies infeasible. It presents a novel approximation algorithm which constrains the asymptotic time complexity, significantly reducing the running time of the system, with only a minor performance penalty. Finally, the chapter describes a parallelized version of the similarity system which, running on a Beowulf cluster, allows similarity measurements using contexts extracted from a 2 billion word corpus of shallow-parsed text, the largest such corpus known at this time.

Chapter 6 presents a detailed analysis of the large-scale similarity results from Chapter 5 and describes the application of these similarity results to supersense tagging. The detailed analysis of the large-scale results show that the quality is acceptable for many NLP applications even for quite rare words. The analysis suggests a the minimum number of contexts required for reasonable performance. Finally, the error analysis breaks down the results into various lexical-semantic relations using WORDNET. These results show that about 40% of the top 10 synonyms are closely related either by synonymy or another lexical relation. Chapter 6 also applies the large-scale results to the task of supersense tagging previously unseen common nouns. My results on this task significantly outperform the approach of Ciaramita and Johnson (2003) who have introduced this task.

Through the detailed and systematic analysis of existing approaches to semantic similarity, this thesis has proposed and evaluated a novel approach that significantly outperforms the current state-of-the-art. It presents algorithms that make this approach feasible on unprecedented quantities of text and demonstrates that these results can contribute to advancing wider NLP applications.

Appendix A

Words

TERM	RANK PTB	FREQUENCY PTB	BNC	RCV1	SENSES MQ	OX	WN	DEPTH MIN/MAX	WORDNET ROOT NODES
percent	18	6 131	9 509	836 640	–	–	1	8/8	ABS
company	38	4 098	57 723	459 927	8	5	9	5/8	ENT, GRP, STT
year	42	4 179	163 807	614 675	2	1	4	5/6	ABS, GRP
market	45	3 232	33 563	537 763	4	3	4	4/10	ACT, ENT, GRP
share	65	3 204	16 815	450 387	4	1	5	4/12	ABS, ACT, ENT, POS
stock	69	2 786	9 544	248 868	15	11	17	5/11	ABS, ENT, GRP, POS, STT
sale	88	1 801	19 123	197 873	2	3	5	3/9	ABS, ACT, STT
trading	91	1 269	1 266	75 310	1	–	1	6/6	ACT
president	96	1 453	11 019	172 713	2	3	6	7/10	ACT, ENT
business	102	1 438	39 154	143 163	10	4	9	4/8	ACT, GRP, PSY
price	106	1 935	27 737	335 369	2	3	7	6/10	ABS, ENT, POS
government	110	1 051	66 892	333 080	3	2	4	5/9	ACT, GRP, PSY
cent	111	996	403	131 378	1	–	2	9/14	ABS
quarter	113	1 012	9 225	125 770	11	4	13	5/14	ABS, ENT
time	116	1 318	180 053	173 378	14	8	10	3/8	ABS, EVT, PSY
people	118	907	123 644	147 061	4	5	4	3/8	GRP
investor	119	1 193	3 486	107 147	1	–	1	6/6	ENT
yesterday	124	1 708	96	36 182	1	–	2	5/6	ABS
month	125	1 468	39 779	234 134	–	–	2	5/6	ABS
week	127	1 131	47 367	271 427	–	–	3	6/7	ABS
bond	132	1 219	3 933	160 035	13	3	10	5/11	ABS, ENT, PHE, POS, PSY
group	134	1 270	60 653	221 114	8	5	3	2/4	ABS, ENT, GRP
interest	138	925	38 007	147 376	12	8	7	4/10	ABS, ACT, GRP, POS, STT
earnings	139	759	3 169	85 426	2	1	2	8/10	POS
industry	151	927	24 140	121 348	5	3	3	7/7	ABS, ACT, GRP
money	154	693	37 287	73 921	3	3	3	5/12	ABS, POS
official	155	854	7 875	177 957	3	1	2	6/7	ENT
program	156	905	5 733	19 431	7	–	8	6/11	ABS, ACT, EVT, PSY

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
analyst	157	1 000	1 773	169 046	4	–	3	6/12	ENT
rate	163	1 237	30 335	270 409	3	3	3	5/9	ABS, POS
investment	165	887	12 119	114 115	2	4	3	4/9	ACT, ENT, POS
unit	166	836	17 950	85 910	7	3	7	3/8	ABS, ENT, GRP, PSY
day	170	1 075	93 261	195 186	5	4	10	5/7	ABS, ENT, STT
profit	177	756	11 427	116 465	2	2	2	7/8	ABS, POS
state	182	887	44 903	269 878	8	9	8	2/12	ENT, GRP, PHE, PSY, STT
chairman	184	744	10 414	65 285	1	1	1	7/7	ENT
fund	189	926	12 587	113 668	3	3	3	8/12	ABS, GRP, POS
security	191	1 031	15 737	140 834	6	5	9	4/10	ABS, ACT, ENT, GRP, POS, PSY, STT
bank	198	1 414	20 959	431 169	11	5	10	5/11	ACT, ENT, GRP, POS
firm	198	871	13 879	114 245	3	1	1	7/7	GRP
part	202	659	61 852	89 055	6	8	12	3/9	ABS, ACT, ENT, POS, PSY, STT
product	203	813	21 704	101 787	2	2	6	4/8	ENT, GRP, PHE, PSY
plan	207	800	23 710	100 372	6	3	3	6/9	ENT, PSY
issue	218	948	27 437	124 031	12	6	11	3/12	ACT, ENT, EVT, PHE, POS, PSY
trader	223	583	1 666	182 529	3	1	1	8/8	ENT
loss	229	789	15 430	91 084	3	5	8	4/8	ABS, ACT, EVT, PHE, POS
house	230	687	49 954	69 124	10	7	12	5/8	ACT, ENT, GRP
way	231	590	109 981	65 929	10	11	12	4/8	ABS, ACT, ENT, POS, PSY, STT
tax	233	610	18 694	109 496	5	2	1	6/6	POS
growth	238	460	13 043	114 105	5	5	7	4/7	ENT, EVT, GRP, PHE, STT
index	244	545	4 587	123 960	5	3	5	9/11	ABS, ENT
executive	252	722	7 921	38 103	2	2	3	7/8	ENT, GRP
concern	268	550	12 385	39 354	7	6	5	5/7	GRP, PSY, STT
computer	278	705	17 300	30 538	2	1	2	6/8	ENT
case	287	562	61 488	52 481	11	12	18	4/10	ABS, ACT, ENT, EVT, GRP, PSY, STT
today	292	400	528	57 237	1	2	2	5/6	ABS
number	295	465	60 584	91 595	8	5	11	5/10	ABS, ENT, EVT, GRP
trade	307	556	20 394	168 530	5	3	7	5/9	ACT, GRP, PHE
oil	310	436	11 040	138 946	4	3	2	8/9	ENT
law	311	470	31 004	61 579	8	7	7	4/10	ABS, ACT, GRP, PSY
end	313	390	46 458	98 507	10	6	14	3/9	ABS, ACT, ENT, EVT, PSY, STT
value	315	440	25 308	56 954	12	3	6	4/9	ABS, PSY
dollar	321	581	3 700	153 394	2	–	4	7/14	ABS
system	324	716	61 885	95 161	9	4	9	3/8	ABS, ENT, GRP, PSY
street	326	431	14 777	47 275	2	1	5	5/8	ENT, GRP, STT
result	331	581	33 834	114 062	2	4	3	3/8	ABS, EVT, PHE
point	341	665	50 858	188 234	28	11	24	3/11	ABS, ENT, PSY, STT
problem	344	623	56 361	63 344	4	3	3	5/9	ABS, PSY, STT
world	344	502	59 062	122 434	6	6	8	3/8	ENT, GRP, PSY, STT
country	374	502	48 146	172 593	5	5	5	4/7	ENT, GRP
work	382	354	75 277	36 454	9	10	7	4/8	ACT, ENT, PHE, PSY
report	411	490	34 119	120 036	10	8	7	5/9	ABS, ACT, EVT, PSY
power	414	367	38 447	86 578	16	9	9	3/10	ABS, ENT, GRP, PHE, PSY, STT
service	419	776	54 938	120 161	7	8	15	4/9	ABS, ACT, ENT, GRP, PHE

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
home	425	435	39 798	56 565	6	6	9	4/10	ENT, GRP, STT
order	430	503	25 582	58 825	14	13	12	3/11	ABS, ACT, GRP, STT
member	444	384	47 003	86 821	4	3	5	5/7	ABS, ENT, GRP
level	464	422	32 505	118 852	3	4	7	3/10	ABS, ENT, STT
line	464	432	32 766	59 035	27	20	29	4/12	ABS, ACT, ENT, GRP, PHE, POS, PSY
drop	492	256	3 502	19 362	16	7	8	6/9	ABS, ACT, ENT, EVT
life	508	333	64 797	36 123	9	10	13	4/11	ABS, ENT, PHE, PSY, STT
area	519	395	58 417	74 497	5	5	6	5/7	ABS, ENT, PSY, STT
change	536	407	40 065	55 487	9	3	10	4/14	ABS, ACT, ENT, EVT, PHE
information	566	238	38 630	41 975	3	1	5	4/15	ABS, GRP, PSY
thing	566	373	77 246	27 601	7	16	12	3/8	ABS, ACT, ENT, EVT, PSY, STT
car	595	390	35 184	45 867	4	2	5	9/10	ENT
gas	623	242	8 176	64 562	10	1	6	5/10	ENT, PHE, STT
fact	631	217	42 199	19 100	3	3	4	5/8	ABS, PSY, STT
family	653	257	42 486	27 718	8	5	7	4/7	ENT, GRP
statement	666	226	13 988	126 527	7	1	7	4/10	ABS, ACT, ENT
talk	666	238	11 266	89 743	4	6	5	5/9	ABS, ACT
place	685	207	53 534	43 489	9	8	16	4/10	ABS, ACT, ENT, PSY, STT
dealer	710	227	3 473	126 775	5	2	5	7/9	ENT, GRP
parent	735	234	20 046	12 525	4	2	1	9/9	ENT
magazine	742	260	6 008	8 417	5	1	6	7/10	ENT, GRP
head	753	183	38 526	48 014	39	7	32	4/11	ABS, ENT, EVT, GRP, PHE, PSY
something	778	161	–	11 645	2	–	1	4/4	ENT
institution	826	206	11 389	20 980	8	5	5	5/9	ACT, ENT, GRP, PSY
course	836	162	26 849	16 095	18	9	8	4/7	ACT, ENT, GRP
team	843	183	22 794	41 252	6	2	2	5/6	GRP
trust	898	260	8 963	22 367	6	3	6	4/7	ABS, GRP, POS, PSY, STT
man	929	269	98 731	43 989	9	6	11	3/11	ENT, GRP
question	936	245	39 108	26 486	3	3	6	6/9	ABS, ACT
floor	1 008	138	12 690	12 056	6	4	9	5/12	ENT, GRP, PSY
night	1 022	147	39 188	23 955	4	1	8	5/10	ABS, PSY, STT
announcement	1 041	135	2 758	24 180	2	2	2	8/9	ABS
school	1 052	239	52 132	28 876	5	4	7	5/8	ABS, ENT, GRP, PSY
side	1 052	184	39 608	42 559	9	8	12	5/10	ABS, ENT, EVT, GRP, PSY
software	1 058	125	9 347	17 240	–	–	1	10/10	ABS
woman	1 058	224	63 042	33 566	5	3	4	5/10	ENT, GRP
party	1 078	258	52 944	130 214	8	5	5	5/6	ENT, EVT, GRP
child	1 086	165	70 868	28 602	4	1	4	5/7	ENT
game	1 086	178	21 174	38 069	6	6	8	4/8	ABS, ACT, ENT, EVT
hand	1 086	206	53 432	25 307	13	7	14	4/11	ABS, ACT, ENT, GRP, PSY
size	1 102	116	14 422	14 290	6	1	5	4/8	ABS, ENT, STT
space	1 102	133	14 108	12 231	8	6	10	3/10	ABS, ENT
energy	1 142	178	13 078	41 270	3	2	6	5/12	ABS, GRP, PHE, STT
letter	1 142	158	21 471	19 178	2	3	5	9/11	ABS, ACT, ENT
study	1 142	167	33 083	18 947	9	4	10	6/9	ABS, ACT, ENT, PSY
justice	1 152	141	5 790	15 752	3	4	4	7/12	ABS, ACT, ENT, GRP

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
chain	1 184	155	4 968	10 863	8	4	10	4/9	ABS, ENT, GRP
total	1 184	116	4 306	28 254	4	1	2	4/8	ENT, PSY
age	1 208	117	25 419	9 774	3	4	5	5/7	ABS
idea	1 220	134	32 754	13 535	10	6	5	5/9	ENT, PSY
newspaper	1 220	164	8 539	58 723	1	1	4	7/10	ENT, GRP
form	1 269	131	37 651	14 684	18	12	16	4/10	ABS, ENT, GRP, PHE, PSY
water	1 286	111	33 873	23 515	4	2	5	3/8	ENT
date	1 299	102	17 324	32 483	8	4	8	5/10	ABS, ENT, GRP
room	1 414	131	36 352	12 451	3	3	4	5/7	ABS, ENT, GRP, STT
image	1 466	97	11 026	6 697	10	8	7	5/9	ABS, ENT, PSY
book	1 487	151	37 661	16 270	7	3	8	7/9	ABS, ENT
land	1 547	100	20 922	22 477	5	5	11	4/9	ACT, ENT, GRP, POS, STT
aircraft	1 586	94	6 200	17 165	1	1	1	9/9	ENT
hurricane	1 646	101	695	3 655	3	1	1	9/9	PHE
limit	1 661	116	6 741	14 530	2	4	6	5/8	ABS, ENT
improvement	1 683	107	6 610	19 417	2	1	3	4/6	ACT, EVT, STT
scientist	1 705	88	5 522	4 897	2	1	1	6/6	ENT
word	1 766	124	43 744	8 839	8	11	10	6/10	ABS, ACT, PSY
sport	1 791	115	8 703	11 554	9	2	5	4/8	ABS, ACT, ENT
fraud	1 805	71	1 751	7 334	7	4	3	7/9	ACT, ENT
opinion	1 935	80	9 295	16 378	4	1	6	6/10	ABS, ACT, PSY
truck	1 962	126	1 863	14 004	7	2	2	10/10	ENT
apple	2 000	100	3 237	5 927	4	–	2	10/11	ENT
sun	2 037	73	10 925	8 392	4	2	5	7/9	ABS, ENT, PHE
contrast	2 062	59	7 000	4 290	3	2	4	5/8	ABS, ACT, PSY
bit	2 127	59	14 842	17 888	12	3	10	5/9	ABS, ENT, EVT
fear	2 187	109	9 936	19 814	2	4	2	5/6	PSY
professor	2 267	65	2 274	4 038	2	1	1	9/9	ENT
radio	2 267	98	9 072	26 060	2	–	3	8/10	ENT
eye	2 344	82	39 153	8 131	7	7	5	6/8	ENT, PSY
patient	2 432	63	21 653	8 048	1	1	1	7/7	ENT
crop	2 467	65	3 011	32 327	9	4	3	7/10	ACT, ENT
picture	2 467	101	15 986	10 237	7	6	9	4/9	ABS, ENT, EVT, PSY, STT
sea	2 608	62	11 556	19 226	4	3	3	4/7	ABS, ENT, PHE
cause	2 818	61	10 696	7 426	2	4	5	3/10	ABS, ACT, ENT, EVT
stage	2 874	59	20 630	19 662	10	5	8	3/11	ABS, ACT, ENT, STT
challenge	2 944	61	6 438	11 394	3	3	5	4/9	ABS, ACT, STT
concept	2 944	41	9 071	3 278	2	1	1	6/6	PSY
purpose	3 006	74	15 180	9 031	3	6	3	6/6	ABS, PSY
arrangement	3 071	58	9 051	7 349	6	4	8	3/8	ABS, ACT, ENT, GRP, PSY
promotion	3 071	61	3 696	4 258	5	3	4	5/9	ABS, ACT
star	3 199	65	8 563	11 538	11	4	7	6/10	ABS, ENT
analysis	3 265	40	14 229	5 675	8	2	6	6/11	ABS, ACT, PSY
location	3 265	57	5 499	5 470	7	1	3	3/8	ACT, ENT
remark	3 265	39	3 325	9 408	1	2	2	8/8	ABS, PSY
experiment	3 514	55	5 728	1 505	3	2	3	7/9	ACT, PSY

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
wine	3 603	49	7 349	3 559	1	1	2	9/10	ABS, ENT
apparel	3 788	29	67	1 978	2	1	1	7/7	ENT
novel	3 788	37	3 401	1 129	3	1	2	8/10	ABS, ENT
tool	3 788	52	5 382	4 849	5	2	4	6/8	ACT, ENT
indictment	3 900	32	401	1 842	1	1	2	10/15	ABS
crowd	4 030	35	5 650	7 486	6	4	2	5/5	GRP
consequence	4 262	32	7 719	3 909	5	2	3	3/9	ABS, EVT, PHE
skill	4 262	29	12 276	2 544	3	2	2	5/5	PSY
baby	4 577	31	11 496	3 747	5	3	6	5/8	ACT, ENT
bureaucracy	4 577	25	1 529	1 038	2	2	1	8/8	GRP
explanation	4 577	25	6 404	2 175	4	2	2	7/8	ABS, PSY
conflict	4 726	35	7 097	11 077	3	3	7	3/7	ABS, ACT, PSY, STT
missile	4 726	37	1 902	9 878	1	1	2	9/9	ENT
component	5 099	37	5 019	9 778	1	1	3	4/8	ABS, ENT, PSY
furniture	5 099	21	3 545	2 269	2	1	1	7/7	ENT
human	5 099	37	2 593	7 138	1	1	2	4/11	ENT
knowledge	5 099	19	14 580	2 836	3	5	1	3/3	PSY
tradition	5 099	23	6 740	2 242	2	3	2	5/5	PSY
box	5 314	38	11 478	5 169	16	3	10	6/11	ABS, ACT, ENT, STT
song	5 314	28	6 832	2 110	3	2	6	5/9	ABS, ACT, ENT, EVT, GRP, POS
dream	5 537	23	6 416	2 223	8	4	6	4/8	PSY, STT
entity	6 077	28	1 818	4 352	2	3	1	2/2	ENT
enthusiasm	6 401	14	2 949	2 005	4	2	3	4/8	ABS, PSY, STT
taste	6 401	18	4 413	1 173	6	7	7	5/9	ABS, ACT, EVT, PSY
laboratory	6 760	50	3 748	3 285	2	1	1	7/7	ENT
anger	7 130	12	3 691	2 281	3	1	3	5/8	ACT, PSY, STT
ball	7 130	29	8 750	7 730	9	3	10	5/11	ABS, ACT, ENT, GRP
chaos	7 130	13	1 633	2 445	2	1	3	5/9	PHE, PSY, STT
limitation	7 130	14	2 734	1 106	3	2	5	5/8	ABS, ACT, PSY
boat	8 136	17	7 345	6 128	2	1	2	10/10	ENT
carpet	8 136	16	3 284	882	4	2	1	7/7	ENT
disadvantage	8 136	11	1 966	874	3	2	1	7/7	ABS
suburb	8 136	12	1 097	2 991	1	1	1	7/7	ENT
artery	8 721	16	575	865	1	2	2	8/9	ENT
fish	8 721	9	9 711	3 042	3	1	2	7/8	ENT
catholic	9 429	12	1 246	4 722	–	–	1	7/7	ENT
nervousness	9 429	8	326	1 589	2	1	3	5/7	ABS, PSY, STT
reinforcement	9 429	11	608	861	4	3	5	6/9	ABS, ACT, ENT, PSY
garbage	10 247	14	262	666	6	2	1	7/7	ENT
ring	10 247	10	5 759	2 582	11	7	8	5/8	ABS, ENT, EVT, GRP
viewpoint	10 247	9	1 139	338	3	1	2	7/7	ENT, PSY
village	10 247	13	13 359	9 949	2	–	3	6/7	ENT, GRP
bat	11 236	9	1 360	976	11	1	5	5/9	ACT, ENT
bomb	11 236	10	4 070	13 236	9	3	3	6/10	ENT, EVT
fence	11 236	7	2 244	1 055	7	2	2	8/9	ENT
imagination	11 236	6	2 652	405	2	3	3	6/7	PSY

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
nonsense	11 236	6	1 632	551	4	3	2	6/7	ABS, ENT
probability	11 236	7	1 946	1 253	1	2	2	4/5	ABS
resentment	11 236	6	1 005	371	2	1	1	7/7	PSY
resilience	11 236	6	227	430	3	2	2	6/7	ABS, EVT
t-shirt	11 236	7	762	469	–	–	–	–/–	
championship	12 636	8	4 667	11 711	4	–	2	5/7	ACT, STT
jolt	12 636	10	192	225	3	4	2	6/7	ACT, EVT
pants	12 636	5	547	282	3	2	2	9/11	ENT
religion	12 636	7	5 127	1 596	2	1	2	6/10	ABS, GRP, PSY
sweater	12 636	7	773	129	2	–	2	5/9	ENT
forum	14 425	9	1 823	6 067	4	3	3	6/10	ENT, GRP
object	14 425	6	10 087	1 332	5	3	4	3/10	ABS, ENT, PSY
slogan	14 425	7	815	1 519	2	1	1	9/9	ABS
glitch	16 942	5	45	409	1	–	1	5/5	STT
graph	16 942	4	1 417	89	2	1	1	9/9	ABS
pistol	16 942	5	892	805	1	1	1	11/11	ENT
powder	16 942	5	1 413	898	2	1	3	6/8	ENT
routine	16 942	4	2 493	442	7	2	3	5/11	ABS, ACT, EVT
silence	16 942	5	5 803	1 398	4	3	4	4/6	ABS, STT
aroma	20 917	3	348	123	2	1	2	5/9	ABS, PSY
cab	20 917	4	1 570	438	1	2	3	9/11	ENT
disgust	20 917	2	620	196	1	2	1	5/5	PSY
felon	20 917	4	84	89	2	–	2	7/7	ENT, STT
instructor	20 917	3	868	387	1	1	1	8/8	ENT
moisture	20 917	2	699	1 806	2	1	1	5/5	STT
organisation	20 917	2	8 324	20 356	3	4	7	4/7	ABS, ACT, GRP, PSY
revenge	20 917	2	1 037	1 040	1	2	1	5/5	ACT
solicitor	20 917	4	5 582	350	1	1	2	7/8	ENT
spectacle	20 917	2	637	239	3	3	3	6/9	ABS, ACT, ENT
walk	20 917	3	5 554	1 217	7	4	7	5/8	ABS, ACT, ENT
warrior	20 917	4	1 069	670	1	1	1	5/5	ENT
alligator	28 728	2	109	107	1	–	2	10/10	ENT
bedding	28 728	1	416	91	6	1	2	5/7	ENT
connotation	28 728	2	373	49	3	1	2	7/8	ABS, PSY
grief	28 728	1	1 409	366	2	2	2	6/7	PSY
happiness	28 728	1	1 695	198	5	1	2	4/6	PSY
influenza	28 728	1	139	335	3	–	1	10/10	STT
psyche	28 728	1	242	71	4	1	3	4/9	ENT, PSY
tightness	28 728	1	122	2 025	5	–	3	6/7	ABS, STT
trousers	28 728	1	2 257	309	1	1	1	9/9	ENT
additive	–	2	232	565	1	1	1	7/7	ENT
aristocrat	–	–	329	240	3	1	1	6/6	ENT
automatic	–	4	33	85	3	–	2	13/13	ENT
beam	–	7	1 698	317	10	3	6	7/9	ABS, ENT, PHE
bloke	–	–	1 616	26	1	1	1	7/7	ENT
cafeteria	–	8	142	107	1	1	1	8/8	ENT

300 headword evaluation set

	RANK	FREQUENCY			SENSES			DEPTH	WORDNET
TERM	PTB	PTB	BNC	RCV1	MQ	OX	WN	MIN/MAX	ROOT NODES
capacity	–	118	6 244	29 087	9	3	8	5/9	ABS, ACT, PHE, PSY, STT
celebrant	–	–	46	8	4	–	2	5/9	ENT
cipher	–	4	88	2	10	4	5	5/10	ABS, ENT
coincidence	–	6	982	311	4	3	3	5/9	ABS, EVT
colliery	–	–	533	255	1	–	1	7/7	ENT
cunning	–	–	114	16	4	1	3	7/10	ABS, PSY
diagnosis	–	5	1 864	403	3	2	1	9/9	ACT
estuary	–	–	650	90	2	1	1	4/4	ENT
fortress	–	–	589	216	1	1	1	7/7	ENT
grin	–	1	1 253	118	2	1	1	9/9	ABS
hair	–	16	14 999	1 388	3	3	6	6/7	ABS, ENT
handful	–	36	1 489	2 574	2	2	2	6/6	ABS
intuition	–	3	572	40	2	2	2	6/7	PSY
knob	–	–	444	26	4	4	4	6/7	ABS, ENT
leadership	–	61	4 870	9 747	2	2	4	4/6	ACT, GRP, PSY, STT
luggage	–	–	654	434	1	1	1	8/8	ENT
manslaughter	–	–	504	355	1	1	1	8/8	ACT
mix	–	26	1 908	2 933	3	1	3	6/8	ACT, ENT, EVT
monarch	–	1	961	796	1	1	2	9/11	ENT
monument	–	–	1 298	830	2	3	3	6/8	ENT
morale	–	7	957	1 085	1	1	2	5/7	ABS, STT
mug	–	–	1 033	46	6	4	4	6/9	ABS, ENT
nothing	–	133	29	14 227	3	4	1	5/5	ABS
novelist	–	7	853	421	1	1	1	7/7	ENT
paradigm	–	–	677	107	1	1	4	5/7	ABS, GRP, PSY
pastry	–	4	580	114	–	2	2	8/9	ENT
pint	–	1	1 743	171	1	–	3	9/9	ABS
recess	–	3	363	926	5	3	5	4/7	ABS, ACT, ENT, STT
sausage	–	–	985	224	2	1	2	8/12	ENT
sceptic	–	–	288	252	2	2	1	6/6	ENT
scream	–	2	980	191	2	3	3	6/9	ABS, EVT
scuffle	–	1	145	382	2	1	3	6/9	ACT, ENT
sermon	–	–	662	97	6	2	2	5/11	ABS, ACT
spanner	–	–	222	58	–	–	1	9/9	ENT
standpoint	–	7	339	540	1	1	1	7/7	PSY
terrier	–	–	300	45	2	–	1	13/13	ENT
thesis	–	1	1 812	138	4	2	2	9/11	ABS
throw	–	–	677	527	6	2	6	5/8	ABS, ACT, ENT, EVT, STT
tonne	–	–	2 182	149 620	1	–	1	8/8	ABS
virus	–	18	1 980	2 582	3	1	1	5/5	ENT
vocation	–	–	301	87	3	1	2	5/6	ACT, GRP
whisky	–	–	1 935	277	1	1	1	10/10	ENT

300 headword evaluation set

Appendix B

Roget's Thesaurus

company
assembly 74 n.
band 74 n.
accompaniment 89 n.
actor 594 n.
personnel 686 n.
workshop 687 n.
association 706 n.
corporation 708 n.
party 708 n.
formation 722 n.

74 Assemblage

N. *assembly*, mutual attraction, 291 *attraction*; getting together, ganging up; forgathering, congregation, concourse, conflux, concurrence, 293 *convergence*; gathering, meeting, mass meeting, protest meeting, sit-in, meet; coven; conventicle; business meeting, board m.; convention, convocation, 985 *synod*; gemot, shire moot, legislature, conclave, 692 *council*; eisteddfod, mod, festival, 876 *celebration*; reunion, get-together, gathering of the clans, ceilidh, 882 *social gathering*; company, at home, party, 882 *sociality*; circle, sewing bee, knit-in; encounter group, 658 *therapy*; discussion group, focus g., quality circle, symposium, 584 *conference*.

band, company, troupe; cast, 594 *actor*; brass band, dance b., pop group, rock g., boy-band tribute b., 413 *orchestra*; team, string, fifteen, eleven, eight; knot, bunch; set, coteri, dream team; clique, ring; gang, squad, party, work p., fatigue p.; ship's company, crew, complement, manpower, workforce, staff, 686 *personnel*; following, 67 *retinue*; squadron, troop, platoon, unit, regiment, corps, 722 *formation*; squad, posse; force, body, host, 722 *armed force*; 104 *multitude*; (Boy) Scouts, Girl Guides, 708 *society*; band of brothers, sisters, merry men, 880 *friendship*; committee, commission, 754 *consignee*; panel, 87 *list*; establishment, cadre, 331 *structure*.

89 Accompaniment

N. *accompaniment*, concomitance, 71 *continuity*, 45 *union*, 5 *intrinsicality*; inseparability, permanent attribute; society, 882 *sociability*; companionship, togetherness, 880 *friendship*; partnership, marriage, 706 *association*; coexistance, coagency, 181 *concurrence*; coincidence, contemporaneity, simultaneity, 123 *synchronism*; attendance, company; parallel course, 219 *parallelism*.

Figure B.1: Roget's Thesaurus of English words and phrases Davidson (2002) entry for **company**

company
assembly 74 n.
band 74 n.
accompaniment 89 n.
actor 594 n.
personnel 686 n.
workshop 687 n.
association 706 n.
corporation 708 n.
party 708 n.
formation 722 n.

594 Drama, Ballet

N. *actor*, actress, Thespian, Roscius, luvvy (inf); mimic, mime, pantomimist, 20 *imitator*; mummer, masker, guisard; play-actor, player, strolling p., trouper, cabotin(e); barn-stormer, ham; rep player, character actor; actor-manager, star, star actor *or* actress, star of stage and screen, film star, starlet, matineé idol, 890 *favourite*; tragedian, tragedienne; comedian, comedienne, comedy actor *or* actress; opera singer, prima donna, diva; ballet dancer, ballerina, prima b., coryphée; danseur, danseuse, figurant(e); protagonist, lead, second l., leading man, leading lady, juvenile lead, jeune premier; understudy, stand-in, body double, stunt man *or* woman, 150 *substitute*; lookalike, 18 *analogue*; supernumerary, super, extra, bit player; chorus, gentlemen *or* ladies of the chorus, corps de ballet, troupe, company, repertory c., stock c.; dramatis personae, characters, cast; presenter, narrator; prologue, 579 *speaker*.

686 Agent

N. *personnel*, staff, force, company, team, gang, squad, crew, complement, cadre, 74 *band*; dramatis personae, 594 *actor*; co-worker, fellow w., mate, colleague, associate, partner, 707 *colleague*; workpeople, hands, men, payroll; labour, casual l.; workforce, labour pool, labour force, human resources, liveware, peopleware, manpower; working classes, proletariat; personnel management, human resource management; staff turnover, churn, churn rate.

687 Workshop

N. *workshop*, studio, atelier; workroom, study, den, library; laboratory, research l.; plant, installation; business park, industrial estate, science park, technopole; works, factory, manufactory; workshop, yard; sweatshop; mill, cotton m., loom; sawmill, paper mill; foundry, metalworks; steelyard, steelworks, smelter; blast furnace, forge, smithy, stithy, 383 *furnace*; power hourse, power station, gasworks, 160 *energy*; quarry, mine, 632 *store*; colliery, coal-mine, pit, coalface; tin mine, stannary; mint; arsenal, armoury; dockyard, shipyard, slips; wharf, dock, 192 *shed*; construction site, building s.; refinery, distillery, brewery, maltings; shop, shopfloor, bench, production line; nursery, 370 *farm*; dairy, creamery, 369 *stock farm*; kitchen, laundry; office, bureau, call centre; business house, firm, company; offices, secretariat, Whitehall; manufacturing town, hive of industry, 678 *activity*.

Figure B.1: Roget's Thesaurus of English words and phrases Davidson (2002)
entry for **company** (continued)

company
assembly 74 n.
band 74 n.
accompaniment 89 n.
actor 594 n.
personnel 686 n.
workshop 687 n.
association 706 n.
corporation 708 n.
party 708 n.
formation 722 n.

706 Cooperation

N. *association*, coming together; colleagueship, co-ownership, copartnership, partnership, 775 *participation*; nationalization, internationalization, 775 *joint possession*; pooling, pool, kitty; membership, affiliation, 78 *inclusion*; connection, hook-up, tie-up, 9 *relation*; consociation, ecosystem; combination, consolidation, centralization, 45 *union*; integration, solidarity, 52 *whole*; unification, 88 *unity*; amalgamation, fusion, merger; voluntary association, coalition, cohabitation, alliance, league, federation, confederation, confederacy, umbrella organisation; axis, united front, common f., people's f., popular f., 708 *political party*; association, fellowship, college, club, sodality, fraternity, sorority, 708 *community*; set, clique, coterie, cell, 708 *party*; workers' association, trade union, chapel; business association, company, joint-stock c., limited liability c., private c., public c., public limited c., PLC, Ltd, syndicate, combine, consortium, trust, cartel, ring, 708 *corporation*; housing association, economic community, cooperative, workers' c., commune, 708 *community*.

708 Party

N. *party*, movement; group, class, 77 *classification*; subsect, confession, communion, denomination, church, 978 *sect*; faction, groupuscule, cabal, cave, splinter group, breakaway, movement, 489 *dissentient*; circle, inner c., charmed c., kitchen cabinet; set, clieque, incrowd, coterie, galère; caucus, junta, camarilla, politburo, committee, quango, club, cell, cadre; ring, closed shop; team, eight, eleven, fifteen; crew, team, complement, 686 *personnel*; troupe, company, 594 *actor*; gang, knot, bunch, outfit, 74 *band*; horde, 74 *crowd*; side, camp.

corporation, body; incorporated society, body corporate, mayor and corporation, 692 *council*; company, livery c., joint-stock c., limited liability c., public limited c., holding c.; private c.; multinational c., transnational corporation, dotcom, 706 *association*; firm, concern, joint c., partnership; house, business h.; establistment, organization, institute; trust, combine, monopoly, cartel, syndicate, conglomerate, 706 *assocation*; trade association, chamber of commerce, guild, cooperative society; consumers' association.

722 Combatant. Army. Navy. Air Force

N. *formation*, array, line; square, phalanx; legion, cohort, century, decury, maniple; column, file, rank; unit, group, detachment, corps, army c., division, armoured d., panzer d.; brigade, rifle b., light b., heavy b.; artillery brigade, battery; regiment, cavalry r., squadron, troop; battalion, company, platoon, section, squad, detail, party, 74 *band*.

Figure B.1: Roget's Thesaurus of English words and phrases Davidson (2002) entry for **company** (continued)

Bibliography

Naoki Abe and Hang Li. Learning word association norms using tree cut pair models. In *Proceedings of the 13th International Conference on Machine Learning*, pages 3–11, Bari, Italy, 3–6 July 1996.

Steven Abney. Parsing by chunks. In Robert C. Berwick, Steven P. Abney, and Carol Tenny, editors, *Principle-Based Parsing: Computation and Psycholinguistics*, pages 257–278. Kluwer Academic Publishers, Boston, MA USA, 1991.

Steven Abney. Partial parsing via finite-state cascades. *Journal of Natural Language Engineering*, 2(4):337–344, December 1996.

Steven Abney. The SCOL manual - version 0.1b, 28 April 1997.

Steven Abney and Marc Light. Hiding a semantic hierarchy in a Markov model. In *Proceedings of the Workshop on Unsupervised Learning in Natural Language Processing*, pages 1–8, College Park, MD USA, 21 June 1999.

Eugene Agichtein, Eleazar Eskin, and Luis Gravano. Combining strategies for extracting relations from text collections. Technical Report CUCS-006-00, Department of Computer Science, Columbia University, New York, NY USA, March 2000.

Eugene Agichtein and Luis Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM Conference on Digital Libraries*, pages 85–94, San Antonio, TX USA, 2–7 June 2000.

Jean Aitchison, Alan Gilchrist, and David Bawden. *Thesaurus Construction: a practical manual*. Fitzroy Dearborn Publishers, London, UK, 4th edition, 2002.

Michael R. Anderberg. *Cluster Analysis for Applications*. Probability and Mathematical Statistics. Academic Press, New York, NY USA, 1973.

Jordi Atserias, Salvador Climent, Xavier Farreres, German Rigau, and Horacio Rodríguez. Combining multiple methods for the automatic construction of multilingual WordNets. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pages 143–149, Tzigov Chark, Bulgaria, 11–13 September 1997.

R. Attar and A. S. Fraenkel. Local feedback in full-text etrieval systems. *Journal of the Association for Computing Machinery*, 24(3):397–417, July 1977.

Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, New York, NY USA, 1999.

L. Douglas Baker and Andrew McCallum. Distributional clustering of words for text classification. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 96–103, Melbourne, Australia, 24–28 August 1998.

Timothy Baldwin, Colin Bannard, Takaaki Tanaka, and Dominic Widdows. An empirical model of multiword expression decomposability. In *Proceedings of the Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 89–96, 12 July 2003.

Satanjeev Banerjee and Ted Pederson. An adapted lesk algorithm for word sense disambiguation using WordNet. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics*, pages 136–145, Mexico City, Mexico, 17–23 February 2002.

Satanjeev Banerjee and Ted Pederson. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 805–810, Acapulco, Mexico, 9–15 August 2003.

Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 26–33, Toulouse, France, 9–11 July 2001.

Colin Bannard, Timothy Baldwin, and Alex Lascarides. A statistical approach to the semantics of verb-particles. In *Proceedings of the Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 65–72, Sapporo, Japan, 12 July 2003.

Roberto Basili, Alessandro Cucchiarelli, Carlo Consoli, Maria Teresa Pazienza, and Paola Velardi. Automatic adaptation of WordNet to sublanguages and to computational tasks. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 80–86, Montréal, Québec, Canada, 16 August 1998.

Doug Beeferman. Lexical discovery with an enriched semantic network. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 358–364, Montréal, Québec, Canada, 16 August 1998.

Laura Benítez, Sergi Cervell, Gerard Escudero, Mònica López, German Rigau, and Mariona Taulé. Methods and tools for building the catalan WordNet. In *Proceedings of the ELRA Workshop on Language Resources for European Minority Languages*, Granada, Spain, 27 May 1998.

Matthew Berland and Eugene Charniak. Finding parts in very large corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 57–64, College Park, MD USA, 20–26 June 1999.

John R.L. Bernard, editor. *The Macquarie Encyclopedic Thesaurus*. The Macquarie Library, Sydney, Australia, 1990.

Bran Boguraev and Ted Briscoe, editors. *Computational Lexicography for Natural Language Processing*. Longman Group UK Limited, Harlow, Essex, UK, 1989a.

Bran Boguraev and Ted Briscoe. Introduction. In *Computational Lexicography for Natural Language Processing* Boguraev and Briscoe (1989a), chapter 1, pages 1–40.

Thorsten Brants. TnT - a statistical part-of-speech tagger. In *Proceedings of the 6th Applied Natural Language Processing Conference*, pages 224–231, Seattle, WA USA, 29 April–4 May 2000.

Eric Brill and Jun Wu. Classifier combination for improved lexical disambiguation. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th annual meeting of the Association for Computational Linguistics*, pages 191–195, Montréal, Québec, Canada, 10–14 August 1998.

Ted Briscoe and John Carroll. Robust accurate statistical annotation of general text. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1499–1504, Las Palmas de Gran Canaria, 29–31 May 2002.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai, and Robert L. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4): 467–479, December 1992.

Sabine Buchholz. README for Perl script `chunklink.pl`, 2000. `http://ilk.kub.nl/~sabine/chunklink/README.html`.

Alexander Budanitsky. Lexical semantic relatedness and its application in natural language processing. Technical Report CSRG-390, Computer Systems Research Group, University of Toronto, Toronto, Canada, August 1999.

Alexander Budanitsky and Graeme Hirst. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Proceedings of the Second meeting of the North American Chapter of the Association for Computational Linguistics*, Pittsburgh, PA USA, 2–7 June 2001.

Paul Buitelaar. *CoreLex: Systematic Polysemy and Underspecification*. PhD thesis, Computer Science, Brandeis University, February 1998.

Anita Burgun and Olivier Bodenreider. Comparing terms, concepts and semantic classes in WordNet and the Unified Medical Language System. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 77–82, Pittsburgh, PA USA, 2–7 June 2001.

Lou Burnard, editor. *Users Reference Guide British National Corpus Version 1.0*. Oxford University Computing Services, 1995.

Sharon A. Caraballo. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 120–126, College Park, MD USA, 20–26 June 1999.

Sharon A. Caraballo and Eugene Charniak. Determining the specificity of nouns from text. In *Proceedings of the Joint ACL SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 63–70, College Park, MD USA, 21–22 June 1999.

John Carroll, Ted Briscoe, and Antonio Sanfilippo. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, pages 447–454, Granada, Spain, 28–30 May 1998.

Gabriela Cavaglià. The development of lexical resources for information extraction from text combining WordNet and Dewey decimal classification. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 225–228, Bergen, Norway, 8–12 June 1999.

Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information and lexicography. In *Proceedings of the 27th annual meeting of the Association for Computational Linguistics*, pages 76–82, Vancouver, BC Canada, 1989.

Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information and lexicography. *Computational Linguistics*, 16(1):22–29, March 1990.

Massimiliano Ciaramita. Boosting automatic lexical acquisition with morphological information. In *Proceedings of the Workshop on Unsupervised Lexical Acquisition*, pages 17–25, Philadelphia, PA, USA, 12 July 2002.

Massimiliano Ciaramita, Thomas Hofmann, and Mark Johnson. Hierarchical semantic classification: Word sense disambiguation with world knowledge. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, Acapulco, Mexico, 9–15 August 2003.

Massimiliano Ciaramita and Mark Johnson. Supersense tagging of unknown nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 168–175, Sapporo, Japan, 11–12 July 2003.

Stephen Clark. *Class-Based Statistical Models for Lexical Knowledge Acquisition*. PhD thesis, University of Sussex, 2001.

Stephen Clark and David Weir. A class-based probabilistic approach to structural disambiguation. In *Proceedings of the 18th international conference on Computational Linguistics*, pages 194–200, Saarbrücken, Germany, 31 July–4 August 2000.

Stephen Clark and David Weir. Class-based probability estimation using a semantic hierarchy. *Computational Linguistics*, 28(2):187–206, June 2002.

Michael Collins and James Brooks. Prepositional phrase attachment through a backed-off model. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 27–38, Cambridge, MA USA, 30 June 1995.

Ann Copestake. An approach to building the hierarchical element of a lexical knowledge base

from a machine readable dictionary. In *Proceedings of the First International Workshop on Inheritance in Natural Language Processing*, pages 19–29, Tilburg, The Netherlands, 1990.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley and Sons, Inc., New York, NY USA, 1991.

Koby Crammer and Yoram Singer. Ultraconservative online algorithms for multiclass problems. In *Proceedings of the 14th annual Conference on Computational Learning Theory and 5th European Conference on Computational Learning Theory*, pages 99–115, Amsterdam, The Netherlands, 16–19 July 2001.

Carolyn J. Crouch. Construction of a dynamic thesaurus and its use for associated information retrieval. In *Proceedings of the eleventh international conference on Research and Development in Information Retrieval*, pages 309–320, Grenoble, France, 13–15 June 1988.

D.A. Cruse. *Lexical Semantics*. Cambridge University Press, Cambridge, UK, 1986.

Isabel Cruz, Stefan Decker, Jérôme Euzenat, and Deborah McGuinness, editors. *The Emerging Semantic Web*. IOS Press, 2002.

James R. Curran. Ensemble methods for automatic thesaurus extraction. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 222–229, Philadelphia, PA, USA, 6–7 July 2002.

James R. Curran and Stephen Clark. Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, pages 91–98, Budapest, Hungary, 12–17 April 2003a.

James R. Curran and Stephen Clark. Language independent NER using a maximum entropy tagger. In *Proceedings of the 7th Conference on Natural Language Learning*, pages 164–167, Edmonton, Canada, 31 May–1 June 2003b.

James R. Curran and Marc Moens. Improvements in automatic thesaurus extraction. In *Proceedings of the Workshop on Unsupervised Lexical Acquisition*, pages 59–66, Philadelphia, PA, USA, 12 July 2002a.

James R. Curran and Marc Moens. Scaling context space. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 231–238, Philadelphia, PA, USA, 7–12 July 2002b.

James R. Curran and Miles Osborne. A very very large corpus doesn't always yield reliable estimates. In *Proceedings of the 6th Conference on Natural Language Learning*, pages 126–131, Taipei, Taiwan, 31 August–1 September 2002.

Walter Daelemans. Machine learning approaches. In Hans van Halteren, editor, *Syntactic Wordclass Tagging*, chapter 17. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.

Walter Daelemans, Antal van den Bosch, and Ton Weijters. IGTree: Using trees for compression and classification in lazy learning algorithms. *Artificial Intellligence Review*, 11(1–5): 407–423, February 1997.

Ido Dagan, Lillian Lee, and Fernando Pereira. Similarity-based methods for word-sense disambiguation. In *Proceedings of the 35th annual meeting of the Association for Computational Linguistics and the 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 56–63, Madrid, Spain, 7–11 July 1997.

Ido Dagan, Lillian Lee, and Fernando Pereira. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 34(1–3):43–69, 1999.

Ido Dagan, Shaul Marcus, and Shaul Markovitch. Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st annual meeting of the Association for Computational Linguistics*, pages 164–171, Columbus, Ohio USA, 22–26 June 1993.

Ido Dagan, Shaul Marcus, and Shaul Markovitch. Contextual word similarity and estimation from sparse data. *Computer, Speech and Language*, 9:123–152, 1995.

Ido Dagan, Fernando Pereira, and Lillian Lee. Similarity-based estimation of word cooccurrence probabilities. In *Proceedings of the 32nd annual meeting of the Association for Computational Linguistics*, pages 272–278, Las Cruces, New Mexico, USA, 27–30 June 1994.

George Davidson, editor. *Thesaurus of English words and phrases: 150th Anniversary Edition*. Penguin Books, London, UK, 2002.

Carl G. de Marcken. Parsing the LOB corpus. In *Proceedings of the 28th annual meeting of the Assocation for Computational Linguistics*, pages 243–251, Pittsburgh, PA USA, 6–9 June 1990.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

James E. Deese. On the structure of associative meaning. *Psychological Review*, 69(2):161–175, 1962.

James E. Deese. The associative structure of some common English adjectives. *Journal of Verbal Learning and Verbal Behaviour*, 3(5):347–357, 1964.

Lee R. Dice. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26(3):297–302, 1945.

Thomas G. Dietterich. Ensemble methods in machine learning. In Josef Kittler and Fabio Roli, editors, *Multiple Classifier Systems: First International Workshop*, volume 1857 of *Lecture Notes in Computer Science*, pages 1–15. Springer-Verlag, Heidelberg, Germany, 2000.

Chrysanne DiMarco. The nature of near-synonymic relations. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 691–695, Kyoto, Japan, 5–9 August 1994.

Chrysanne DiMarco, Graeme Hirst, and Manfred Stede. The semantic and stylistic differentiation of synonyms and near-synonyms. In *Building Lexicons for Machine Translation. Papers from the* AAAI *Spring Symposium*, pages 114–121, Stanford, CA USA, 1993.

Philip Edmonds and Graeme Hirst. Near-synonymy and lexical choice. *Computational Linguistics*, 28(2):105–144, June 2002.

Donald L. Emblen. *Peter Mark Roget: The Word and the Man*. Longman Group, London, UK, 1970.

David Evans, Steve K. Henderson, Robert G. Lefferts, and Ira A. Monarch. A summary of the CLARIT project. Technical Report CMU-LCL-91-2, Laboratory for Computational Linguistics, Carnegie-Mellon University, November 1991.

Robert M. Fano. *Transmission of Information: a Statistical Theory of Communications*. MIT Press, Cambridge, MA USA, 2nd print. with corr. edition, 1963.

Xavier Farreres, German Rigau, and Horacio Rodríguez. Using WordNet for building wordnets. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 65–72, Montréal, Québec, Canada, 16 August 1998.

Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA USA, 1998.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, January 2002. available from `http://www.cs.technion.ac.il/~gabr/resources/data/wordsim353/`.

Douglas J. Foskett. Thesaurus. In Karen Spärck Jones and Peter Willett, editors, *Readings in Information Retrieval*, pages 111–134. Morgan Kaufman, 1997.

Henry W. Fowler. *A Dictionary of Modern English Usage*. Oxford University Press, 1926.

Edward A. Fox, J. Terry Nutter, Thomas Ahlswede, Martha Evans, and Judith Markowitz. Building a large thesaurus for information retrieval. In *Proceedings of the Second Conference on Applied Natural Language Processing*, pages 101–108, Austin, TX USA, 9–12 February 1988.

W. Nelson Francis and Henry Kucera. *Frequency analysis of English usage. Lexicon and grammar*. Houghton Mifflin, Boston, MA USA, 1982.

William Gale, Kenneth Ward Church, and David Yarowsky. Work on statistical methods for word sense disambiguation. In *Intelligent Probabilistic Approaches to Natural Language*, number FS-92-04 in Fall Symposium Series, pages 54–60, Stanford University, CA USA, 25–27 March 1992.

Tanja Gaustad. Statistical corpus-based word sense disambiguation: Pseudowords vs. real ambiguous words. In *Companion Volume to the Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 61–66, Toulouse, France, 9–11 July 2001.

Daniel Gildea. Corpus variation and parser performance. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 167–172, Pittsburgh, PA USA, 3–4 June 2001.

Philip B. Gove, editor. *Webster's Seventh New Collegiate Dictionary*. G. & C. Merriam Company, Cambridge, MA USA, 1963.

David Graff. North American News Text Corpus. Technical Report LDC95T21, Linguistic Data Consortium, Philadelphia, PA USA, 1995.

David Graff. The AQUAINT corpus of English news text. Technical Report LDC2002T31, Linguistic Data Consortium, Philadelphia, PA USA, 2002.

David Graff. English Gigaword. Technical Report LDC2003T05, Linguistic Data Consortium, Philadelphia, PA USA, 2003.

David Graff, Roni Rosenfeld, and Doug Paul. CSR-III Text. Technical Report LDC95T6, Linguistic Data Consortium, Philadelphia, PA USA, 1995.

Rebecca Green, Lisa Pearl, Bonnie J. Dorr, and Philip Resnik. Mapping lexical entries in a verbs database to WordNet senses. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 244–251, Toulouse, France, 9–11 July 2001.

Stephen J. Green. Using lexical chains to build hypertext links in newspaper articles. In *Internet-Based Information Systems. Papers from the AAAI Workshop*, pages 56–64, Portland, Oregon USA, 5 August 1996.

Gregory Grefenstette. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Boston, MA USA, 1994.

Gregory Grefenstette and Pasi Tapanainen. What is a word, what is a sentence? Problems of tokenization. In *The 3rd International Conference on Computational Lexicography*, pages 79–87, Budapest, Hungary, 1994.

Ralph Grishman and John Sterling. Generalizing automatically generated selectional patterns. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 742–747, Kyoto, Japan, 5–9 August 1994.

William Gropp, Ewing Lusk, Nathan Doss, and Anthony Skjellum. A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing*, 22(6):789–828, September 1996.

Claire Grover, Colin Matheson, Andrei Mikheev, and Marc Moens. LT TTT - a flexible tokenisation tool. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, pages 1147–1154, Athens, Greece, 31 May–2 June 2000.

Patrick Hanks, editor. *The New Oxford Thesaurus of English*. Oxford University Press, Oxford, UK, 2000.

Sanda M. Harabagiu, George A. Miller, and Dan I. Moldovan. WordNet 2 - a morphologically and semantically enhanced resource. In *Proceedings of SIGLEX99: Standardising Lexical Resources*, pages 1–7, College Park, MD USA, 21–22 June 1999.

Vasileios Hatzivassiloglou and Kathleen McKeown. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the 31st annual meeting of the Association for Computational Linguistics*, pages 172–182, Columbus, Ohio USA, 22–26 June 1993.

Marti A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th international conference on Computational Linguistics*, pages 539–545, Nantes, France, 23–28 July 1992.

Marti A. Hearst and Gregory Grefenstette. A method for refining automatically-discovered lexical relations: Combining weak techniques for stronger results. In *Statistically-Based Natural Language Programming Techniques: Papers from the AAAI Workshop*, Technical Report WS-92-01, pages 72–80, Menlo Park, CA USA, 1992. AAAI Press.

Marti A. Hearst and Hinrich Schütze. Customizing a lexicon to better suit a computational task. In *Proceedings of the Workshop on Acquisition of Lexical Knowledge from Text*, pages 55–69, Columbus, OH USA, 21 June 1993.

John C. Henderson and Eric Brill. Exploiting diversity in natural language processing: Combining parsers. In *Proceedings of the Joint ACL SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 187–194, College Park, MD USA, 21–22 June 1999.

Ralph Hickok, editor. *Roget's II: the new thesaurus. Third edition*. Houghton Mifflin Company, Boston, MA USA, 1995. available from `http://www.bartleby.com/62`.

Donald Hindle. Noun classification from predicate-argument structures. In *Proceedings of the 28th annual meeting of the Assocation for Computational Linguistics*, pages 268–275, Pittsburgh, PA USA, 6–9 June 1990.

Lynette Hirschman, Jong C. Park, Junichi Tsujii, and Limsoon Wong. Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, 18(12):1553–1561, 2002.

Graeme Hirst. Near-synonymy and the structure of lexical knowledge. In *Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity. Papers from the* AAAI *Spring Symposium*, pages 51–56, Stanford University, CA USA, 27–29 March 1995.

Graeme Hirst and David St-Onge. Lexical chains as representations of context for the detection and correction of malapropisms. In Fellbaum (1998), chapter 13, pages 305–332.

James M. Hodgson. Informational constraints on prelexical priming. *Language and Cognitive Processes*, 6:169–205, 1991.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

Rebecca Hwa. Supervised grammar induction using training data with limited constituent information. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 73–79, College Park, MD USA, 20–26 June 1999.

Mario Jarmasz. Roget's thesaurus as a lexical resource for natural language processing. Master's thesis, Ottawa-Carleton Institute for Computer Science, University of Ottawa, Ottawa, Ontario Canada, July 2003.

Mario Jarmasz and Stan Szpakowicz. Roget's thesaurus and semantic similarity. In *International Conference on Recent Advances in Natural Language Processing*, pages 212–219, Borovets, Bulgaria, 10–12 September 2003.

Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th International Conference on Research in Computational Linguistics (ROCLING X)*, Academia Sinica, Taipei, Taiwan, 22–24 August 1997.

Hongyan Jing and Evelyne Tzoukermann. Information retrieval based on context distance and morphology. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 90–96, Berkeley, CA USA, 15–19 August 1999.

Eric Joanis. Automatic verb classification using a general feature space. Master's thesis, Department of Computer Science, University of Toronto, Toronto, Canada, October 2002.

Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Series in Probability and Mathematical Statistics. Wiley and Sons, New York, NY USA, 1990.

Frank Keller, Martin Corley, Steffan Corley, Lars Konieczny, and Amalia Todirascu. WebExp: A Java toolbox for web-based psychological experiments. Technical Report HCRC/TR-99, Human Communication Research Centre, University of Edinburgh, Edinburgh, UK, 26 November 1998.

Frank Keller, Maria Lapata, and Olga Ourioupina. Using the Web to overcome data sparseness. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 230–237, Philadelphia, PA, USA, 6–7 July 2002.

Adam Kilgarriff. Inheriting polysemy. In Patrick St. Dizier and Evelyne Viegas, editors, *Computational Lexical Semantics*, pages 319–335. Cambridge University Press, Cambridge, UK, 1995.

Adam Kilgarriff and Colin Yallop. What's in a thesaurus? In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, pages 1371–1379, Athens, Greece, 31 May–2 June 2000.

Oi Yee Kwong. Aligning WordNet with additional lexical resources. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 73–79, Montréal, Québec, Canada, 16 August 1998.

Sidney I. Landau. *Dictionaries: The Art and Craft of Lexicography*. Cambridge University Press, Cambridge, UK, 1989.

Thomas Landauer and Susan Dumais. A solution to Plato's problem: The latent semantic

analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240, 1997.

Maria Lapata. The automatic interpretation of nominalizations. In *Proceedings of the 17th National Conference on Artificial Intelligence*, pages 716–721, Austin, TX USA, 30 July – 3 August 2000.

Maria Lapata. *The Acquisition and Modeling of Lexical Knowledge: A Corpus-based Investigation of Systematic Polysemy*. PhD thesis, Institute for Communicating and Collaborative Systems, University of Edinburgh, 2001.

Maria Lapata and Chris Brew. Using subcategorization to resolve verb class ambiguity. In *Proceedings of the Joint ACL SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 266–274, College Park, MD USA, 21–22 June 1999.

Maria Lapata, Frank Keller, and Scott McDonald. Evaluating smoothing algorithms against plausibility judgements. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 346–353, Toulouse, France, 9–11 July 2001.

Maria Lapata and Alex Lascarides. A probabilistic account of logical metonymy. *Computational Linguistics*, 29(2):261–315, June 2003.

Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. In Fellbaum (1998), chapter 11, pages 265–283.

Changki Lee, Geunbae Lee, and Seo Jung Yun. Automatic WordNet mapping using word sense disambiguation. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 142–147, Hong Kong, 7–8 October 2000.

Lillian Lee. *Similarity-Based Approaches to Natural Language Processing*. PhD thesis, Harvard University, Cambridge, MA USA, 1997. *published as* TR-11-97.

Lillian Lee. Measures of distributional similarity. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 25–32, College Park, MD USA, 20–26 June 1999.

Lillian Lee. On the effectiveness of the skew divergence for statistical language analysis. In *Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics*, pages 65–72, Key West, FL USA, 4–7 January 2001.

Lillian Lee and Fernando Pereira. Distributional similarity models: Clustering vs. nearest neighbors. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 33–40, College Park, MD USA, 20–26 June 1999.

Geoffrey Leech, Roger Garside, and Michael Bryant. CLAWS4: The tagging of the British National Corpus. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 622–628, Kyoto, Japan, 5–9 August 1994.

M. Lesk and E. Schmidt. Lex–a lexical analyzer generator. Technical Report CSTR 39, AT&T Bell Laboratories, Murray Hill, NJ USA, 1975.

Michael Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of SIGDOC*, pages 24–26, Toronto, Ontario, Canada, 1986.

Beth Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL USA, 1993.

Hang Li and Naoki Abe. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244, June 1998.

Dekang Lin. Principle-based parsing without overgeneration. In *Proceedings of the 31st annual meeting of the Association for Computational Linguistics*, pages 112–120, Columbus, Ohio USA, 22–26 June 1993.

Dekang Lin. PRINCIPAR–an efficient, broad-coverage, principle-based parser. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 42–48, Kyoto, Japan, 5–9 August 1994.

Dekang Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th annual meeting of the Association for Computational Linguistics*, pages 768–774, Montréal, Québec, Canada, 10–14 August 1998a.

Dekang Lin. Dependency-based evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems*, pages 234–241, Granada, Spain, 28–30 May 1998b.

Dekang Lin. Extracting collocations from text corpora. In *Proceedings of the first Workshop on Computational Terminology*, Montréal, Québec, Canada, 15 August 1998c.

Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304, Madison, WI USA, 24–27 July 1998d.

Dekang Lin. Automatic identification of non-compositional phrases. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 317–324, College Park, MD USA, 20–26 June 1999.

Dekang Lin and Patrick Pantel. DIRT - discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328, San Francisco, CA USA, 26–29 August 26 2001a.

Dekang Lin and Patrick Pantel. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360, 2001b.

Dekang Lin, Shaojun Zhao, Lijuan Qin, and Ming Zhou. Identifying synonyms among distributionally similar words. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 1492–1493, Acapulco, Mexico, 9–15 August 2003.

Michael L. Littman, Greg A. Keim, and Noam Shazeer. A probabilistic approach to solving crossword puzzles. *Artificial Intelligence*, 134(1–2):23–55, January 2002.

Alpha K. Luk. Statistical sense disambiguation with relatively small corpora using dictionary definitions. In *Proceedings of the 33rd annual meeting of the Association for Computational Linguistics*, pages 181–188, Cambridge, MA USA, 26–30 June 1995.

Robert MacIntyre. North American News Text Supplement. Technical Report LDC98T30, Linguistic Data Consortium, Philadelphia, PA USA, 1998.

Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. The use of WordNet in information retrieval. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 31–37, Montréal, Québec, Canada, 16 August 1998.

Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. Complementing WordNet with Roget and corpus-based automatically constructed thesauri for information retrieval. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 94–101, Bergen, Norway, 8–12 June 1999.

Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA USA, 1999.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1994.

Margaret Masterman. The potentialities of a mechanical thesaurus. *MT: Mechanical Translation*, 11(3):369–390, July 1956.

Diana McCarthy, Bill Keller, and John Carroll. Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 73–80, Sapporo, Japan, 12 July 2003.

Scott McDonald. *Environmental determinants of lexical processing effort*. PhD thesis, University of Edinburgh, 2000.

Michael L. McHale. A comparison of WordNet and Roget's taxonomy for measuring semantic similarity. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 115–120, Montréal, Québec, Canada, 16 August 1998.

Rada Mihalcea and Dan I. Moldovan. eXtended WordNet: progress report. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 95–100, Pittsburgh, PA USA, 2–7 June 2001.

George A. Miller and Walter G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

Guido Minnen, John Carroll, and Darren Pearce. Robust applied morphological generation. In *Proceedings of the First International Natural Language Generation Conference*, pages 201–208, Mitzpe Ramon, Israel, 12–16 June 2000.

Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223, 2001.

Natalia N. Modjeska, Katja Markert, and Malvina Nissim. Using the Web in machine learning for other-anaphora resolution. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 176–183, Sapporo, Japan, 11–12 July 2003.

Dan Moldovan, Sanda Harabagiu, Marius Pasca, Rada Mihalcea, Roxana Girju, Richard Goodrum, and Vasile Rus. The structure and performance of an open-domain question answering system. In *Proceedings of the 38th annual meeting of the Association for Computational Linguistics*, pages 563–570, Hong Kong, 3–6 October 2000.

Jane Morris and Graeme Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48, March 1991.

Herbert C. Morton. *The Story of Webster's Third: Philip Gove's Controversial Dictionary and its Critics*. Cambridge University Press, Cambridge, UK, 1994.

Tom Morton. Grok tokenizer, 2002. part of the Grok OpenNLP toolkit.

Preslav I. Nakov and Marti A. Hearst. Category-based pseudowords. In *Companion Volume to the Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 70–72, Edmonton, Alberta Canada, 27th May–1 June 2003.

Vivi Nastase and Stan Szpakowicz. Word sense disambiguation in Roget's thesaurus using WordNet. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 17–22, Pittsburgh, PA USA, 2–7 June 2001.

Donie O'Sullivan, Annette McElligott, and Richard F.E. Sutcliffe. Augmenting the Princeton WordNet with a domain specific ontology. In *Proceedings of the Worksop on Basic Ontological Issues in Knowledge Sharing*, Montréal, Québec, Canada, 19–20 August 1995.

Sebastian Padó and Mirella Lapata. Constructing semantic space models from parsed corpora. In *Proceedings of the 41st annual meeting of the Association for Computational Linguistics*, pages 128–135, Sapporo, Japan, 7–12 July 2003.

Patrick Pantel and Dekang Lin. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th annual meeting of the Association for Computational Linguistics*, pages 101–108, Hong Kong, 3–6 October 2000.

Patrick Pantel and Dekang Lin. Discovering word senses from text. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 613–619, Edmonton, Alberta, Canada, 23–26 July 2002a.

Patrick Pantel and Dekang Lin. Document clustering with committees. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 199–206, Tampere, Finland, 11–15 August 2002b.

Marius Pasca and Sanda M. Harabagiu. The informative role of WordNet in open-domain question answering. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 138–143, Pittsburgh, PA USA, 2–7 June 2001.

Darren Pearce. Synonymy in collocation extraction. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 41–46, Pittsburgh, PA USA, 2–7 June 2001a.

Darren Pearce. Using conceptual similarity for collocation extraction. In *Proceedings of the 4th annual CLUK Research Colloquium*, Sheffield, UK, 10–11 January 2001b.

Ted Pederson. A simple approach to building ensembles of Naïve bayesian classifiers for word sense disambiguation. In *Proceedings of the 6th Applied Natural Language Processing Conference and the 1st meeting of the North American Chapter of the Association for Computational Linguistics*, pages 63–69, Seattle, WA USA, 29 April–4 May 2000.

Fernando Pereira, Naftali Tishby, and Lillian Lee. Distributional clustering of English words. In *Proceedings of the 31st annual meeting of the Association for Computational Linguistics*, pages 183–190, Columbus, Ohio USA, 22–26 June 1993.

Willian H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

Roy Rada, Hafadh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantics nets. *Transactions on Systems, Man, and Cybernetics*, 19(1):17–39, February 1989.

Adwait Ratnaparkhi. A maximum entropy part-of-speech tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142, Philadelphia, PA USA, 17–18 May 1996.

Philip Resnik. Using information content to evaluate semantic similarity. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, Montreal, Canada, 20–25 August 1995.

Jeffrey C. Reynar and Adwait Ratnaparkhi. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 16–19, Washington, D.C. USA, 31 March–3 April 1997.

Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 474–479, Orlando, FL USA, 18–22 July 1999.

Ellen Riloff and Jessica Shepherd. A corpus-based approach for building semantic lexicons. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124, Providence, RI USA, 1–2 August 1997.

Brian Roark and Eugene Charniak. Noun-phrase co-occurrence statistic for semi-automatic semantic lexicon construction. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th annual meeting of the Association for Computational Linguistics*, pages 1110–1116, Montréal, Québec, Canada, 10–14 August 1998.

Peter Mark Roget. *Thesaurus of English words and phrases*. Longmans, Green and Company, London, UK, 1911. available from `http://promo.net/pg/`.

Tony Rose, Mark Stevenson, and Miles Whitehead. The Reuters corpus volume 1 - from yesterday's news to tomorrow's language resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas de Gran Canaria, 29–31 May 2002.

Herbert Rubenstein and John B. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, October 1965.

Gerda Ruge. Automatic detection of thesaurus relations for information retrieval applications. In *Foundations of Computer Science: Potential - Theory - Cognition, Lecture Notes in Computer Science*, volume LNCS 1337, pages 499–506. Springer Verlag, Berlin, Germany, 1997.

Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, NY USA, 1983.

Geoffrey R. Sampson. *English for the Computer*. Oxford University Press, Oxford, UK, 1995.

Mark Sanderson and Bruce Croft. Deriving concept hierarchies from text. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 206–213, Berkeley, CA USA, 15–19 August 1999.

Beatrice Santorini. Part-of-speech tagging guidelines for the Penn Treebank project. Technical Report MS-CIS-90-47/LINC LAB 178, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA USA, July 1990.

Sabine Schulte im Walde. Clustering verbs semantically according to their alternation behaviour. In *Proceedings of the 18th international conference on Computational Linguistics*, pages 747–753, Saarbrücken, Germany, 31 July–4 August 2000.

Sabine Schulte im Walde. Experiments on the choice of features for learning verb classes. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, pages 315–322, Budapest, Hungary, 12–17 April 2003.

Hinrich Schütze. Context space. In *Intelligent Probabilistic Approaches to Natural Language*, number FS-92-04 in Fall Symposium Series, pages 113–120, Stanford University, CA USA, 25–27 March 1992a.

Hinrich Schütze. Dimensions of meaning. In *Proceedings of the 1992 conference on Supercomputing*, pages 787–796, Minneapolis, MN USA, 16–20 November 1992b.

Hinrich Schütze and Jan O. Pedersen. Information retrieval based on word senses. In *Proceedings of the 4th annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, NV USA, 24–26 April 1995.

Frank Smadja, Vasileios Hatzivassiloglou, and Kathleen R. McKeown. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38, March 1996.

Karen Spärck Jones. *Synonymy and Semantic Classification*. Edinburgh University Press, Edinburgh, UK, 1964/1986.

Karen Spärck Jones. *Automatic Keyword Classification for Information Retrieval*. Butterworths, London, UK, 1971.

C.M. Sperberg-McQueen and Lou Burnard, editors. *TEI P4: Guidelines for Electronic Text Encoding and Interchange*. Text Encoding Initiative Consortium, Oxford, Providence, Charlottesville, Bergen, 2002. XML Version.

Richard Sproat, Alan Black, Stanley Chen, Shankar Kumar, Mari Ostendorf, and Christopher Richards. Normalization of non-standard words. *Computer Speech and Language*, 15(3): 287–333, 2001.

Padmini Srinivasan. Thesaurus construction. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval. Data Structures & Algorithms*, chapter 9, pages 161–218. Prentice-Hall PTR, Upper Saddle River, NJ USA, 1992.

David St-Onge. Detecting and correcting malapropism with lexical chains. Master's thesis, Department of Computer Science, University of Toronto, Toronto, Canada, March 1995. Published as CSRI-319.

Manfred Stede. *Lexical semantics and knowledge representation in multilingual sentence generation*. PhD thesis, Department of Computer Science, University of Toronto, May 1996. Published as technical report CSRI-347.

Manfred Stede. The hyperonym problem revisited: Conceptual and lexical hierarchies in language generation. In *Proceedings of the First International Conference on Natural Language Generation*, pages 93–99, Mitzpe Ramon, Israel, 12–16 June 2000.

Mark Stevenson. Augmenting noun taxonomies by combining lexical similarity metrics. In *Proceedings of 19th International Conference on Computational Linguistics*, pages 577–583, Taipei, Taiwan, 24 August–1 September 2002.

Suzanne Stevenson and Paola Merlo. Automatic verb classification using distributions of grammatical features. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 45–52, Bergen, Norway, 8–12 June 1999.

Suzanne Stevenson and Paulo Merlo. Automatic lexical acquisition based on statistical distributions. In *Proceedings of the 18th international conference on Computational Linguistics*, Saarbrücken, Germany, 31 July–4 August 2000.

Alexander Strehl. *Relationship-based Clustering and Cluster Ensembles for High-dimensional Data Mining*. PhD thesis, Laboratory for Artificial Neural Systems, University of Texas at Austin, Austin, TX USA, 2002.

Neel Sundaresan and Jeonghee Yi. Mining the web for relations. In *Proceedings of the 9th International World Wide Web Conference*, Amsterdam, Netherlands, 15–19 May 2000.

Michael Sussna. Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the second international conference on Information Knowledge Management*, pages 67–74, Washington, DC USA, 1–5 November 1993.

T.T. Tanimoto. An element mathematical theory of classification. Technical report, I.B.M. Research, New York, NY USA, November 1958. *internal report*.

LOC. *Library of Congress Subject Headings*. U.S. Library of Congress, 26th edition, 2003.

NLM. *Medical Subject Headings*. U.S. National Library of Medicine, 2004.

Erik F. Tjong Kim Sang. Noun phrase recognition by system combination. In *Proceedings of the Language Technology Joint Conference ANLP-NAACL2000*, pages 50–55, Seattle, Washington, USA, 29 April–4 May 2000.

Erik F. Tjong Kim Sang, Walter Daelemans, Hervé Déjean, Rob Koeling, Yuval Krymolowski, Vasin Punyakanok, and Dan Roth. Applying system combination to base noun phrase identification. In *Proceedings of the 18th international conference on Computational Linguistics*, pages 857–863, Saarbrücken, Germany, 31 July–4 August 2000.

Davide Turcato, Fred Popowich, Janine Toole, Dan Fass, Devlan Nicholson, and Gordon Tisher. Adapting a synonym database to specific domains. In *Proceedings of the Workshop on Recent Advances in Natural Language Processing and Information Retrieval*, pages 1–11, Hong Kong, 7–8 October 2000.

Peter D. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the Twelfth European Conference on Machine Learning*, pages 491–502, Freiburg, Germany, 3–7 September 2001.

Peter D. Turney. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 417–424, Philadelphia, PA USA, 7–12 July 2002.

Peter D. Turney, Michael L. Littman, Jeffrey Bigham, and Victor Shnayder. Combining independent modules to solve multiple-choice synonym and analogy problems. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pages 482–489, Borovets, Bulgaria, 10–12 September 2003.

Hans van Halteren, Jakub Zavrel, and Walter Daelemans. Improving data driven wordclass tagging by system combination. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th annual meeting of the Association for Computational Linguistics*, pages 491–497, Montréal, Québec, Canada, 10–14 August 1998.

C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, UK, 1979.

Ellen M. Voorhees. Using WordNet for text retrieval. In Fellbaum (1998), chapter 12, pages 285–303.

Grady Ward. *Moby Thesaurus*. Moby Lexicon Project, 1996. available from `http://etext.icewire.com/moby/`.

Julie Weeds and David Weir. A general framework for distributional similarity. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 81–88, Sapporo, Japan, 11–12 July 2003.

Dominic Widdows. Unsupervised methods for developing taxonomies by combining syntactic and statistical information. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 276–283, Edmonton, Alberta Canada, 27 May–1 June 2003.

Robert Wilensky. Extending the lexicon by exploiting subregularities. In *Proceedings of the 13th International Conference on Computational Linguistics*, Helsinki, Finland, 20–25 August 1990.

Yorick Wilks. Language processing and the thesaurus. In *Proceedings of the National Language Research Institute Fifth International Symposium*, Tokyo, Japan, 1998.

Ian H. Witten, Alistair Moffat, and Timothy C. Bell. *Managing Gigabytes. Compressing and Indexing Documents and Images*. Academic Press, San Diago, CA USA, 2nd edition, 1999.

Zhibiao Wu and Martha Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd annual meeting of the Association for Computational Linguistics*, pages 133–138, Las Cruces, New Mexico, USA, 27–30 June 1994.

Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, Zurich Switzerland, 18–22 August 1996.

Jinxi Xu and W. Bruce Croft. Corpus-based stemming using cooccurrence of word variants. *ACM Transactions on Information Systems*, 16(1):61–81, January 1998.

David Yarowsky. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of the 14th international conference on Computational Linguistics*, pages 454–460, Nantes, France, 23–28 July 1992.

Diana Zaiu Inkpen and Graeme Hirst. Building a lexical knowledge-base of near-synonym differences. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 47–52, Pittsburgh, PA USA, 2–7 June 2001.

Diana Zaiu Inkpen and Graeme Hirst. Acquiring collocations for lexical choice between near-synonyms. In *Proceedings of the Workshop on Unsupervised Lexical Acquisition*, pages 67–76, Philadelphia, PA, USA, 12 July 2002.

Diana Zaiu Inkpen and Graeme Hirst. Near-synonym choice in natural language generation. In *International Conference on Recent Advances in Natural Language Processing*, pages 204–211, Borovets, Bulgaria, 10–12 September 2003.

Ladislav Zgusta. *Manual of Lexicography*. Publishing House of the Czechoslovak Academy of Sciences, Prague, Czech Republic, 1971.

George K. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley, Cambridge, MA, USA, 1949.