Automatic Acquisition of Lexical Semantic Knowledge from Large Corpora: The Identification of Semantically Related Words, Markedness, Polarity, and Antonymy

Vasileios Hatzivassiloglou

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

1998

ABSTRACT

Automatic Acquisition of Lexical Semantic Knowledge from Large Corpora: The Identification of Semantically Related Words, Markedness, Polarity, and Antonymy

Vasileios Hatzivassiloglou

Lexical semantic knowledge is useful, even indispensable, for many natural language processing applications. Yet, traditional approaches for acquiring this knowledge manually are expensive and cannot easily handle the requisite domain dependence. In this dissertation, I address four closely related problems from lexical semantics, describing a fully automatic system that extracts information about semantic groups and scales from large free-text corpora. The system forms groups of semantically related terms such as {*cold*, *warm*, *hot*}, {*final*, *preliminary*}, and {*court*, *jury*, *law*, *regulation*}. Using gradability indicators, it identifies those of the groups that are actually linguistic scales, i.e., contain terms that can be –linearly ordered on the basis of semantic strength. Scalar groups are further partitioned into two subgroups according to evaluative orientation, distinguishing between positively loaded terms (e.g., *beautiful*, *ingenious*, *unbiased*) and their negative counterparts (e.g., *ugly*, *plain*, *lazy*). Finally, the semantic orientation of each subgroup is identified. Combining the above four stages results in an automatic method for the retrieval of possibly domain-dependent pairs of antonyms. All this information is actively learned from the corpus; the system does not access any type of stored information about words

such as dictionaries, thesauri, or similar databases.

I have adopted a statistical approach that combines both supervised and unsupervised learning methods and is informed by linguistic models of the data and the tasks at hand. I rely on robust, non-parametric statistical methods; multiple knowledge sources justified by linguistic analyses; and shallow syntactic and morphological processing during information extraction. I describe and justify the linguistic sources, and present the results (sometimes quite unexpected) of experimental studies that are designed to validate related hypotheses made in the linguistics literature. I also present a novel evaluation method which simultaneously employs multiple reference models without inducing a single "best" model, and results produced for several collections of adjectives and nouns. Finally, I present evidence of the strengths of the hybrid linguistic-statistical approach, and discuss applications of the system's output to language problems.

Contents

List of Figures

List of Tables

Acknowledgements

A number of people, at Columbia University and elsewhere, have been instrumental in improving the position of the present work on the scale of dissertation quality, contributing ideas, suggestions, and feedback on various issues from concepts to implementation to writing. Others helped me stay focused on this work during the six and a half long years that were needed for its completion.

As in many other endeavors, pure luck played a significant role in the development of the thesis: I had the extremely good fortune to work under the supervision of Kathy McKeown. Kathy was willing to expand her research interests to match my own, and explore new areas and paradigms that were novel to both of us. She supported me throughout investigations that did not fall exactly in line with the conventional wisdom and the funding priorities at the time. She fought against my long sentences, steered me away from being immersed too much in the details, and patiently accommodated my delays when I insisted on thoroughly exploring some side topics. Most importantly, she taught me about the big picture and how to convey that to others. For her unwavering support, patience, informed guidance, advice, and genuine niceness, I am most grateful to her.

I am also grateful to the other members of my dissertation committee: Ken Church, John Kender, Mitch Marcus, and Shree Nayar. They got involved with the research work at an early stage and provided valuable feedback, both during my thesis proposal and several times thereafter. Their advice allowed me to improve the work, concentrate on its most important aspects, and provide convincing evidence about its practical, as well as theoretical, significance.

The members of the natural language group at Columbia University helped me with their daily input and comments as the present work matured. At an early stage, Frank Smadja guided me towards some of the problems addressed in this thesis, whose significance he had noted in his own dissertation. Although I ended up following quite a different approach from the one he initially suggested, I am indebted to him for our long discussions. Michael Elhadad and Jacques Robin provided early models of erudition, intellectual curiosity, and perseverance I could look up to. Working in a multi-disciplinary research area that combines computer science, linguistics, and statistics, I relied on the expertise and formal training of several linguists in our group; I want to thank particularly Rebecca Passonneau (who first showed me that linguists can count, too), Judith Klavans, and Nina Wacholder. My fellow students turned empirical statisticians, Eric Siegel and Pascale Fung, were there for suggestions and explorations of distributions, hypotheses, and tests. Thanasis Tsantilas lent me a hand whenever I came up with computational complexity and algorithmic questions, and Dragomir Radev and Tony Weida shared with me their knowledge on several technical, implementation issues.

In addition to the regular members of our group, I benefited from discussions with several other people who visited with us at Columbia or whom I met at other venues. I thank Ido Dagan, Marti Hearst, Christian Jacquemin, Fred Jelinek, Slava Katz, Diane Litman, Johanna Moore, Philip Resnik, Jeff Siskind, Evelyne Tzoukermann, and Dekai Wu for their valuable input. My outside work with Kevin Knight and Edward Hovy at the Information Sciences Institute of the University of Southern California on machine translation and language modeling, and with Karen Kukitch at Bellcore on concept aggregation, helped broaden my interests and my understanding of the field, and thus materially contributed to the overall perspective of the present dissertation.

Several people donated their time to prepare evaluation models for the various experiments reported in this thesis, and to them I extend my thanks: Michael Elhadad, David Evans, Pascale Fung, Chih-Peng Hung, Hongyan Jing, Min-Yen Kan, Judith Klavans, Diane Litman, Hsin Liu, Kathy McKeown, Gregory Sean McKinley, Tom O'Donnell, Shimei Pan, Rebecca Passonneau, Dragomir Radev, Jacques Robin, Carl Sable, Barry Schiffman, James Shaw, Eric Siegel, Frank Smadja, David Torok, Tony Weida, David Worenklein, and David Yang. I am also thankful to the Columbia students who worked under my supervision in directed research projects, in particular to Hsin Liu who investigated many of the alternative algorithms discussed in Chapter 2.

Some of my experiments relied on text corpora and previously developed tools that were kindly made available to me by their authors. I thank Ken Church and the AT&T Bell Laboratories for providing the PARTS part-of-speech tagger; Karen Kukich and Frank Smadja for making a corpus of Associated Press newswire reports available to me; Donald Hindle, Diane Litman, and again the AT&T Bell Laboratories for parsing this corpus with Hindle's FIDDITCH parser; and Steve Johnson for extracting, preprocessing, and arranging for my access to thousands of hospital discharge summaries. I would also like to acknowledge the support of the funding agencies who made this work possible: the National Science Foundation (under contract GER-90-24069), the Office of Naval Research (under contract N00014-95-1-0745), the Advanced Research Projects Agency (under contract N00014-89-J-1782 jointly with ONR), and the New York State Science and Technology Foundation / Center for Advanced Technology (under contracts NYSSTF-CAT(91)-053, NYSSTF-CAT(94)-013, NYSSTF-CAT(95)-013, and NYSSTF-CAT(96)-013, and also under the auspices of the Columbia University CAT in High Performance Computing and Communications in Healthcare). Any opinions, findings, conclusions, or recommendations are those of the author and do not necessarily reflect the views of the above agencies and in particular of the New York State Science and Technology Foundation.

During the long period of development of the present work, I relied on several good friends for encouragement, support, and a good time. Thanks to Apostolos Dailianas, Tom O'Donnell, Pascale Fung, Stelios Pantelopoulos, Maria Papadopouli, Jacques Robin, Ioannis Stamos, and Gong Su, for all the pleasant distractions; and to Jenny Stergioti for the summers spent on the Greek islands. I also thank my sister Polina and brother-in-law Yiannis Fovos, for the family moments I shared with them.

Finally, my parents, Theofilos and Lela, deserve a special mention. They supported and constantly encouraged me in pursuing an academic career, and helped me with the difficulties I faced when doing so. Although separated from me by thousands of miles, they continued to stand by me during my graduate studies in the United States. They never failed to take pride in my achievements and encourage me to continue, aiming for higher goals. Their contribution to this work is intangible but invaluable, and I feel most indebted to them for their help and love, especially during the last, most difficult months of intense writing.

To my parents,

Θεόφιλο and Λέλα

Synopsis

What is it all about, and why The dissertation's goal is to enable access to *lexical semantic* information, that is, information that has the following two defining characteristics: First, it is information *local* to specific words which cannot be predicted or synthesized from smaller units. This locality of information can be extended to certain kinds of phrases, such as idioms and technical terms whose meaning cannot be derived compositionally. Second, lexical semantic information pertains to the *meaning* of the words, and the semantic relationships between them, rather than to constraints on their combinations (syntactic or lexical) or their surface properties (morphological or phonological). Such information includes relationships such as synonymy, antonymy, hyponymy (the relationship of a more specific term to a more general one), and meronymy (the relationship of part to whole).

Lexical semantic knowledge of this kind is useful for a variety of natural language processing tasks, including language modeling, understanding, and generation (see Section 1.10 and Chapter 11 for detailed discussions of applications). We can access some of this lexical semantic knowledge using existing sources such as machine-readable dictionaries (originally compiled for human use), thesauri, or manually constructed knowledge sources specifically designed for computational use (e.g., WordNet [Miller *et al.* 1990]). However, these sources share several defects, including incompleteness (both in terms of the words they provide entries for and in terms of the types of semantic relationships they contain); lack of domain specificity and adaptability (which limits their usefulness in many applications where specific meanings of words in a particular domain are desired); cost of building and maintenance; and potential for bias (especially with the lack of a formal model for the semantic processes they represent).

In this dissertation, I suggest an alternative approach: Rather than relying on knowledge sources previously constructed by human lexicographers and linguists, I discuss methods for the *automatic* construction of such sources. The methods I present automatically learn components of the meaning of words, exploiting two underlying principles: The use of a *large corpus* (without any kind of semantic or syntactic annotation) as a source of positive and negative examples, combined with *shallow linguistic knowledge*. The latter knowledge, in the form of general, non-lexical rules, provides additional sources of information, refines the results of the statistical inference process, and transforms and filters the input data to increase accuracy and learning efficiency.

The automatic learning approaches of the sort described above are especially suitable to the task of lexical semantic knowledge acquisition, because, first, *lexical* knowledge items are much more numerous than grammatical or morphological items (by definition, lexical knowledge is the knowledge that is exceptional for each word and cannot be generalized across broad word classes). Furthermore, *semantic* information is more domain-dependent than syntactic or morphological information; thus, the acquisition task is more critical and needs to be repeated more frequently for semantic knowledge.

Of course, lexical semantics covers a wide variety of topics, and it would have been too ambitious to attempt to present methods that extract knowledge for all of them. Instead, I have focused on several specific tasks, dealing with particular types of lexical semantic knowledge:

1. Separate a set of words from a particular syntactic category into groups of *semantically related* terms.

2. Decide which of these groups of semantically close terms are *scalar*, and which contain *complementary* terms.

3. Separate the scalar groups into subgroups of different *semantic orientation*.

4. Identify the *positive* and *negative* subscale in each scalar group so divided.

The remainder of this summary examines each of these tasks more closely, presenting an overview of the methods and highlights of the results.

Finding groups of semantically related words Semantic relatedness is one among many linguistic concepts that are hard to define formally, especially as there is no widely acceptable formal semantic theory that provides precise descriptions of "meaning". In light of this lack of a formal definition, I have relied on an indirect one, that nevertheless has been verified through the confluence of judgements of human evaluators and the automated system's output. I consider two words as semantically related if they are synonyms or near-synonyms (e.g., *deadly–fatal*), antonyms (e.g., *positive–negative*), or related through meronymy (e.g., *wheel–car*), hyponymy (e.g., *doctor–professional*) or co-hyponymy (e.g., *doctor–lawyer*). Words that can substitute for each other in the same semantic context, or describe the same property in different degrees, or refer to the same underlying concept are also considered semantically related; for example, {*cold*, *warm*, *hot*} and {*court*, *jury*, *judge*}.

A particular class of semantically related words are the *scalar* words. These are the members of a set of words that all express a common property in different degrees and can be ordered according to informativeness or strength with regard to that property. The {*cold*, *warm*, *hot*} example given above is (part of) one such linguistic scale describing different degrees of temperature. Linguistic scales have additional properties: They typically can be divided into a positive and a negative subscale (e.g., {*warm*, *hot*} and {*frigid*, *cold*}); terms are ordered within each subscale; and their members obey additional pragmatic constraints such as scalar implicature (see Section 1.7).

With the broad definition of semantic relatedness given above, many words can be deemed related to many other words, but in different degrees and through different types of relationships. Determining simply if two words are or are not related is thus not reflective of how well an automated process can distinguish between these different states of relatedness. To address this concern, I have selected as the primary task for the semantic relatedness component of the thesis not just to identify semantically related words, but also to automatically form the most coherent classes according to semantic relatedness. In this manner, the word *nurse* should be placed in the same group with *doctor* (through their strong mutual paradigmatic substitutability) and not with *lawyer*, although all three words are more weakly semantically related as co-hyponyms of *professional*. The output of the classification process is dependent on the entire set of terms that is partitioned into semantic groups. In this manner, we can vary the degree of semantic coherence in that set and examine the performance of the system on semantic relationships of varying strength.

The algorithm I have developed for placing words into semantic groups involves five stages (Figure 1.4 on page 64 depicts the architecture of this part of the system): First, relevant data for independent indicators of similarity or dissimilarity are extracted from the corpus. This is distributional evidence, such as, for example, what nouns are modified by each adjective that is clustered. The relevant data also includes negative examples, such as adjective-adjective or noun-noun pairs in specific syntactic relationships that indicate that the two words do *not* belong in the same semantic group. At this substage, shallow linguistic knowledge is used to improve the accuracy of the information extraction process through a finite-state parser, and to transform and filter the evidence via the application of morphology and spell-checking modules.

Once the data for the words we are clustering has been collected, a measure of similarity is computed separately for each indicator. An overall measure of similarity between any two words of interest combines the partial similarity values, according to the expected accuracy of each of them. I utilize Kendall's τ, a robust non-parametric estimator of distributional similarity to measure the extent that distributions of modified words are related for two of the clustered words. The overall similarity measure is linearly transformed to a measure of dissimilarity at the fourth stage, and drives a non-hierarchical clustering algorithm based on the *exchange method*. This last stage determines the optimal partition of the words in the input set that minimizes the average within-cluster dissimilarity. The details of the algorithms and the mathematical formulae for the similarity computation and clustering stages are given in Chapter 2. The same chapter provides linguistic justification for the positive and negative similarity indicators.

I have run the group finding component of the system on several collections of adjectives and nouns, using two different corpora as the source for the distributional data. The two

• foreign global international	• common preferred
• current future initial past previous	• cost expense fee
• affluent impoverished prosperous	• analysis report study
• deadly fatal	• acquisition purchase sale
• heavy light	

Figure S.1: A representative sample of (correct) semantic groups of adjectives and nouns that have been automatically constructed by the system.

corpora contained 8 and 21 million words, respectively; the sets of clustered words included five sets of adjectives (containing up to 3,243 words) and four sets of nouns (containing up to 1,893 words). I have experimented with large sets, varying the frequency and semantic specificity (i.e., polysemy or lack thereof) of the words in each set, to establish the scalability of the approach and show that the size of the problem warrants an automatic solution.

Figure S.1 shows some of the word groups that were automatically constructed by the system during these experiments.[1] These include groups of complementary, non-scalar terms (e.g., the first two groups in Figure S.1); groups of scalar terms with positive and negative subscales (e.g., {*affluent*, *impoverished*, *prosperous*}); groups of synonyms (e.g., {*deadly*, *fatal*}); groups of antonyms (e.g., {*heavy*, *light*}); and groups of related nouns (e.g., the last three groups in Figure S.1). The group {*common*, *preferred*} in Figure S.1 merits some additional discussion, as it is an example of a domain-dependent semantic relationship that the system successfully retrieves: while these two words may seem an odd pair in everyday English, they are indeed strongly related in financial corpora (from which the system was

[1]As befits an executive summary, all groups shown here are correct ones. For a more representative view of the system's actual performance, see Figures 4.6–4.12 and 4.19–4.22 on pages 177–178, 180–182, and 198–200.

trained) as they refer to two different types of stocks.

Evaluation To formally evaluate the performance of the system on the semantic grouping task, I have developed a new evaluation methodology, since the resulting partitions of word sets are structured and the decisions made by the clustering component are interdependent (e.g., if words A and B, and words A and C, are considered to be related and placed in the same group, so must be words B and C). I start by arguing that the inexact definition of the task necessitates the construction of models by humans independently of the system's output and the comparison of these models to the actual output, rather than relying on internal measures of performance or on grading the system's output. I present procedures for converting the structure implied by a partition to a list of "yes"–"no" decisions to which the standard information retrieval evaluation measures of precision, recall, and F-measure can be applied. Further, I generalize these measures to *simultaneously* take into account multiple, disagreeing reference models and automatically weigh the significance of each decision in the overall score according to the variance of opinion for that particular decision point in the various models.

This generalization of the evaluation methodology to multiple models is based on the assumption that for some linguistic problems, competing (and incompatible) solutions may be simultaneously correct. Whether this view of the models is adopted or not, the evaluation techniques developed for the thesis offer several advantages over earlier approaches that implicitly construct a single model solution from multiple reference models (such as the *majority method* [Gale *et al.* 1992a]). In particular, this new approach allows for decision points for which there is greater agreement in the models to be automatically (and proportionally to the extent of the agreement) weighed as more important for the system to find; and it reflects the disagreements between the models as lower performance ceilings for the evaluated system.

I have also addressed issues related to interpreting the meaning of the evaluation scores, i.e., *meta-evaluation* issues. Through Monte-Carlo simulation, I establish a lower performance bound as the score that would be achieved by randomly assigning the words into groups. The density of this distribution is not theoretically known; Figure 4.14 on page 193 shows a smoothed density for the F-measure score empirically determined through the simulation process. Not only is this useful for determining a baseline for performance comparisons, but the shape of the density function indicates how hard it is to progress into higher score ranges.

At the other end of the performance spectrum, I use the average score when each human-constructed model is evaluated against all other models as an upper bound for system performance. Note that the evaluation process axiomatically assumes that the system can never be better than the humans; thus, the system will be penalized if it actually discovers semantic relationships that are valid but not identified in the human models (possibly because they are domain-dependent).

The conversion of a structured partition to a list of "yes"–"no" decisions and the introduction of multiple reference models both introduce non-linearities in the score, i.e., situations where a small change in the output results in a disproportionately large change in the evaluation measure. A set of additional score transformations accounts for these non-linearities and for interdependencies between the two types of errors (false positives and false negatives). These transformations are discussed in detail, along with the other components of the evaluation methodology, in Chapter 3.

Applying these evaluation techniques to the partitions automatically constructed by the system resulted in F-measure (i.e., combined precision/recall) scores of 70% to 90% of the average score obtained by the humans on the same problem instance for sets of adjectives, and 78% to 98% of the average human score for sets of nouns. These scores do not include the transformations that correct for non-linearities and interdependencies; when the latter are incorporated into the scores, the average F-measure for the system rises to more than 85% of the human score for adjectives and more than 90% for nouns.

Identifying scalar words To identify scalar words, I rely on the close correlation between scalarity and gradability. The latter is the characteristic property of words that can express different degrees of a property, through explicit modification or inflection (e.g., *small–smaller–very small–not so small*) or through an implicit norm (e.g., compare the size of a *small planet* and a *large molecule*).

I use two text-level indicators of gradability, inflection for degree and modification by certain grading adverbials. Because not every occurrence of a word in an apparently gradable context implies that the word is generally gradable, I train a logistic regression (log-linear) model using reference labels for gradability from the COBUILD dictionary [Sinclair 1987b]. This avoids exceptions such as *very dead* or *redder and redder*.

Using the automatically determined weights for the two indicators on a collection of 496 adjectives, I obtained classification accuracy of over 90%. Furthermore, when the gradability decisions for individual words are combined with the earlier grouping stage to decide if a semantic group is a scale, accuracy rises to 98–99%. This is due to *decision aggregation*, i.e., the fact that most of the individual decisions within a group will be correct ones. In addition, erroneous decisions frequently receive marginal confidence values that are subsumed by the confidence values of the correct decisions in the same group. Chapter 6 has the details on this task.

Predicting semantic orientation In order to identify special cases of semantically related scalar words, such as antonyms, I have designed methods that partition a scalar group into two different subscales. Elements of each subscale express the same *orientation* (i.e., positive versus negative, or present versus absent) of the underlying property that the entire scale describes. There are several subtle distinctions between types of semantic orientation, discussed in Section 7.1; I have chosen to adopt an *evaluative* criterion for defining orientation, where modifier words are assigned positive or negative orientation if their use ascribes in general a positive or negative quality to the modified item (compare, for example, *beautiful* and *ugly* under this criterion). Such a definition subsumes several other types of semantic orientation, and in an empirical study I found that human subjects assigned with remarkable consistency (89–97% agreement) orientation labels to a sample of 500 adjectives, establishing that evaluative semantic orientation is indeed an objective linguistic property.

Because it is not easy to directly determine the orientation of a given word in isolation, I have developed an indirect approach by which the sameness or difference in orientation between two words is established. The method primarily uses information from conjunctions where the two words appear; using the type of conjunction, the kind of modification (attributive, predicative, etc.), the presence or absence of negation, and the number feature of the modified noun as predictors, a log-linear model was automatically trained for a set of 1,336 adjectives. With the added information offered by the very few but highly accurate pairs of words with an overt morphological negation relationship (e.g., *accurate–inaccurate*), the method predicts whether a given conjunction and syntactic context indicate same or different orientation with 82% accuracy. Then these indications are aggregated by an optimizing clustering algorithm that separates the words into two large groups, corresponding to the positive and negative orientations. The overall performance of the method depends on the density of observed conjunctions between words; in my experiments, it ranged between 78% when each word appears in just two conjunctions to more than 92% when each word appears in five or more conjunctions. Monte-Carlo simulation experiments in artificial sets of conjunctions (Figure S.2) show that performance increases rapidly as more links between the words become available and reaches 100% with a modest number of conjunctions per word.

The clustering stage produces two groups of words of (hopefully) different orientations, but does not label them as positive or negative. For this task, I draw on the close association between semantic orientation and *semantic markedness*. I performed an analysis of various predictors for semantic markedness proposed in the linguistics literature, with some surprising findings: several of the suggested predictors did not work any better than chance, and a very simple predictor, text frequency, performed almost as well as the most complex model that synthesized several independent predictors. Using the latter model, I am able to assign markedness labels with 82% accuracy. But when this is translated to assigning orientation labels to reasonably large word groups, the method always predicted the correct label because of decision aggregation across the group members.

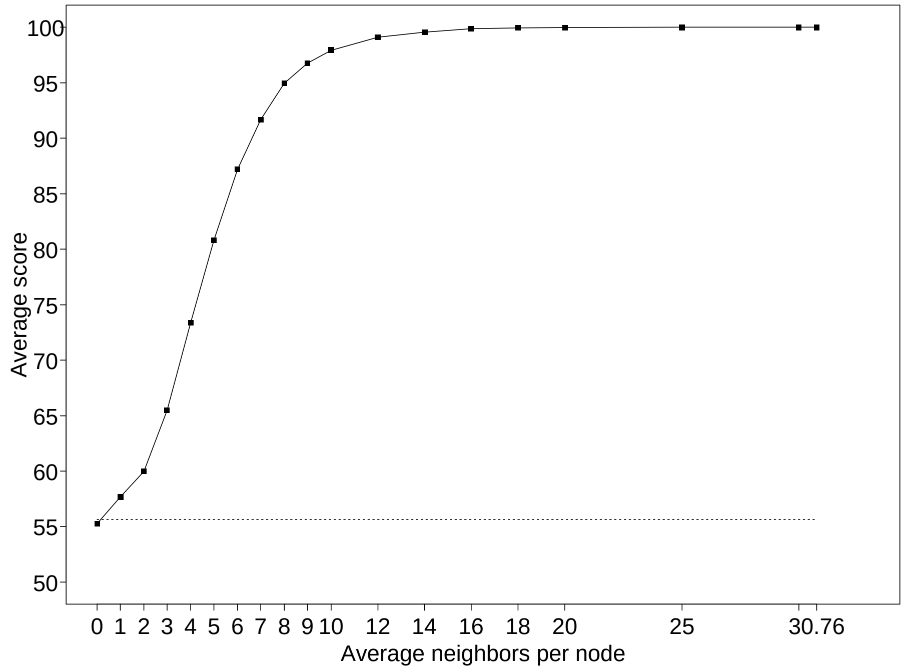


Figure S.2: Simulated performance of the semantic orientation classification subsystem as the number of conjunctions per word varies. The results shown were obtained with an average link classification accuracy (as same or different orientation) of 80%, slightly less than what is obtained in experiments involving real word sets.

A case study: The Deese antonyms To demonstrate how the various components of the thesis can be used together to determine sets of synonyms and antonyms from unmarked corpus data, I applied them to a well-known set of antonymous adjectives, the Deese set. The Deese set contains 75 high-frequency adjectives in 40 pairs of antonyms. It was constructed through human-specified experimental association data by Deese in the early sixties, and has since become a standard benchmark for the study of antonymy [Deese 1964; Justeson and Katz 1991; Grefenstette 1992a].

The semantic groups constructed by the present system can be directly mapped to the word associations that are implied by the Deese pairings (e.g., *active* should be in the same group with *passive*, but not with *large*). Using just the first stage of the system to partition the Deese words, I obtain groupings with about 35% recall and 25% precision, in terms of recovered associations. Note that in these experiments, associations that the system finds that are not of an antonymous nature or are not listed among Deese's pairs, are considered false positives. In some of these cases, the system actually finds a valid semantic association, e.g., *good–poor*. Nevertheless, we can improve the measured performance scores by combining information from multiple partitions of these words, each with a different number of semantic clusters. The combining algorithm produces an *association strength* for every word pair, and allows the generation of a list of *candidate associated words* for each word in the Deese set. For example, the top ten associations recovered for the Deese set are, in order: *high–low*, *inside–outside*, *right–wrong*, *heavy–light*, *strong–weak*, *narrow–wide*, *large–small*, *back–low*, *hard–sad*, and *fast–slow*; of those, only *back–low* and *hard–sad* are not among the Deese pairs.

When only the top candidate for each word is considered, this procedure results in 42% precision and recall; if five or ten candidates are examined for each word, the correct answer according to the Deese set is recovered in 64% and 74% of the cases, respectively. Thus, the method can be effectively used in a new specialized domain to present a reasonably accurate and not too long list of candidate associated words to a domain expert who would then select the correct answers, rather than having to produce them from scratch.

The produced lists of candidate associated words can be further refined by using the gradability and semantic orientation subsystems. Operating on the assumption that antonymous words should be either both gradable or both non-gradable, and if gradable, of opposite semantic orientations, I remove words that violate these criteria from the list of candidate antonyms (with additional constraints detailed in Section 9.5). This step improves both precision and recall by 10–20%, resulting in accurately recovering the correct antonym in 61% of the cases when three candidates are proposed and in 73% of the cases when five candidates are proposed. Examples of antonyms correctly proposed as the top selection for specific members of the Deese set include *passive* for *active*, *wide* for *narrow*, and *black* for *white*, and vice versa; Table 9.8 on pages 373–377 has the full details.

Importance of linguistic knowledge Throughout the dissertation, I argue that selected, shallow linguistic knowledge that is relatively easy to extract or model can significantly affect the accuracy of learning methods that use text corpora as input. In Chapter 5 I present a designed experiment that measures the effect of four linguistic modules on the word grouping subsystem, using three different adjective sets and four different corpus sizes. The performance gained by the introduction of these modules is statistically significant at the 0.1% level for three of the four modules. Furthermore, the gain is quantitatively significant as well, averaging to a 36% improvement in the F-measure classification score.

Summary In the preceding brief synopsis, I introduced the topics covered by the dissertation and highlighted the techniques I employed to address the various problems. I also presented glimpses of the experimental results, which are more fully documented in the chapters that follow. I have described a portable, modular, and extensible system for lexical knowledge acquisition that incorporates novel applications of statistical and machine learning techniques, as well as algorithms and knowledge sources for new kinds of lexical semantic information. The system adopts a flexible architecture utilizing multiple knowledge sources and shallow linguistic knowledge in addition to the statistical evidence, and novel evaluation techniques suitable for scoring solutions with multiple, disagreeing reference models. It also has a number of practical applications, including class-based smoothing in language modeling, domain-dependent sense disambiguation, medical terminology extraction and verification, handling argumentative force, and interpreting scalar implicature and negation. For more evidence on these claims and all the details, please turn the page to begin reading the thesis.

Chapter 1

Introduction

1.1 Overview

Lexical semantics, as a field, presents many unsolved problems that are hard to formalize or address with the methods of traditional linguistics. Few of these problems have been approached with the data-intensive methods recently applied with much success in other areas of computational linguistics. In this work, I concentrate on the automatic acquisition of lexical semantic knowledge for several closely related problems.

Lexical semantics covers foremost the meaning of individual words; it also covers the meaning of phrases to the extent that the latter cannot be predicted on the basis of compositionality. Traditional approaches to lexical semantic knowledge acquisition, based on the codification of such knowledge by hand, suffer from a number of disadvantages:

- Dictionaries (or other databases of lexical descriptions compiled by humans) are largely incomplete, both in terms of the number of words and expressions they cover and in terms of the types of semantic knowledge they provide for each word.

- Compiling large lexicons by hand is prohibitive in terms of the human effort required.

- Lexical semantic knowledge is sensitive to domain changes. As a consequence, either the domain sublanguage must be ignored or a new lexicon must be constructed for each new domain.

- Static dictionaries are difficult to maintain; new terms are often added to the language and older terms fall into disuse.

- While several formal models of lexical semantic knowledge have been proposed (see Section 1.6), at the present time there is no sufficiently broad consensus on which of these theories should be adopted. The lack of such a broadly accepted formal theory for lexical semantics makes the construction of lexicons more difficult, and leaves the whole process prone to omissions, inconsistencies, and bias on the part of the lexicographer. It is no surprise that little semantic information is included in many of the dictionaries designed for human use, and that any two dictionaries are likely to have significant differences in their descriptions of the same words.

Yet, lexical semantic knowledge is indispensable for natural language processing systems, whether human or computer-based, if at least we are interested in more than superficial understanding of the text that is analyzed or generated. Automated methods for lexical semantic knowledge acquisition can be useful in the development of lexical semantic theories, by providing an experimental basis for testing proposed models. Besides their theoretical significance, such methods are also useful for practical reasons: most natural language processing applications that need access to descriptions of the meaning of words will benefit from automated methods for acquiring this knowledge.

This dissertation addresses issues related to the organization of the language vocabulary into groups of semantically related words and the further analysis, subdivision, and labeling of these groups. First, given a list of words with roughly the same syntactic behavior (i.e., words that all have the same part of speech), we want to automatically separate them into groups containing semantically related words, in such a way that words that are most closely related in meaning fall into the same group, and words that are unrelated are placed into different groups. These groups generally contain words that relate to the same property or its opposite. If the words in a group can be independently graded with other words (e.g., *good* → *very good*) and ordered according to how strongly they associate to the property underlying the entire group, then the group is a *scale*; otherwise, the group contains complementary, ungradable terms. For example, the group {*cold*, *cool*, *lukewarm*, *warm*, *hot*} is an adjectival scale describing various levels of temperature, while the collections {*blue*, *green*, *red*} and {*public*, *private*} are non-scalar groups of complementary adjectives. The second task addressed in the dissertation is how the scalar groups can be automatically identified.

Scalar groups are further subdivided according to evaluative criteria, so that all positive terms (e.g., *beautiful* in the opposition *beautiful–ugly*) go into one side and all negative terms (e.g., *ugly* in the above example) go into the other side; terms from different sides of the same group are antonyms (in a broad sense), while terms from the same side exhibit identical semantic orientation. The final step is to identify which side is which, and thus actually label the subgroups as positive or negative. Through the application of the above four steps we reach a classification of the original words into tight semantic classes, and achieve the extraction of semantic relationships such as antonymy. In the future, additional techniques can be applied to the words in each scalar subgroup so that they be partially ordered according to their semantic strength, i.e., the degree to which they express the semantic property that underlies the entire group. For example, starting from the set of words {*beautiful*, *cold*, *economic*, *financial*, *hot*, *ugly*, *warm*}, the system should formulate three semantic groups: {*beautiful*, *ugly*}, {*cold*, *hot*, *warm*}, and {*economic*, *financial*}; it should characterize the words in the last group as ungradable complementaries but label the other two groups as adjectival scales; it should separate {*beautiful*, *ugly*} into the subscales {*beautiful*} and {*ugly*}, and the group {*cold*, *hot*, *warm*} into the subscales {*cold*} and {*hot*, *warm*}; it should label the subscales {*beautiful*} and {*hot*, *warm*} as positive and the subscales {*ugly*} and {*cold*} as negative, thus detecting the pairs of antonyms *beautiful–ugly*, *cold–hot*, and *cold–warm*; and, eventually, it should detect that within the subscale {*hot*, *warm*} *hot* is a semantically stronger term than *warm*. The first four of these tasks are addressed in this dissertation, while some potential approaches to accomplish the last task are discussed in Chapter 12 on future extensions.

These problems have been studied to a considerable extent in the linguistics literature, as they relate to significant issues in lexical semantics: the structure of the vocabulary, semantic relatedness between words, gradability, antonymy and synonymy, and scalar properties of words. However, the linguistics approach has been theoretical, largely informal, based on intuition, and unsuitable for computer implementation. Recently, and in parallel with the work reported in this dissertation, one of the problems mentioned above (grouping of semantically related words) has attracted researchers with a computational perspective. Descriptions of two systems for addressing the grouping problem, developed independently from each other and independently from the one described in this dissertation, have been recently published [Brown *et al.* 1992b; Pereira *et al.* 1993]. The work described here contains several components not found in these independently developed systems, as explained in the next paragraph. In addition, the present work addresses the three subsequent problems related to the analysis of the retrieved groups which have not been studied before from a computational knowledge acquisition perspective.[1]

To solve the four problems outlined above I chose to use exploratory data analysis and probabilistic methods that are both based on large text corpora, following and extending a research paradigm that currently receives a lot of attention in the natural language processing community. For each problem, appropriate linguistic properties and text-level indicators of these properties are identified, and methods for the empirical validation of the relevance of these properties and indicators are developed. Unlike other approaches, the learning algorithms described in the dissertation take advantage of both positive and negative examples. Furthermore, a unique aspect of this research is that it incorporates both statistical and shallow linguistic knowledge modules to maximize the performance of a completely automated system on these difficult tasks. For this purpose, I have devel-

[1]Except, for some of the problems, for superficial approaches based on simple pattern matching techniques; the latter can recover only a fraction of the information extractable from the corpus. See Section 1.8 and Chapter 10.

oped flexible architectures that allow the combination of local information from multiple knowledge sources and provide for the resolution of conflicts arising when the interactions of local decisions are taken into account. Statistical supervised learning methods such as log-linear regression are used to appropriately weigh the contributions of various modules, while unsupervised learning methods such as non-hierarchical clustering are employed when training examples are not expected to generalize to new problem instances.

For the evaluation of the system's performance, I have developed new methods that extend well-established quantitative measures from information retrieval. The extensions address how methods originally designed for the comparison of lists of "yes"–"no" decisions can be applied to the structured solutions produced by this system (for example, to partitions of sets of words). They also include a generalization of the measures to a reference set of more than one model, thus enabling the simultaneous use of several solutions independently prepared by humans for the evaluation of the system. I discuss the issue of interpreting the evaluation scores at length, and I provide upper and lower bounds for the system scores by applying the same evaluation methodology on the solutions produced by the humans and by comparing with random solutions through a Monte Carlo analysis. The evaluation and meta-evaluation methodology developed in the context of this thesis is not limited to the problems at hand or even to computational linguistics, but rather is applicable to many other problems, including problems from information retrieval.

Since the system uses several types of shallow linguistic knowledge for the processing of the corpus data and also uses additional modules that are justified on deeper linguistic analyses (e.g., the negative examples used during both the word grouping and semantic orientation identification stages), it provides a testbed for assessing how the various linguistic components contribute to the performance of the system in contrast to the basic statistical component. Using the evaluation methodology described above, quantitative measurements of performance can be collected and analyzed statistically. The results of comparative experiments of this type establish the usefulness of the linguistic modules, relative to the statistical component and to each other. Furthermore, by extending the comparative experiments along several values of a controlling parameter, trends in the performance can be established, and asymptotic projections can be made. I have performed such comparative experiments measuring performance with and without the linguistic knowledge modules as the size of the training data varied. These experiments establish that shallow linguistic processing of the input text, such as using a finite-state parser for the extraction of co-occurrence data or a morphology module, offers a clear performance advantage. Preliminary evidence also indicates that these benefits of the linguistic knowledge cannot be offset by an increase of the corpus size. These results, and the experimental methodology developed for obtaining them, can be generalized to a variety of other statistical language learning applications.

The reported work offers several contributions to computational linguistics, as it provides new methods for the problems of determining semantic classes of words and identifying gradability, scalarity, orientation, and opposition information. Detailed empirical analyses of the relevant linguistic properties and text-level indicators for each of these problems are also presented. Several distinct statistical approaches are combined in the system in a novel way, and solutions are offered for problems related to experimental validation, combining of evidence, using negative examples, and enforcing global consistency. Many of the results obtained are relevant to non-computational linguistics, as hypotheses formulated in theoretical linguistics are tested against real corpus data. The extensions to the evaluation methodology are relevant to many other computational linguistic problems and to information retrieval. Finally, the methodology developed for the comparative experiments and the obtained results can be transferred to several other learning problems.

In addition to the theoretical contributions, the system can be used to populate the semantic component of a lexicon with minimal human intervention, automatically adjusting to the sublanguage of the domain. This is a major application, the results of which will be useful to many natural language systems that use lexical semantic knowledge currently entered by hand. Even before a full lexicon is constructed, parts of the system's output can be directly used by other natural language systems, e.g., for selecting terms with the appropriate argumentative force during text generation [Elhadad 1991], for interpreting scalar implicature [Hirshberg 1985], and for detecting occurrences of multiple incompatible qualifiers in noun phrases [Moore 1993]. In Chapter 11 I list several potential applications of the knowledge extracted by the system to practical problems. I particularly focus on how these types of domain-specific lexical knowledge are relevant to applications from the medical domain, including the MAGIC system currently under development at Columbia University. MAGIC generates multimedia explanations of patient status after a bypass operation [Dalal *et al.* 1996a; Dalal *et al.* 1996b], and would benefit from the automatic acquisition of lexical relations in order to scale up its coverage of the domain.

1.2 Organization of the dissertation

In the remainder of this introductory chapter I present background material about lexical semantics (Section 1.3) and corpus-based methods as used in computational linguistics and lexicography (Section 1.4). In Section 1.5, I discuss the extent of automation that was set as the goal of the implemented system, and why I chose to avoid relying on any pre-compiled lexical sources such as dictionaries and thesauri. I then provide definitions of the problems presented above, discussing the meaning of the term *semantically related*, introducing the related concepts of *lexical fields* and *linguistic scales*, and listing a series of properties of linguistic scales and groups of semantically related words that either are useful in extracting such groups from text or imply applications of the knowledge of such groups in other tasks (Sections 1.6–1.7). Since the problems addressed in the thesis have received considerable attention in the linguistics literature, I present in Section 1.8 manual methods for them, and comment on their potential for automation. With this background properly set, I delineate in Section 1.9 the methodology I have developed for constructing semantic groups, and identifying, separating, and labeling the members of a scale, and briefly describe my approach for evaluating the system's performance. I close this chapter with a brief discussion of applications for the knowledge the system extracts, and of contributions to computational linguistics and related fields. I also include a discussion of the syntactic classes of words that the system is operating on, and explain how these classes were selected.

The main body of the dissertation is organized into three parts. The first part (Chapters 2–5) describes how words are placed into semantic groups automatically, while the second part (Chapters 6–9) covers the subsequent analysis of these groups to extract gradability, scalarity, semantic orientation, semantic markedness, and antonymy information. The final part (Chapters 10–13) presents supporting material such as related work, applications, potential future extensions, and contributions.

In Chapter 2 I present the linguistic properties and textual indicators I use in the first stage of the system, and the statistical approach and algorithms that utilize this information to measure the semantic similarity between words and form semantic word classes. The evaluation of the results of this stage is complicated because of the structure and interdependence of the classes; consequently, I present in Chapter 3 novel methods for an objective, quantitative evaluation of the system's performance. That chapter also includes a discussion of how the evaluation results should be interpreted in comparison to human and random performance in a statistical framework.

Results from several experiments on grouping adjectives and nouns, from both the financial and the medical domains, are shown in Chapter 4, along with the output of the evaluation algorithms on the produced clusterings. Using the evaluation methodology of Chapter 3, additional data-analytic experiments have been carried out in order to quantify the importance of linguistic knowledge for this statistical learning system. Chapter 5 identifies the different ways that linguistic knowledge has been introduced into the group-forming system, and discusses comparative experiments performed in this vein and the effect that each linguistic knowledge module has on the overall performance of the system. Several of the conclusions are generalizable to other tasks and apply to many of the published statistical natural language processing systems.

In the second part of the dissertation, I turn to the further analysis of the word groups. Chapter 6 presents simple yet effective methods for separating gradable and non-gradable words, which constitute a reasonable approximation for segregating scalar and non-scalar word groups. Chapter 7 describes methods for further subdividing the scalar word groups into positive and negative subscales, and offers results from applying these methods to a large collection of adjectives. These techniques can effectively separate positive from negative terms, but cannot help in deciding which is which. To solve this labeling task, I draw on the close correlation between semantic orientation and semantic markedness. I present in Chapter 8 a detailed corpus-based analysis of several linguistic predictors of semantic markedness and methods for combining their suggestions into a single decision. Then, I show how the predictions for single words can be aggregated across classes to achieve the labeling of the words with high precision. I conclude this part by successively applying the four stages of the system on the same collection of adjectives, thus showing how antonyms can be identified from free text.

The last four chapters are the standard fare to be found in all dissertations: After a discussion of related work from the linguistics and computational linguistics literature in Chapter 10, Chapter 11 identifies several practical applications for the knowledge extracted by this system, with an emphasis on how the output of the system can be used to augment the lexicon of the MAGIC system [Dalal *et al.* 1996a; Dalal *et al.* 1996b; McKeown *et al.* 1997]. Subsequently, Chapter 12 presents a critique of the limitations of the adopted approach and outlines several extensions that can be made in the future. In particular, I look at the problem of polysemous words which should ideally be placed in several groups, assess its extent from a lexicographic point of view, and discuss methods to alleviate it. I also discuss potential methods for ordering the words that have been placed in a given subscale according to their semantic strength, thus completing the analysis of the properties of scalar words. While these extensions are beyond the scope of the thesis, the multitude of directions for future work with impact in computational and traditional linguistics, lexicography, information retrieval, and psycholinguistics shows the richness and current under-developed status of the topic. Finally, Chapter 13 summarizes the theoretical and practical contributions of this work to computational linguistics, as well as the contributions to evaluation methodology that can be transferred outside the field of natural language processing.

Additional low-level details about the operation of the system are presented in a final appendix. The appendix describes the text corpora I use, how these corpora are preprocessed and automatically assigned part-of-speech information, and how the types of interlexical relationships that are used by the system are automatically retrieved from the text. It also presents the grammar that I developed for identifying these relationships and its implementation as a collection of nested finite-state automata, the modules that further filter and modify the retrieved information, and additional components outside the grammar that retrieve other types of information from the text (e.g., morphological relationships between words).

1.3 Lexical semantics

Linguistic knowledge can be characterized across two broad and orthogonal dimensions. The first dimension specifies the nature of the knowledge: whether it relates to the acoustic surface manifestation of linguistic entities (*phonological knowledge*), to structural constraints of the language below or above the word level (*morphological* and *syntactic knowledge*), to the meaning of words and phrases (*semantic knowledge*) or to constraints on the use of words and phrases in particular (possibly extra-linguistic) settings (*pragmatic knowledge*). The second dimension specifies the scope of applicability of each piece of knowledge: whether it pertains to the entire language (*grammatical* or *general knowledge*) or to a particular word (*lexical knowledge*). The general type of linguistic knowledge encompasses not only syntactic rules but also any other knowledge that is generally applicable across many linguistic items, e.g., antonymy in semantics or Grice's maxims in pragmatics.

The borders of the categories in the above classification are somewhat unclear for both dimensions. For example, grammatical knowledge is usually expressed as rules involving general classes of words. Thus the rule that the subject of a declarative sentence (usually a noun) precedes the verb in English is considered part of the syntactic grammar of English, while the fact that a particular word is a noun and can therefore function as a subject is part of the lexical knowledge for that particular word. However, most grammar rules have numerous exceptions; these exceptions can be viewed as specialized rules associated with some words which supersede the general rule. Taking this point of view further, all grammatical knowledge could be replaced with myriads of rules specialized for each lexical item. Nevertheless, for reasons of expressive parsimony, the distinction between general grammar rules and specific lexical entries has proved useful both for human users of the language and for computer systems. Yet, what exactly should be considered lexical knowledge is far from being satisfactorily solved, as manifested by the disagreement of any two dictionaries on what kinds of information to include.

Figure 1.1 shows sample entries from two leading contemporary dictionaries: the Collins COBUILD (*CO*llins *B*irmingham *U*niversity *I*nternational *L*anguage *D*atabase) English Language Dictionary [Sinclair 1987b] and the *L*ongman *D*ictionary *o*f *C*ontemporary *E*nglish (LDOCE) [Procter 1978]. Information from these dictionaries has been used in computational systems, especially from the LDOCE which is also available in machine-readable form. However, the information given by each of the dictionaries is not equivalent. For example, the LDOCE provides at several places more detailed lexical syntactic knowledge (note the *[U9]* code in the entry for *clemency* which refers to a subdivision of the uncountable nouns), while the COBUILD dictionary provides more semantic information. Examples of such lexical semantic information include the marking of *collaborate* with the RECIP feature to indicate a reciprocal action; the distinction between gradable and non-gradable adjectives (the former are marked with the feature QUALIT as in the example of *colossal*); and the inclusion of synonyms, antonyms, and superordinates (marked with the symbols =, $\neq$, and $\Uparrow$ respectively in Figure 1.1). Nevertheless, the dictionaries are far from complete for many types of lexical semantic knowledge, and some of the information given is controversial. For example, is *huge* a superordinate of *colossal* as COBUILD claims, or is

COBUILD		LDOCE
clemency /kl**ɛ**mənsi¹/ is kind treatment that a person receives, especially from someone who has authority to punish them; a formal word. EG ... *appeals for clemency by the lawyers of the condemned men.*	N UNCOUNT = leniency ≠ harshness	**clemency** /ˈklemənsi/ *n* **1** [U] mercy, esp. when used to make a punishment less severe **2** [U9] (esp. of the weather) gentleness; MILDness
collaborate /kəl**æ**bəreɪt/, **collaborates, collaborating, collaborated.** **1** If two or more people **collaborate**, they work together to produce a piece of work, especially a book or some research. EG *Anthony and I are collaborating on a paper for the conference... The film was directed by Carl Jones, who collaborated with Rudy de Luca in writing it.*	V OR V + A *(with)*:RECIP, IF +PREP THEN *in/on* = team up	**collaborate** /kəˈlæbəreɪt/ *v* [I∅] (*with* and/or *on*) **1** to work together or with someone else: *Quirk and Greenbaum collaborated (on the new grammar)* **2** [(*with*)] *derog* to help an enemy country which has entered and taken control of one's own: *Anyone who collaborates should be shot* — **-rator** *n*
2 If someone **collaborates** with an enemy army or government which has taken control of their country by force, they give help to the enemy, for example by giving information. EG *I believe he collaborated with the Nazis... We are not willing to collaborate.*	V OR V + A *(with)*:RECIP ⇑ comply = co-operate	
colossal /kəl**ɒ**sə⁰l/. Something that is **colossal** is very large. EG ... *colossal sums of money... The problems would be colossal.* ◇ **colossally**. EG *She's improved colossally over the period of the course...* ... *colossally powerful teaching computers.*	ADJ QUALIT ⇑ huge ◇ ADV = immensely	**colossal** /kəˈlɒsəl ‖ kəˈlɑ-/ *adj* very large in size or quantity: *a colossal building* | *a colossal rate of interest* **colossally** /kəˈlɒsəli ‖ kəˈlɑ-/ *adv* **1** *infml* very: *That was colossally interesting.* | *They had a colossally big house* **2** in a COLOSSAL manner: *a colossally planned palace*

Figure 1.1: Sample entries from the Collins COBUILD dictionary and the Longman Dictionary of Contemporary English (LDOCE).

colossal a stronger (but not more specialized) term?[2] As for pragmatic lexical knowledge, note that COBUILD marks *clemency* as a formal word, while LDOCE doesn't; similarly, LDOCE marks *colossally* as informal while COBUILD doesn't.

[2]My intuition supports the second alternative. I performed an informal survey here at Columbia University and all informants found this second analysis preferable, with a varying degree of preference. In any case, it seems that the relationship between *colossal* and *huge* is very different from a clear adjectival hyponymy relationship, e.g., the relationship between *scarlet* and *red*.

Attempts to formally model linguistic knowledge have been far more successful with the lower levels (phonology, morphology, syntax) than with semantics or pragmatics. Also, grammatical and general knowledge has been explored and codified much more than lexical knowledge. This is largely due to the higher number of rules of lexical knowledge, since these rules are by definition specific to each word in the language. However, access to all kinds of linguistic knowledge is needed for fluent use of the language. Therefore, methods are needed to enable the acquisition of lexical knowledge, and in particular for the harder aspects of it, the semantic and pragmatic components.

1.4 Corpus-based methods

In recent years, corpus-based methods have been steadily gaining ground, both among lexicographers who build dictionaries for use by humans and among computational linguists. Lexicographers use a large collection of texts as the source of observations about particular words of interest, thus extracting and formalizing lexical knowledge from the patterns of use of the words. Words and their senses, as well as word collocations, can be collected in a more accurate and exhaustive manner from a corpus than by introspection alone [Kjellmer 1984; Edwards 1992]. The corpus also provides real examples of the words in context, which can be included in the dictionary instead of artificial examples written by the dictionary creators. The COBUILD dictionary, mentioned in the previous section, is a recent example of a dictionary that relies heavily on corpus data. It uses the Birmingham Collection of English Texts, a collection of 20 million words from fiction and nonfiction texts and newspaper articles and of 20 million words of transcribed speech. Sinclair [1987b] argues that access to the corpus lends new authority to the examples listed in the dictionary entries, since they are taken from actual language use, and allows an informed decision of which words to include in the dictionary, according to their currency and frequency of use. To this, I add another obvious advantage, the increased accessibility of examples and word or word pattern counts from the machine-readable corpus. Atkins, Clear, and Ostler [1992], Sinclair and Kirby [1990], and the articles collected in [Sinclair 1987a] and [Gellerstam 1988] discuss in more detail how text corpora are employed in lexicography and the benefits of the corpus-based approach.

Grammarians that prepare grammars for use by humans have also used corpora for similar purposes, mostly for the empirical verification of a particular construct in the language, or for the comparison of the frequency of use of some grammatical constraints and rules according to dialect and regional variety (e.g., British versus American English). For example, Quirk *et al.* have used four text corpora[3] as reference sources during the preparation of their grammar [Quirk *et al.* 1985]. This *exploratory* use of corpora as a reference of examples of language use by humans has also been successful in linguistics and computational linguistics. Corpora have been used for the confirmation or rejection of hypotheses based on theoretical linguistics (e.g., [Kučera 1982; Justeson and Katz 1991; Passonneau and Litman 1993; Hatzivassiloglou and McKeown 1995]) and for investigating surface manifestations of a particular linguistic phenomenon (e.g., [Kukich 1983; Bourbeau *et al.* 1990; Carcagno and Iordanskaja 1993; Iordanskaja *et al.* 1994; Robin and McKeown 1996]). For further general

[3]The Brown University corpus, the Lancaster-Oslo/Bergen (LOB) corpus, the Survey of English Use (SEU) corpus, and the London-Lund corpus of spoken material.

discussion of the role of text corpora in linguistics research, see [Leech 1991], [Chafe 1992], [Fillmore 1992], [Halliday 1992], and [Leech 1992].

In the computational linguistics community, a different kind of corpus-based work has also become established, namely the use of corpora as the source of numerous training examples for training *probabilistic* learning systems. Initial attempts to model language processing abilities on a computer during the seventies and early eighties focused on grammatical problems, mostly on the implementation of syntactic formalisms for automatic parsing although considerable work was also done in modeling the semantic and pragmatic aspects of language. Characteristic of most of the work in that period was the adoption of a *symbolic* approach, where knowledge is collected by hand and explicitly represented and manipulated by the system in symbolic form. While these systems met with reasonable success and were able to provide detailed analyses or solutions for isolated problems and limited domains, they failed to scale up when the vocabulary was considerably expanded, or when the domain was not a highly controlled, confined one. The reason for this failure was in a large part the lack of specific lexical knowledge. The lexical knowledge needed for a small prototype of the system was entered by hand by the system builders, who used dictionaries or their own intuition as the source of this knowledge. However, this method of acquiring lexical knowledge can cover only a small number of lexical items, resulting in the failure of the system when deployed in a different or unrestricted domain.

A problem that perhaps best exemplifies the importance of lexical knowledge for an applied computational system is machine translation. Highly specialized systems, typically developed by an international corporation for its own internal translation needs, perform at very satisfactory levels [Mitamura and Nyberg 1995; Hayes *et al.* 1996; van der Eijk *et al.* 1996]. This is achieved by highly restricting the domain of the source texts and controlling the grammar and especially the vocabulary used in it; in some cases, authors of texts in the source language have to conform to specific prescriptive rules, which are checked and automatically enforced as the text is written [Wojcik *et al.* 1990; Adriaens 1992; Hoard *et al.* 1992; Ravin 1993; Richardson and Braden-Harder 1993; Clémencin 1996; Douglas and Hurst 1996] On the other hand, general-purpose knowledge-based machine translation systems fail to perform adequately when run on unrestricted text such as newspaper articles. This has led to a heightened awareness of the importance of (monolingual and bilingual) lexical knowledge and of the need for its automatic acquisition.

Probabilistic natural language systems address this deficiency, making the acquisition of new knowledge, and especially lexical knowledge, much more practical. Such systems incorporate a particular statistical model for the task at hand which has a number of numeric parameters left unspecified by the designer. These parameters can be only a few dozen for the entire language as in the case of systems attempting the automatic acquisition of *grammatical* knowledge such as probabilistic parsers, or several *for each word*, when the knowledge extracted is lexical in nature. The systems use the actual examples observed in collections of texts to infer (estimate) the values of these parameters. Such texts have been made widely available recently (and especially during the past decade), starting with the Brown corpus [Kučera and Francis 1967; Francis and Kučera 1982] of approximately one million words in the sixties and including recent corpora of up to 100 million words [Quirk 1992]; see [Edwards 1992] for a discussion of several widely available corpora. Restricted-access corpora, typically held by private research laboratories, are even larger, e.g., the more than 250 million word Bellcore Lexical Research Corpora [Walker 1987] or the IBM collection of 583 million words [Brown *et al.* 1992a]. With corpora of large size, statistical methods become applicable. For example, a statistical system might count the number of times a particular word has a particular part of speech in a suitably marked training corpus, and separate these counts according to the previous words. Differences between the expected probability of seeing a particular part of speech for a word given only that word, and the expected probability of seeing the same part of speech for the same word given the word and its predecessor in the text can be noted and used to decide the correct part of speech for the word in a new, unmarked text. In this way, the system can decide that *bolts* is a plural noun when it follows *the*, but a verb when it follows *he*. Systems using more refined versions of the same concept [Church 1988; de Rose 1988; de Marcken 1990; Merialdo 1991; Meteer *et al.* 1991; Kupiec 1992; Cutting *et al.* 1992; Charniak *et al.* 1993; Merialdo 1994; Tzoukermann and Radev 1996] currently address the problem of part-of-speech disambiguation with an accuracy of 94-99%, depending on the text and set of tags used. Furthermore, the methodology is a quite general one. Statistical corpus-based systems have been developed to address a wide variety of problems, including grammar induction [Garside and Leech 1985; Magerman and Marcus 1990; Sharman *et al.* 1990; Magerman and Marcus 1991; Briscoe and Waegner 1992; Carroll and Charniak 1992; Jones and Eisner 1992; Pereira and Schabes 1992; Black *et al.* 1993; Magerman 1995], sense disambiguation [Brown *et al.* 1991; Dagan *et al.* 1991; Gale *et al.* 1992b; Liddy and Paik 1992; Yarowsky 1993; Dagan and Itai 1994], machine translation [Brown *et al.* 1990; Brown *et al.* 1993; Berger *et al.* 1994; Yamron *et al.* 1994], term identification and translation

[Kupiec 1993; van der Eijk 1993; Dagan and Church 1994; Daille 1994; Fung 1995; Justeson and Katz 1995b; Zhou and Dapkus 1995; Evans and Zhai 1996], extraction and translation of collocations [Smadja 1993; Smadja *et al.* 1996], language and topic identification [Dunning 1994; Canvar and Trenkle 1994], text restoration [Yarowsky 1994], and lexical choice in text generation [Knight and Hatzivassiloglou 1995].

Naturally, the output of such systems rarely is 100% correct; it either provides a reasonably approximate solution with a low error rate, or a basis for further post-editing by humans. The disadvantage of imperfect output is offset by the advantages of robustness and generality offered by probabilistic methods. However, perhaps following the path of least resistance, most work in this new sub-area of computational linguistics has been done once again in the area of syntax. Yet, automating the acquisition of semantic lexical knowledge from text corpora can be most beneficial for the following reasons:

1. Semantic lexical knowledge is missing from current dictionaries, whether machine-readable or not. While many syntactic features of words (such as part-of-speech, number, and form of complement) are present in dictionaries, semantic features such as synonyms and antonyms are not generally available (see Figure 1.1). Even when dictionary designers try to emphasize the semantic component of lexical knowledge, the results are not fully satisfactory. For example, I note three deficiencies in COBUILD's coverage of semantic lexical knowledge: First, only three types of semantic links (synonyms, antonyms, superordinates) are employed, so relationships such as "word X is a stronger term for the same concept expressed by word Y" are not in the dictionary. Second, usually at most one link of each type is given. Third, only part of the entries

which would qualify for such links actually have links listed.

2. Semantic lexical knowledge is most sensitive to domain changes. Syntactic constraints on the use of a word are relatively stable across different senses of a word, or across different uses of the word in disparate domains. Consequently, in most dictionaries syntactic features of a word are listed just after the citation form and apply to all senses. However, semantic features tend to change as the word is used in a different way in different domains. For example, the system described here identified "*preferred*" as the word most closely semantically related to "*common*"; this association may seem peculiar at first glance, but is indeed a correct one for the domain of stock market reports and financial information from which the training material was collected.

3. Semantic lexical knowledge is also harder to collect for a human lexicographer, as the linguistic theory and tests needed for such identification are less developed than the corresponding ones for syntactic knowledge; sometimes there is no clear test, or there is considerable disagreement among linguists on the merit of proposed tests and theories. Part of this problem stems from the fact that there is no broadly accepted general theory of semantics, and this problem certainly contributes to the conspicuous absence of semantic lexical features from most dictionaries.

4. The lack of clearly defined tests or formal definitions also means that the lexicographers frequently have to employ their intuition. This in turn may lead to the introduction of bias in the compiled lexicon entries, according to the particular linguistic theories that the lexicographer adheres to. It is also a potential source of incomplete

entries, due to human oversight, and of inconsistencies, especially in the case of works with many authors. In contrast, an automatic system can be more thorough and impartial, since it bases its decisions on actual examples drawn from the corpus. Even if bias is introduced because of the linguistic model at the core of the system, at least this bias is known in advance and can be taken into account when the results are interpreted.

1.5 Extent of automation and accessible knowledge resources

A major difference of the current work from other probabilistic approaches is that the system described here uses linguistic, and in particular syntactic, knowledge to extract appropriate learning examples from the corpus. The design goal I set was to produce a probabilistic, corpus-based system that operates on a given set of words without human assistance, partitioning this set into groups of semantically related words, and subsequently identifying which of these groups correspond to linguistic scales, separating positive and negative terms in each group, and assigning a positive or negative orientation label to each word. The techniques employed by the system had to be general across syntactic classes, although because of time limitations, their actual implementation was not planned for and was not extended to all open-class words. While the incorporation of shallow linguistic knowledge results in increased performance (see Chapter 5 for an empirical validation of this claim), it requires different extraction rules for each syntactic class, at a cost of significant additional implementation effort. This is not the case for other systems that operate on words or word sequences without explicit regard for syntactic constraints.

The overall goal was a system that operates without human intervention or supervision and actually produces useful results. The output of the system is not expected to be fully correct, something which is invariably true for all probabilistic approaches to learning in general. However, the output is consistently good according to formal evaluation methods that will be presented in later sections (Chapter 3). At this stage, after the system has finished processing the input words, a human lexicographer may edit the output to remove mistakes made by the system.

I have noted above that the system is fully automatic, without need to be explicitly informed by human intervention. A related point is the kinds of linguistic knowledge sources that the system uses, i.e., shallow syntactic and morphological knowledge versus lexical semantic knowledge. I have made the design decision to avoid any access to explicit lexical knowledge sources such as dictionaries or thesauri during the system's operation. Some syntactic and morphological knowledge is encoded in the form of rules in the system; other than that, the system has access only to a large collection of texts where the words that are to be grouped appear. This collection of texts constitutes the learning material, from which distributional properties of the words of interest, and eventually measures of semantic relatedness, gradability, or semantic orientation are extracted.

There are several reasons for avoiding access to lexical knowledge bases such as machine readable dictionaries. First, much of the information that would be useful for the system's tasks is simply not there. Second, even if such information is present, it may be sporadic, inconsistent, and biased. Third, using any precompiled source of lexical knowledge limits the applicability of the system to the domain from which that source was compiled, since the system's output should be sensitive to domain changes [Hirshberg 1985]. Finally, not using any kind of lexical knowledge source provides an opportunity to see how successful a "knowledge-poor" system is in extracting high-level specialized semantic knowledge using just a text corpus and shallow, non-lexical knowledge. In this way, the behavior of the system can help in determining how much of lexical knowledge is extractable from distributional properties alone, and thus influence future research in lexical acquisition and even language modeling.

1.6 Further definitions: Semantic relatedness, lexical fields, and linguistic scales

In order to provide a formal definition of the linguistic objects and relationships that I identified in the thesis, I would like to have access to and make use of a sufficiently developed and broadly acceptable theory of formal linguistic semantics, at both the compositional and lexical levels. Indeed, several attempts to provide such a general formal theory of nature language semantics have been made by linguists, philosophers, or logicians [Carnap 1947; Bar-Hillel and Carnap 1952; Davidson 1967; Hintikka 1969; Montague 1970; Lewis

1972; Lewis 1973; Montague 1973; Kamp 1981; Barwise and Perry 1983; Keenan and Faltz 1985]. However, despite the common reliance of most of these proposals on Tarski's [1935] theory of truth, it seems that little broad consensus has developed on what the proper approach is. In any case, most of the proposed formalisms have been developed in the form of a prototype for a small set of linguistic phenomena or lexical items, and have not been extended to and empirically tested on a significantly large part of the semantics of a natural language.[4] Kempson [1977, p. 159] notes

> "... [T]here remains a pervasive ignorance about so many aspects of the semantic properties of sentences, that to give a complete semantic account of any sentence is not unfortunately a realistic task at the present time."

and later [*ibid.*, p. 184]

> "... [T]here is almost no aspect of semantics over which there is any degree of certainty: there is considerable disagreement over the basic assumptions of a semantic theory, the nature of semantic representations is contentious, and there is no agreement about the relation of semantics either to logic or to other parts of a grammar."

While progress has certainly been made in the twenty years since Kempson's statements, her words continue to reflect to a large extent the situation on the formalization of natural

[4] A notable exception is Levin's recent work on verb semantics [Levin 1993], which has been received so well in part exactly because of its extensive coverage of English verbs.

language semantics.

A similar situation also applies to lexical semantics, for which several formal theories have also been proposed [Dowty 1979; Mel'čuk 1981; Jackendoff 1985; Mel'čuk and Pertsov 1987; Mel'čuk and Polguère 1987; Jackendoff 1990; Pustejovsky 1995; Pustejovsky and Boguraev 1996]. As in compositional natural language semantics, in most cases the proposed formalisms have not been carried on to extensive subsets of the vocabulary nor have they been tested empirically on a large scale. An exception to this limitation of formal models of the lexicon to prototype systems can be found in the natural language generation community, where the work of Mel'čuk and his colleagues on Meaning Text Theory [Mel'čuk and Pertsov 1987; Mel'čuk and Polguère 1987] has found a number of followers who implemented extensive computational systems based on his proposal [Boyer and Lapalme 1985; Bourbeau *et al.* 1990; Carcagno and Iordanskaja 1993; Iordanskaja *et al.* 1994]. Even in text generation, the problems of extending a general formal theory to the coverage required for unrestricted text have been noted, and researchers have argued for the need of specialized models finely tuned to each particular domain rather than an all-encompassing general theory [Kittredge and Lehrberger 1982; Kittredge and Grishman 1986; Danlos 1986].

Within this context, it is inevitable that the term "semantically related" is not formally defined. Linguists have a long tradition of work associated with "semantic relatedness" and "groups of semantically related words", starting with Trier [1934]. Trier proposed that the vocabulary of a natural language is organized in groups that he called *lexical fields* ("Wortfeld"), which represent *conceptual fields* ("Sinnfeld"), the latter being sets of closely related concepts.[5] Trier's lexical fields closely correspond to the "groups of semantically related words" (Section 1.1) that the present system aims to find. The work of Trier focused more on the diachronic and cross-linguistic aspect of lexical semantics, but a considerable number of linguists followed and extended his approach; see [Ullmann 1957] and [Lehrer 1974] for comprehensive treatments of the theory of lexical and semantic fields. Unfortunately, the theory is mostly based on intuitive judgements; as Lyons [1977, p. 277] remarks

> "What is lacking so far ... is a more explicit formulation of the criteria which define a lexical field ... The majority of lexical fields are not so neatly structured or as clearly separated as Trier originally suggested."

Given the lack of a formal definition of semantic groups, I will use an informal, intuitive definition for semantic relatedness and, treating semantic relatedness as an equivalence relation in the vocabulary, I will then define the semantic groups as the equivalence classes induced by this relation. What I mean by "semantically related" is that words belonging to the same group should consistently express a closeness in their meaning, for example by being synonyms, antonyms, complementary terms, special cases of the same superordinate concept (co-hyponyms), or terms describing the same property. Informal criteria such as the above, which can be further modified when the particular syntactic class from which the members of the groups come is known, have been frequently used in linguistics (computational or not). As in many cases where an informal definition is used, results based on this definition are open to criticism such as the remarks by Lyons above and there is

[5] The origins of this organization of words and of their associated concepts can be traced back to [Herder 1877] and [von Humboldt 1836]; cf. [Lyons 1977, p. 250].

disagreement on what exactly is the correct answer, given the same input. Correspondingly, there is difficulty in interpreting the results; I discuss these problems and the steps taken to alleviate them in Chapter 3 on evaluation (Section 3.4).

Linguistic scales constitute a particular category of such groups of semantically related words. Levinson [1983, p. 133] defines a linguistic scale as "a set of linguistic alternates, or contrastive expressions of the same grammatical category, which can be arranged in a linear order by degree of informativeness or semantic strength." For example, *lukewarm*, *warm*, and *hot* fall along a single adjectival scale since they indicate a variation in the intensity of temperature of the modified noun (at least when used in their non-metaphorical senses; metaphorical usage of scalar words normally also follows the order of the scale by analogy). Scales are not limited to adjectives as in the above example; for example, {*may*, *should*, *must*} and {*sometimes*, *often*, *always*} [Horn 1972] are linguistic scales consisting of auxiliary verbs expressing obligation and of adverbs expressing frequency respectively.

Within scales and sometimes even non-scalar groups of semantically related terms there is frequently a qualitative distinction between positive and negative terms. Positive and negative sides of the group can be defined by comparison with a neutral element of the group if such an element exists, by reference to an extra-linguistic property that the group models, or by evaluative criteria that relate to how the term is perceived in the language [Lyons 1977; Battistella 1990]. For example, if the group corresponds to the physical property of temperature, terms such as *warm* and *hot* belong to the positive side (according to both physical and evaluative criteria), while *frigid* and *cold* belong to the negative side. Similarly, in the opposition between *beautiful* and *ugly*, *beautiful* clearly has a positive evaluative connotation and *ugly* a negative one.[6]

Furthermore, there is a quantitative distinction among the terms belonging to the same (positive or negative) side of a linguistic scale, according to how strongly they convey the common characteristic the group is associated with (or the negation of this characteristic). Terms belonging to such subsets of scales follow a total order[7], and knowledge of this order is useful for natural language applications. In addition, other properties of scales relate to this quantitative relationship, for example the way the endpoints of a scale should be interpreted: A scale may be open-ended, with the extreme elements denoting real world entities at some arbitrarily high or low values of the characteristic (for example, *frigid* or *hot* in the temperature example above), or closed, with the extreme elements denoting entities at absolute endpoints of the characteristic (for example, *dead–alive*).

1.7 Properties of linguistic scales and groups of semantically related words

I now turn to several interesting properties of linguistic scales and semantic groups. These properties can be exploited by a system that has knowledge of these groups or scales, and thus justify the usefulness of such knowledge for applied tasks. Furthermore, some of the properties can be used to justify linguistic indicators that are used by the system, and thus

[6] As with most linguistic scales [Hubler 1983], there is no explicit neutral element in these examples.

[7] And therefore a partial order exists in the entire group.

their presentation offers the necessary background material for the discussion of linguistic properties and indicators in Chapters 2 and 6–8.

One property shared by all members of a scale or of a group of related words as defined in Section 1.6 above, is that they all denote related entities or describe a common characteristic. In addition to this obvious property, there are several other properties that stem from the definitions and from general observations in semantics and pragmatics. For the formal description of these properties, I will use indexed Greek letters for the elements of such groups, $\prec$ for the semantic strength relation ($\alpha \prec \beta$ if and only if β is a stronger or more informative term than α), and upper-case Latin letters for the groups, possibly indexed to denote a particular side (positive or negative) of the group. The function $A()$ will stand for a sentential frame that takes as argument a member of a semantic group or scale and produces a sentence.

1.7.1 Well-formedness of sentences under substitution

Suppose that we are given a sentence where a word from a group of semantically related words appears; then we can substitute this word with any other element of the group and still produce a well-formed sentence. In other words, for any sentential frame $A()$ and semantic group S, if $A(\alpha)$ is a well-formed sentence for some $\alpha \in S$, then $A(\beta)$ is a well-formed sentence for all $\beta \in S$. For example, in the sentence

The water is warm

we can substitute *warm* with any other adjective from the same group (scale), e.g., *hot*, *cold*, etc. Note that the most interesting part of this well-formedness property is the semantic coherence of the produced sentence, as syntactic grammaticality is guaranteed from the fact that a word is replaced by a word from the same grammatical category.

Of course, many other adjectives outside the semantic group that contains *warm* can take the place of *warm* in the above example and still produce a semantically valid sentence. For example, we can say *The water is clear/murky/pure/foul/poisonous/thirst-quenching/...* The point is that all members of the semantic group that contains *warm* can replace *warm*, while not all the adjectives outside the group can do so; cf. **The water is tall/mean/open/round/...* where the resulting sentence is syntactically valid but semantically anomalous.

It should also be noted that this property holds most of the time, but can be violated when the group is dependent on some other constituent of the sentence. For example, adjectival groups are sometimes dependent on the modified noun, and so the phrase *hot ice-cream* is semantically anomalous, while *cold ice-cream* is an acceptable way to indicate that the ice-cream remains in its normal (non-melted) state. This violation of compositionality and its effects on methods for identifying and representing groups and scales are not addressed in this work, and will be left for future extensions to the thesis.

1.7.2 Logical incompatibility

Terms from opposite sides of a semantic group produce incompatible sentences (in the logical sense) when replacing one another in a sentential frame. Formally, assuming that $A(\alpha)$ is a well-formed sentence, $A(\alpha) \wedge \alpha \in S_+ \Rightarrow \forall \beta \in S_-, \neg A(\beta)$, and $A(\alpha) \wedge \alpha \in S_- \Rightarrow \forall \beta \in S_+, \neg A(\beta)$. This property is a direct consequence of the definition of the two sides in semantic groups and scales. For example, the truth of the sentence

The water is warm

implies that the sentence

The water is cold

is false. Similarly, asserting the truth of the second of these sentences implies that the first sentence is false.

1.7.3 Logical entailment

With scales expressing quantifiable properties, asserting the truth of a sentence involving a particular element α of the scale implies the truth of the same sentence with α replaced by any element of the same side of the scale that is semantically less strong. For this property to hold, it should be possible to interpret the semantically less strong term β as "*at least* β"; not all terms and not all contexts allow this interpretation. Formally, $\forall \beta \in S, \alpha \in S \wedge A(\alpha) \wedge \beta \prec \alpha \Rightarrow A(\beta)$. For example, if the statement

The water is hot

is true, the statement

The water is (at least) warm

is also true. In a similar vein, but without requiring qualification by *at least*, the statement

The food is excellent

implies that

The food is good

Note, however, that sentences $A(\beta)$ of the above type would not be uttered despite their true value [Levinson 1983], according to the property discussed next.

1.7.4 Scalar implicature

If a sentence involving a scalar term α is uttered, then the same sentence with α replaced by any stronger term from the same scale is false. Formally, $\forall \beta \in S, \alpha \in S \wedge uttered[A(\alpha)] \wedge \alpha \prec \beta \Rightarrow \neg A(\beta)$. This is a consequence of Grice's maxim of Quantity [Grice 1975]:

"(i) Make your contribution as informative as is required (for the current purposes of the exchange).

(ii) Do not make your contribution more informative than is required."

and in particular of submaxim (i) listed above. This kind of conversational implicature, which is known as *scalar implicature* [Gazdar 1979; Levinson 1983; Hirshberg 1985], is justified as follows: If the substitution of the stronger term β for α produces a sentence $A(\beta)$ that is still true, then using the weaker term α results in a sentence $A(\alpha)$ that is not as informative as $A(\beta)$ and thus is potentially misleading.[8] For example, stating that

New York City is a town

is not acceptable if part of the intent of the communication is to inform the hearer about the size of New York City without being misleading. While New York City is certainly a town, it would be more informative (and expected if the speaker is assumed to know the size of New York City) to say

New York City is a big city / a major metropolis

The same principle is exploited in reverse during the application of scalar implicature; if the speaker says

The water is warm

then it follows that, at least to the speaker's knowledge,

The water is <u>not</u> hot

because if it were in fact hot, the speaker would have been expected to say so in order to convey the maximum amount of information.

---

[8] On the basis of Grice's [1975; 1978] theory of conversational implicature, we can in fact make the stronger inference that by uttering $A(\alpha)$ under the above circumstances, the speaker knows (or believes to know) that $A(\alpha)$ is true and that $A(\beta)$ is false for all $\beta \in S$ with $\alpha \prec \beta$; using the K operator from Hintikka's [1962] epistemic logic, $uttered[A(\alpha)] \Rightarrow \mathrm{K}[A(\alpha)]$ and $\forall \beta \in S, \alpha \in S \wedge uttered[A(\alpha)] \wedge \alpha \prec \beta \Rightarrow \mathrm{K}[\neg A(\beta)]$.

1.7.5 Mutual exclusion of terms from the same group[9]

It is not possible for two terms in the same scale or semantic group to occur next to each other in a sentence, participating in a paratactic relation where both modify the same constituent or fulfill the same syntactic role for another constituent, or in an hypotactic relation where one of them modifies or is a syntactic argument of the other.[10]

To establish that two terms from the same semantic group cannot participate in a paratactic relation we reason as follows (a rather straightforward application of the *reductio ad absurdum* method): Suppose that this happened for two particular terms. The two terms would have to satisfy one of the following two cases, which exhaust all possibilities:

- Either the two terms belong to different sub-scales of the scale, e.g., *cold hot water*, in which case we have a logical (semantic) contradiction. A similar situation arises for members of non-scalar groups, where the two terms represent different values of the common characteristic of the group, e.g., *red green dress* which would imply that the dress in question is simultaneously red and green. Compare with *reddish green dress* (the color of the dress is a particular type of *green* which shares some minor characteristics of the color *red*), and *red-green dress* (the dress is colored partly red

[9]While the properties in Sections 1.7.1–1.7.4 above are either obvious or well expounded in the linguistics literature, the material in this subsection is an original contribution (to the best of my knowledge), developed for this thesis.

[10]The exact surface form of such relations needs to be individually specified for each grammatical class of words participating in semantic groups. For example, the surface realization of this rule for adjectives prohibits the appearance of two members of the same group together with no constituent separating them, but allows their presence in the same noun phrase when coordinated or separated by a comma (apposition or asyndetic coordination).

and partly green).

- Or the terms belong in the same sub-scale of the scale, e.g., *warm hot water*. However, in this case we have a violation of the Gricean maxim of Manner [Grice 1975]:

  "Be perspicuous, and specifically

  (i) Avoid obscurity.

  (ii) Avoid ambiguity.

  (iii) Be brief.

  (iv) Be orderly."

  More specifically, the submaxim (iii) listed above is violated since the same information is conveyed by the strongest of the two terms and the other one is redundant.[11]

In essence, the two terms "compete for the same part of the picture" [Ayer 1952], and thus they cannot be used simultaneously. Similarly, the incompatibility of terms that appear in hypotactic relationships of the type described at the start of this subsection follows from the fact that both terms describe the same property and thus one of them cannot modify or specify the meaning of the other; any attempt to do so leads again to a contradiction or adds no new information.

[11]An exception to this rule may be observed for stylistic reasons, but very rarely.

1.8 Linguistic tests for identifying scales

Before discussing methods for the automatic extraction of semantic groups, it is prudent to examine methods proposed for the particular case of linguistic scales in the linguistics literature. Some specific linguistic tests can be applied to pairs of words of the same grammatical category to determine if they belong to the same scale, and if so, if they belong to the same or different sub-scales and their relative semantic strength. These tests consist of sentential frames where placeholders are replaced by the two words being tested, and the acceptability or not of the formed sentence determines if the two words satisfy the relation that is investigated. A classic example is the test proposed by Horn [1969], which determines if two words α_1 and α_2 belong to the same scale and which of them is the stronger:[12]

It is α_1, *it is even* α_2

If this sentence is acceptable, then α_1 and α_2 belong to the same sub-scale of the same scale and $\alpha_1 \prec \alpha_2$ (i.e., α_1 is semantically weaker than α_2)[13]; also the same sentence with α_1 and α_2 interchanged is not acceptable, e.g.,

It is warm, it is even hot

**It is hot, it is even warm*

---

[12]The forms of the tests listed are syntactically valid only for word classes that can appear as the complement of the copula *is*; however, the tests can be easily adapted to other categories of scalar words, such as verbs.

[13]Horn considered scales with no positive and negative sub-scales but with a total order among their elements.

This test can be extended to work with the more generalized definition of scales used in this thesis (i.e., with scales that have positive and negative sub-scales) by using the forms

It is not α_1, *it is even* α_2

It is not α_2, *it is even* α_1

in addition to the previous one [Smadja 1991]. If any of the sentences in the second group is acceptable, we can conclude that α_1 and α_2 are in the same (generalized) scale. For example, both sentences

It is not cold, it is even hot

It is not hot, it is even cold

are acceptable. This second version of Horn's test does not allow us to determine if the two words belong to the same side of the scale or their relative semantic strength. However, the first form of the test can be subsequently used (twice, with α_1 and α_2 interchanged the second time) to check if the two words are in the same sub-scale and if so, to determine their relative ordering.

This test is also complete, i.e., if the words are in the same sub-scale and $\alpha_1 \prec \alpha_2$ then the sentence

It is α_1, *it is even* α_2

will be acceptable and the sentence

It is α_2, *it is even* α_1

will not. A similar statement can be made for the second, generalized form of Horn's test.

Following the same line of reasoning, we can develop other tests which can be more or less restrictive (and therefore informative) than the one proposed by Horn, using a variety of intensifiers and contrast words. Also, Bolinger [1977] has proposed tests for determining if a word is near the neutral element of the scale or near one of its extremes. Therefore, linguistics provide an arsenal of methods for determining if two words are in the same group or scale, if they are in the same or in different sides of it, and the relative strengths of the two words, compared to one another or to the neutral point of the scale. However, all these methods assume that a human, knowledgeable of the valid sentences in the language, will be available to form and evaluate these test sentences for all the words in question. When we want to perform this task automatically, it seems that we have to restrict ourselves to looking for a test pattern to appear in a large corpus of valid sentences and then draw conclusions about the adjectives that participate in the occurrences of that pattern.

Hearst [1992] has developed a system that extracts hyponymy relationships from a corpus by directly searching for occurrences of a sequence of noun phrases (NP_1, NP_2, $\ldots$) followed by the words *or other* and a final noun phrase NP_0. The structure of the pattern implies that all noun phrases before the words *or other* are hyponyms of NP_0. However, I believe that direct search of the corpus for patterns corresponding to linguistic tests is unlikely to produce a complete enough identification of the groups and scales in a corpus, for any but the most frequent words. At best, only a small subset of the information present in the corpus can be extracted with a direct method. Furthermore, an approach based on direct pattern matching can be successful only if the pattern matches only those words that satisfy the relationship we are looking for; i.e., the number of recovered spurious relationships must be relatively low.

To confirm this belief about the need for deeper methods than pattern matching, I searched a 21 million word corpus of Wall Street Journal (WSJ) articles (see Section A.2 in the Appendix) for occurrences of the pattern "ADJ_1, *even* ADJ_2", where ADJ_1 and ADJ_2 are variables which can be instantiated to specific adjectives. This pattern corresponds to both versions of Horn's test given above, with the explicit *it is* omitted. While the pattern I used can be generalized to match more words by allowing a suitable multi-word expression to appear before the first adjective and also before *even*, I believe that the selected pattern will match more frequently than the original patterns from Horn's test and its extended counterpart.

The adjectives retrieved by searching for this pattern in the Wall Street Journal corpus appear in Figure 1.2. There are two important observations about the collected data:

- Only 42 adjectival pairs were retrieved from 21 million words of text. As I noted above, the pattern can be generalized by allowing for a complex expression to appear between the comma and *even*, provided that the same expression appears at the start of the pattern as well. However, it doesn't seem likely that such a change will increase the number of matching cases to an extent that this approach would be practical.
- Not only are too few matches retrieved, but also many of the retrieved pairs do not correspond to the relationship "ADJ_1 is from the same group as, but is semantically less strong than, ADJ_2". For example, along with *careless–slovenly* and *willing–eager*,

anti-cancer–anti-Communist	*nuclear–strategic*
blunt–rude	*objectionable–dangerous*
capable–popular	*overt–heavyhanded*
careless–slovenly	*possible–likely*
chaotic–untidy	*possible–probable*
combative–snappish	*powerful–shocking*
competitive–entrepreneurial	*reasonable–admirable*
confusing–divisive	*respectable–trendy*
convincing–hypnotic	*restless–obstreperous*
defensive–surly	*rich–Mahlerian*
difficult–dangerous	*rustic–raucous*
distasteful–sexual	*safe–sensible*
extreme–excessive	*spirited–meaningful*
far-fetched–inconceivable	*thin–small*
indifferent–contemptuous	*undignified–blasphemous*
insouciant–cheeky	*unethical–boorish*
intelligent–compassionate	*unseemly–unthinkable*
intricate–imponderable	*unusual–unprecedented*
irrational–undesirable	*vicious–sadistic*
lyrical–tender	*vigorous–relentless*
meaningful–enjoyable	*willing–eager*

Figure 1.2: Pairs of adjectives from the 1987 Wall Street Journal corpus (21 million words) which match the generalized Horn test pattern ("ADJ_1, *even* ADJ_2"). Within each pair the adjectives are listed in the order they matched, so that the second adjective in each pair should be semantically stronger if the adjectives belong in the same semantic group. Four pairs that were incorrectly matched (e.g, *least–strong* from the sentence *For the moment, at least, even strong opponents of the government say* ...) were removed from the figure and were not included in the number of retrieved pairs given in the text.

pairs such as *difficult–dangerous*, *meaningful–enjoyable*, and *powerful–shocking* match the pattern. This does not indicate a failure of Horn's test, as the test is designed to be applied on two adjectives from the same semantic group; whenever that is the case, the semantically stronger adjective appears indeed second (after *even*). However, the data shows that adjectives from different semantic groups can also match the test pattern, and thus the test pattern cannot be used by itself to automatically retrieve scales.

Therefore, direct pattern matching techniques are limited in the number of word pairs they retrieve from the text, and the retrieved pairs are not always instances of the semantic relationships we are interested in. As a result, I decided to abandon these techniques in favor of methods based on a deeper analysis of the text. I discuss the approach I adopted in the next section.

1.9 Methodology

The overall architecture of the system is depicted in Figure 1.3. The system consists of five main stages[14], each containing several substages and further discussed below. All stages use the corpus as their main source of information, actually as their only source of information besides the shallow non-lexical linguistic knowledge that is encoded in each subsystem. The first stage accepts a list of words of some specified category and produces groups of semantically related words. The second stage selects the groups that contain scalar and gradable terms, and therefore can be further subdivided. These groups are partitioned by the third stage in subgroups of different orientations, while the fourth stage actually labels the subgroups as positive or negative. There is a provision for a fifth stage (not currently implemented) that identifies ordering relations among members of the subgroups produced by the third stage.

Since the direct approach of searching for exact matches of linguistic test patterns in

[14] Four of which are implemented as part of the present dissertation.

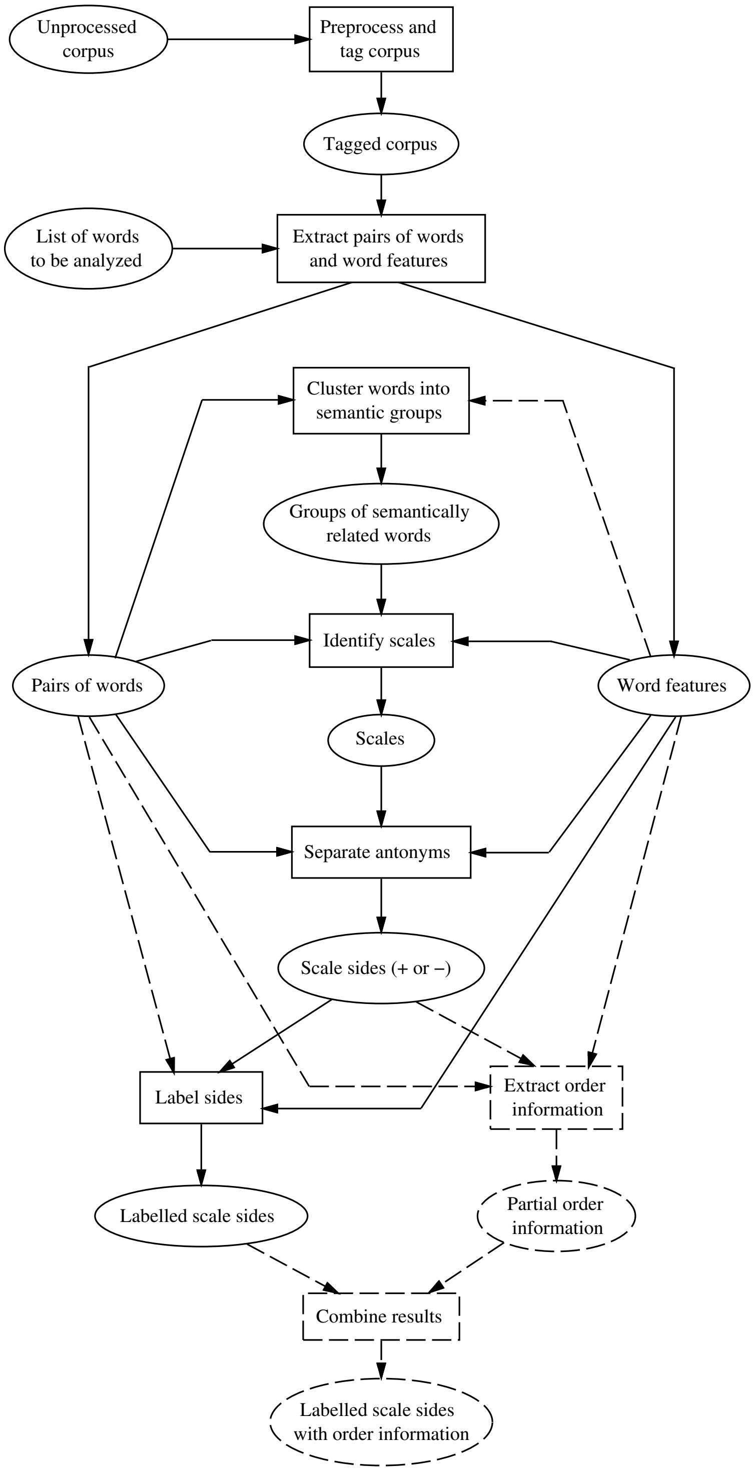


Figure 1.3: Overall architecture of the system. Solid lines indicate currently operational components and interconnections, while the dashed lines indicate subsystems and interconnections that may be implemented in the future.

a machine-readable corpus is not expected to produce useful results, we must look into other informative indicators of the semantics of the words of interest that are present in the corpus. Therefore, the first step towards the solution of the problem consists of identifying linguistic relationships that are likely to correlate with the semantics of the words of interest. For example, we may postulate that syntactically related adjectives and nouns are mutually constrained in their meaning, and consequently adjectives that are related semantically should exhibit similar collections of nouns that appear in the corpus with some kind of syntactic connection with these adjectives. Once such relationships are found, their surface manifestations in the corpus must also be identified. Continuing the above example, adjectives and nouns may be syntactically related through a modification relationship that can take several different surface forms (predicative, attributive, appositive, etc.).

The linguistic relationships that we are looking for can involve either only words from the set we currently analyze, or relate members of this set to words outside that set, possibly to words of different grammatical categories. The semantic incompatibility relationship between scalar words appearing in the same noun phrase which was discussed in Section 1.7.5) is an example of the first type of relationship for the word classification problem, as the observation of two consequent adjectives implies that they cannot belong in the same semantic group. The syntactic relationship between adjectives and nouns exemplifies the second type of relationship that can be implicitly used to measure the semantic relatedness of two adjectives.

No matter which of the two types the selected relationships belong to, they must be informative, and yet not extremely rare. The linguistic tests discussed in the previous section represent one extreme among such relationships; they directly relate two words that we are interested in and provide explicit information about their relative semantic strength, but they are very rare. Other more frequent surface patterns may correspond to semantic relationships only imperfectly. It is our first task to select patterns that satisfy both goals of frequency and informativeness about the underlying semantic relationship.

Such candidate patterns and relationships (e.g., the mutual semantic constraints between adjectives and nouns in the same noun phrase, or the likely modification of gradable adjectives by *very*) may be proposed on the basis of linguistic intuition. Their actual applicability to the task at hand can be verified in two ways: either by adding learning procedures based on them to a pre-existing prototype system for the task and showing that their inclusion leads to improved performance; or through *exploratory data analysis* of the corpus data, where one or more candidate relations are extracted from the corpus and the results (e.g., word groupings or semantic orientation labels) that they produce are compared with an objective evaluation standard. The latter approach is more appropriate when we suspect the relevance of a surface phenomenon to the semantics of the words of interest, but we are unsure of the form and strength of the correlation between that phenomenon and the semantic property being investigated. An example of this is the question of whether frequency of a term is associated with that term's semantic orientation (positive or negative); and if so, whether higher frequencies imply a positive or negative orientation.

In any case, once such relationships and their surface manifestations have been identified, we need to develop methods for extracting occurrences of them from the corpus. This involves at a first stage preprocessing the corpus to identify sentence and word breaks, and then developing procedures that extract the actual observed examples in the form of word pairs involving at least one word from the set of words of interest.[15] Naturally, the methods for extracting such pairs depend on the particular relationship and the syntactic classes of the member words. Also, the useful relationships vary according to the subtask of the problem that is currently being considered (e.g., while nouns may be useful indicators of the semantic relatedness of the adjectives that modify them, they probably cannot inform us of the semantic orientation or gradability of these adjectives). Consequently, the further treatment of the collected examples diverges at this point, and I will present appropriate methods for the tasks of identifying semantic and scalar groups and analyzing these groups separately in the following two subsections.

It should be noted here that both the corpus preprocessing phase and the actual extraction procedures can incorporate a varying degree of linguistic sophistication. The performance effects of the various alternatives are contrasted in Chapter 5, while the Appendix presents the implementation details of the preprocessing and data extraction steps.

1.9.1 Methodology for forming groups of semantically related words

Once relevant information has been collected from the corpus, in the form of word pairs involving at least one of the words to be clustered, the system computes a measure of sim-

[15] As noted above, depending on the type of the linguistic relationship, both of the words in each pair may belong to the set of words that we analyze or classify; or one of the words may belong to that set, and the other in a different category of informative words, such as modified nouns for adjectives during word group formation.

ilarity between any two of the words of interest separately for each identified relationship. Relationships that provide positive or negative examples about the semantic relatedness of two words can be incorporated in this framework. In this way, each relationship corresponds to an independent *similarity module* that processes a particular type of co-occurrence information and produces a real number between 0 and 1 for any two words that are to be grouped. A module may also refrain from making any judgement for a particular pair of words. All the similarity modules operate in parallel, at least conceptually.

The third substage of this subsystem (after the pair extraction substage and the independent similarity computations) combines the opinions of the various similarity modules into a single similarity measure for any pair of words. These combined measures of similarity are converted to dissimilarities by a fourth substage, and the latter drive the fifth substage, where a clustering algorithm places the words into groups according to the dissimilarity measure, so that words with a high degree of pairwise similarity fall in the same cluster (and, consequently, words with a low degree of similarity fall in different clusters).

This architecture, which is depicted in Figure 1.4, is flexible and allows the introduction of new independent similarity modules, as new types of co-occurrence information and semantic relationships become identified in the corpus. The only stage of the system that needs to be changed when a new module is incorporated is the algorithm that combines the similarity measures suggested by the independent modules into a single number.

I discuss in more detail the linguistic indicators used by the word grouping subsystem in Chapter 2, while the actual data extraction methods are presented in the Appendix.

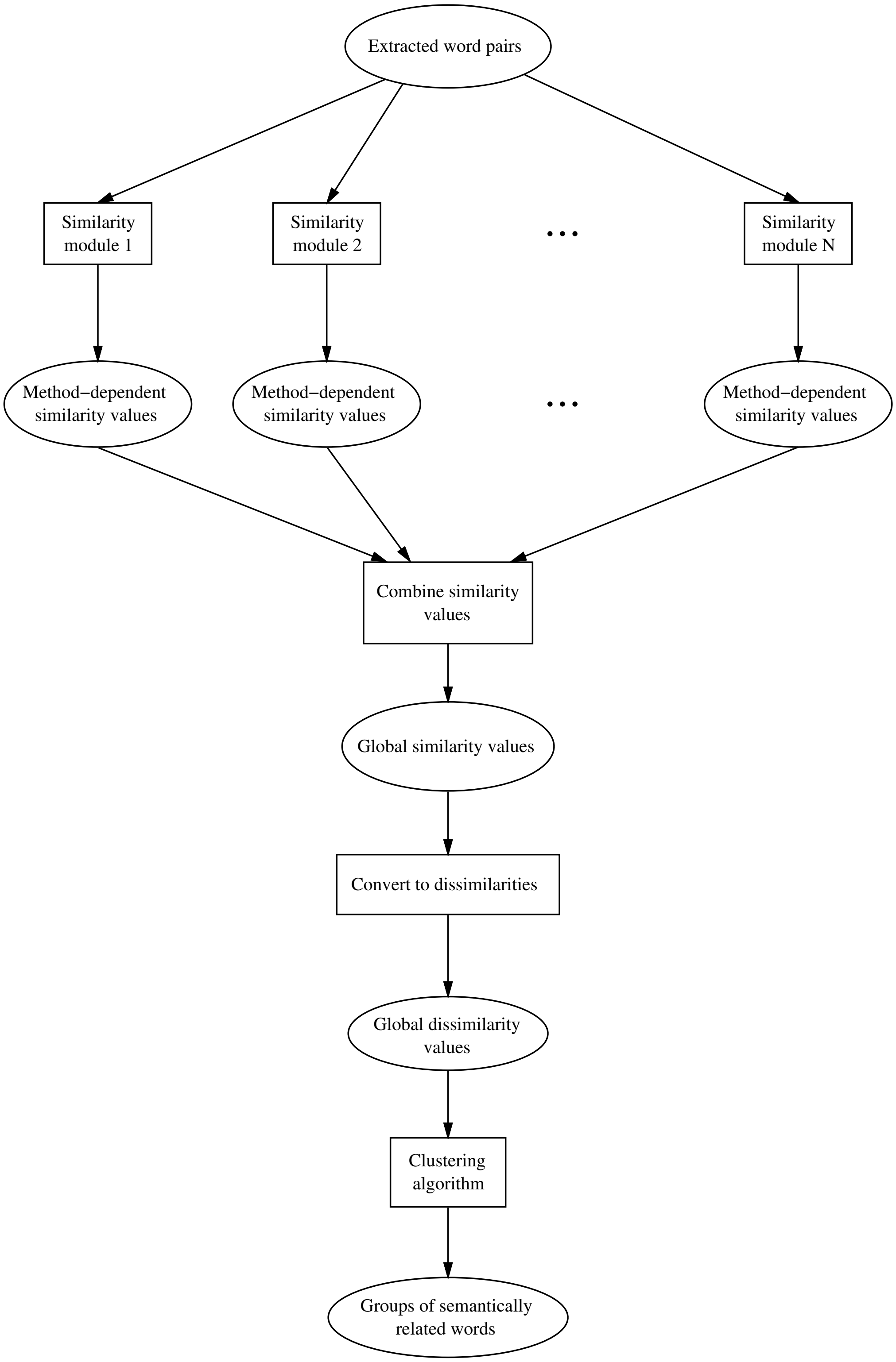


Figure 1.4: Architecture of the subsystem that partitions sets of words into semantic groups.

More details about the algorithms used for computing similarities, combining evidence from different modules, and performing the clustering are given in the later sections of Chapter 2.

1.9.2 Methodology for retrieving gradability and semantic orientation information

A similar data-driven approach has been adopted for the subsequent tasks of identifying scalar groups and finding the positive and negative subgroups of a semantic group or scale. Pairs that involve two words of interest and are indicative of these semantic relationships (i.e., gradability or semantic orientation) are extracted from the corpus. Other features of the words of interest, such as morphological complexity and frequency of occurrence, are also collected. Naturally, this is done for features that are expected to correlate with the semantic properties we are trying to identify. To find such features, I employ the methods of exploratory data analysis that were discussed in Section 1.4.

For the tasks described above, I have identified one or more linguistic properties that correlate with each semantic property I am trying to extract. For example, gradability has a strong positive correlation with scalarity (the property of a word being scalar), and semantic markedness is a strong indicator of semantic orientation. These linguistic properties and their correlation with the task at hand are discussed in more detail in the chapter corresponding to each task (Chapters 6–8), while methods for the extraction of this data from the corpus are given in the Appendix, in parallel with the corresponding methods for the word grouping task.

Since these linguistic properties offer an indirect approach for solving these problems, I have implemented an independent module for each such property. In addition, independent modules have been implemented for properties of words that have not been explicitly studied in linguistics but for which I have observed a meaningful correlation with the problems I am interested in. The occurrence of two words in the corpus coordinated by a conjunction such as *and* is an example of such a property, as these occurrences indicate that the two words share the same semantic orientation.

Each module receives one or more types of word pairs and feature values (or pairs of feature values for two words) collected or estimated at the first substage. Then, it converts the evidence retrieved from the corpus to a single decision for the indirect property we are studying. A numeric measure of confidence in the decision may also be produced by the module. A subsequent substage combines the decisions of the independent modules, producing a binary relation that expresses whether the semantic property we are interested in exists (or does not exist, for negative examples) between two words.[16] General learning techniques are employed to appropriately weigh the evidence offered by each independent decision according to the strength of the correlation of the indirect properties to the main semantic property of interest. Depending on the task, we may have only positive pairs of the form "the semantic property holds between these two words", or both positive and negative information. For example, polarity information is associated with two relations, one indicating that two words are on the same side of the scale, and another indicating that

[16]For some tasks, the result is in the form of a function from the set of words of interest to {TRUE, FALSE}, i.e., a unary relation (a mathematical set). This is the case for example for the problem of deciding which words are scalar in order to decide if a semantic group is really a scale.

two words are definitely on opposite sides (i.e., they are antonyms). These two relations may be assigned different weights by the combination substage, corresponding to differences in their informativeness or in the system's ability to correctly extract their instances. We may also have multiple relations that encode different aspects of the problem; for example, in the (future) subgroup ordering subsystem, we may have a unary relation indicating that a word is at the extreme of the scale, and a binary relation indicating that one word is semantically stronger than another word in the same scale.

All decisions up to this point are at a local level, i.e., they pertain to each particular word or group of words. These local decisions may be in conflict with each other; for example, the subsystem that identifies scales may have classified some words of a semantic group as scalar and some other words from the same group as non-scalar, a situation that causes a conflict when the entire group is to be labeled as scalar or not. When more than one relation is extracted for a task, conflicts may also arise between pairs in different relations. For example, in the subsystem that separates a semantic group into two subgroups of antonyms, one of the extracted relations may directly encode antonymy while another expresses that two words have the same polarity. Conflicts may arise immediately if a pair of words appears in both relations, but since the first of these relations explicitly negates the second and the second is an equivalence relation (and consequently a transitive one), conflicts can also arise indirectly. As an example of the latter case, suppose that the first relation includes the pair (X, Z) and the second includes the pairs (X, Y) and (Y, Z). Then the orientation that Z receives from the first relation relative to X is the opposite from the one assigned by the second relation through transitivity.

To resolve these conflicts and to combine the information present in multiple relations, a conflict resolution algorithm is employed. This substage roughly corresponds to the clustering algorithm of the previous subsection, in that it combines local information to produce local decisions that conform to global constraints. The algorithm can again weigh information from different sources (relations) according to the expected accuracy of each source in order to decide which of the proposed constraints will be violated, if such a violation is necessary (i.e., when the union of the locally derived constraints is not a consistent solution). This algorithm ultimately decides the semantic relationship (such as identical or different semantic orientation) between any two words in the original group, or classifies the entire group according to gradability or scalarity. While the same general algorithm is employed for revising the various local decisions to ensure the global consistency of the solution for all the stages related to the semantic analysis of the word groups, the parameters and the particular relations involved naturally change from stage to stage.

A generic model of the architecture of a subsystem that analyzes word groups is shown in Figure 1.5. As in the case of the initial subsystem that retrieves groups of related words, the architecture allows the expansion of these later subsystems with the easy incorporation of new types of co-occurring word pairs or features when such sources of information are identified and shown to correlate with the semantic relationship or property being extracted. A more mathematical description of the various statistical learning methods that these subsystems employ for tasks such as decision making, combining evidence from different linguistic properties, and resolving conflicts is given in the chapters on each individual subsystem (Chapters 6–8).

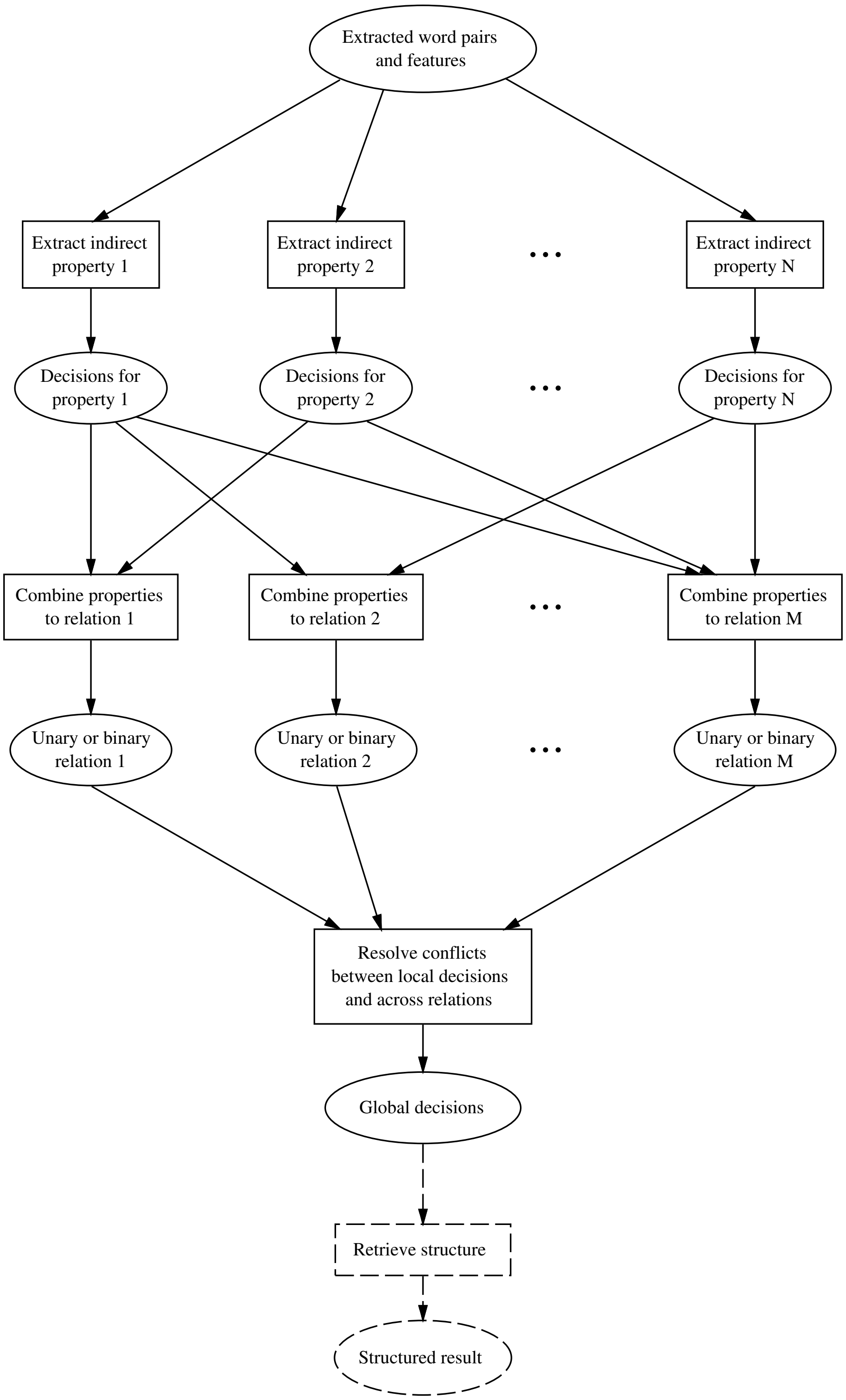


Figure 1.5: Generic architecture model of the subsystems that further process the retrieved semantic groups. The final component (dashed lines) is present only for subsystems where there is inherent structure in the result, such as in the ordering subsystem where words must satisfy a partial order relationship.

Note that different subsystems may interact because conflicting evidence can be found in the corpus.[17] For example, the subsystem that partitions groups may decide that two words should fall in different sides of a scalar group, while at the same time the subsystem that determines orientation may decide that both of these words have positive orientation. In this case, one of the subsystems may be given priority, or a more powerful conflict resolution scheme may be employed. This interaction may also extend to other pairs of subsystems, causing for example grouping decisions made by the first subsystem (see the previous subsection) to be revised according to evidence processed by a subsequent component. In the current implementation I use a simple ordering scheme where earlier decisions are not revised by later subsystems, leaving the complex topics of revising prior decisions and applying global constraints across subsystems as future extensions.

1.9.3 Evaluation methodology

There are two issues that need to be resolved during the design of an evaluation method. First, we need to decide the standard that will provide the correct answer and that the system should strive to match. Second, the procedure that quantifies discrepancies of the system's output from the ideal one should be specified.

To find the correct answer for each instance of the problems being considered, I rely on human evaluators that are asked to perform the same task as the subsystem being evaluated.

[17] Or, more likely, different pieces of evidence may lead different subsystems to conflicting decisions.

These humans are given the same input as the system (list of words to be grouped, groups to be further separated, etc.) and some description of the domain.[18] For practical reasons, I do not require that they be professional lexicographers, yet I do require from them a modicum of linguistics knowledge.[19] Due to the inexact definition of the tasks, the different interpretations of the words and different levels of granularity used by each evaluator, and even human error, the models produced by the humans are not the same but rather contain considerable differences. Therefore, we end up having *a set* of different solutions to the problem; I take these solutions collectively as the standard against which the system should be compared.

Instead of extrapolating a single "correct" model from this set of models, I compare the system's output[20] to the entire set. To this end, I have extended metrics such as precision and recall from Information Retrieval, and provided generalized versions that compare one set of "yes"–"no" answers to a collection of models of "yes"–"no" answers. These extensions are quite natural and the new metrics retain many of the important properties of their simpler counterparts; in particular, they are identical to the simpler versions when there is only one model solution or when there is perfect agreement among the model solutions.

[18]It would be impractical to provide them with a copy of the (very large) corpus used by the system.

[19]Another requirement that could be plausibly claimed of the evaluators is that they be native English speakers. I opted not to place this restriction on the evaluators for two reasons: First, it is unclear whether native speakers provide superior solutions for the kinds of problems this dissertation addresses over fluent non-native speakers (the situation would be entirely different in other circumstances, e.g., in modeling phonetic properties of the language). The second reason for including non-native speakers is practical: Given the present makeup of the graduate student body at Columbia University, restricting the evaluators to non-native speakers would considerably lower the number of available evaluators. I preferred the higher accuracy of a larger number of models to the uncertain increase in model quality that might result from using native speakers only.

[20]Or in general, any single solution, even one prepared by a human.

The above approach is sufficient whenever the output of a subsystem is a series of "yes"–"no" answers to a question, or can be modeled as such a series. This is the case, for example, when we are separating a group into positive and negative subgroups; the relevant "yes"–"no" questions are "Is this term positive?" and "Is this term negative?". When a subsystem produces output with a different structure, this structure needs to be converted to a series of questions of the above type in order for this evaluation method to be applicable.[21] I apply such a conversion method to the structures produced by the first subsystem, converting each word clustering into a collection of pairs of terms being placed in the same group. The resulting set of pairs captures exactly the same information as the original partition[22], and in this way we have a method for quantitatively measuring the similarity between two partitions of the same set, a difficult problem to which no answer has been supplied before.

1.10 Relevance and importance

The work reported here addresses several related problems in lexical semantics, more particularly in the acquisition of lexical semantic knowledge. It offers new methods for the four tasks of classifying words into semantic groups, identifying gradable and scalar words, and

[21] Actually, the previous example of polarity is a conversion of a question with three answers (*positive* versus *negative* versus *don't know*) to two interdependent questions with binary answers. In practice, since the current implementation never refuses to label a word (although it may assign a low confidence in its decision), I have to deal with only a single "yes"–"no" decision for each word.

[22] Although not all sets of pairs can represent valid partitions, because of structural constraints imposed by the partition.

separating and labeling words according to their semantic orientation. It also offers supporting empirical analyses of linguistic hypotheses and textual indicators related to these four problems, as well as a new formal evaluation approach that incorporates multiple reference models and a comparative assessment of the importance of shallow linguistic knowledge in the system.

Several of these problems have never before been addressed in computational linguistics. In Section 1.3, I discussed why the most pressing need for knowledge acquisition is for lexical knowledge, and why semantic knowledge is harder to extract than other forms of linguistic knowledge. In Section 1.4, I gave four additional important reasons that justify the use of an automatic, corpus-based system for the acquisition of lexical semantic knowledge. The present system adheres to the principles of full automation and lack of dependence on precompiled lexical sources (Section 1.5), can be easily ported to a different domain by simply changing the corpus (and the front end that extracts data from it), and its implementation brings the benefits of a fully automatic, domain adapting knowledge acquisition method described above for these four problems.

The output of the system can be used to significantly enrich an automated method for compiling lexicon entries with semantic knowledge. In fact, both the methods described here for the acquisition of semantic knowledge and other similar corpus-based probabilistic methods can also be applied to the acquisition of other types of linguistic knowledge (syntactic, morphological). The end result from such a system would be an automatically produced lexicon for human or computer use, a very important application for automated lexicography and computational linguistics in general. But even before this target is accomplished, the knowledge extracted by the system can be used in a variety of applications, both for understanding and generating natural language. These applications include more precise control of the lexicalization process in natural language generation [Elhadad 1991], compiling domain-dependent lists of antonyms and synonyms, interpreting negation, and correcting the erroneous use of multiple qualifiers that are superfluous or contradict each other, a phenomenon that has been observed in medical reports [Moore 1993].

In addition to the practical uses of the system's output, its design and implementation also offer theoretical contributions to computational linguistics and lexicography. The four major problems discussed above and subproblems related to them are addressed, and the evaluation results indicate that a reasonable level of performance can be achieved with the methods proposed here. Furthermore, the system's success in handling these problems is a measure of the potential of statistical learning systems for lexical knowledge acquisition. Since the system does not build on previously acquired lexical knowledge but only uses the corpus examples plus shallow grammatical linguistic knowledge, its success (as well as any consistent failures) can also help us understand what kinds of knowledge can be extracted with such "knowledge-poor" methods. In this way, the methodology presented in this dissertation can influence the approach taken for entirely different tasks. For example, parts of the work reported here have been found useful in designing and evaluating a system for the automatic identification of the aspectual class of verbs [Siegel 1997a; Siegel 1997b], while I am currently working together with Hongyan Jing, Rebecca Passonneau, and Kathleen McKeown on applying the group finding techniques to the problem of verb and noun sense disambiguation [Jing *et al.* 1997].

The exploratory data analyses that have been performed to confirm hypotheses about the informativeness of particular surface patterns to semantic relationships (e.g., the relationship between conjunctions of adjectives and constraints on their semantic orientation) contribute to (non-computational) linguistics in two ways. First, the results of these analyses offer new knowledge about the properties of language, sometimes rejecting widely held beliefs in the linguistics community. Because these analyses are based on actual data, they can settle questions that have been hotly debated on theoretical grounds for a long time, falsifying proposed theories and offering insight for new ones. For example, Chapter 8 presents the results of a corpus-based analysis of the performance of predictors for semantic markedness. These predictors have been proposed in the linguistics literature, but have never before been experimentally validated. The analysis in Chapter 8 reveals several surprising facts about the predictors, including that one of the simplest ones (text frequency) outperforms all others and contains almost all the information that can be extracted from all predictors together.

A second contribution to non-computational linguistics is that the adopted methodology, which relies on quantitative data and its statistical interpretation, can be adapted to many other linguistic questions. While the methodology is not in all its aspects original, it is unfortunately rarely used in linguistics, and its actual application to produce new results may be a positive step towards a more quantitative, formal, and *falsifiable* [Popper 1965; Lakatos 1972; Kempson 1977] linguistic science.

The thesis also covers two related but somewhat parallel to the main task areas: evaluation, and assessment of the relative importance of statistical and linguistic or knowledge-based methods in natural language processing. In the first of these areas, the approach is an example of formal evaluation which is to be contrasted with the "evaluation-by-example" paradigm that is frequently employed in computational linguistics.[23] With the generalization of information retrieval scoring metrics, the proposal of an evaluation method that simultaneously uses multiple models, the introduction of a quantitative comparison metric for partitions of sets, and the equalization of scores for partitions with scores for independent binary decisions, this work contributes to applied mathematics and information retrieval, since the methods are not limited to the task at hand or even natural language. The evaluation methods also produce, as a by-product, interesting results about the differences between human models and the difficulties the humans have when they build their models.

Quantifying the importance of linguistic information in statistical systems is an area that has received increased attention during recent years.[24] This work confirms expectations that such information is beneficial using quantitative (and thus less vulnerable to theoretical criticisms and easier to independently verify) methods. Preliminary results obtained so far also show some unexpected relationships between the effects of linguistic knowledge and the size of the training material, establishing the overall importance and indispensability of linguistic knowledge. This is another way that the present work can influence future natural language processing systems in a more general scope than that of the specific problems actually addressed in the thesis.

---

[23] And possibly in computer science in general; see [Tichy *et al.* 1995] for some disappointing comparisons of the extent of empirical validation in computer science research relative to other disciplines.

[24] A volume of collected articles on this subject has been recently published [Klavans and Resnik 1996].

1.11 The case for adjectives and nouns

Throughout this introductory chapter I have made no special reference to a particular grammatical category of words that are processed by the system. This is appropriate since the problems that the thesis addresses are general across all[25] open-class categories (adjectives, nouns, verbs, and adverbs). However, there are several parts of the system that need to be specialized to work with each particular syntactic class. These parts obviously include the methods that are used to extract pairs of words corresponding to some identified relation; different rules are needed for extracting adjectives that modify nouns than for extracting nouns that are subjects of verbs. Also, the informativeness of some surface patterns may change as the word categories change, or other relationships that apply to only a few categories may become available. This may make some of the modules that compute similarities or votes from word pairs and features irrelevant for some grammatical categories. It will also almost certainly necessitate changes in the algorithms that combine these similarities or votes in a single measure; at the very least, re-estimating the relative weights will be required.

Consequently, part of the implementation actually depends on the grammatical category of the words that are analyzed. While the overall approach remains the same, this dependence increases the complexity of the system according to the number of grammatical categories that are processed. From a research perspective, there is not much gain by having the system operational across all four open word classes, since similar extraction rules

[25]Or most, depending on the particular subproblem. See the discussion about nouns below.

and similarity or vote producing modules can be developed.[26] The main research gain from extending the system to multiple syntactic classes is to establish empirically the generality of the adopted approach and algorithms.

Note that many other probabilistic corpus-based systems treat the words in the text as simple units, symbols with no attached additional information except their position in the corpus. Such systems necessarily adopt shallow approaches, not using grammatical information. Unlike the present system, they can operate (with varying degrees of success) on any syntactic class, and even on texts from different languages, without any modification.

In view of the cost of incorporating new classes and the relatively reduced benefits, I have decided to limit this version of the system to two classes, adjectives and nouns. I initially developed the system for adjectives only, and consequently most of the examples and results that are presented in later chapters are drawn from this class. I selected this class because it is the prototypical syntactic category used in the linguistics literature for scales, antonymy, and other relevant topics such as markedness (see Section 8.1). I added to the first stage of the system the capability to process a second syntactic class, to show the extensibility of the approach to other word classes, i.e., that while some parts of the system depend on the particular grammatical category, the general approach does not and these parts can be replaced with similarly functioning modules. I selected nouns as this second category because of several reasons:

---

[26] Of course, from a practical perspective, the system needs to be able to operate on the class(es) of words that we are interested in.

Syntactic (part-of-speech) category	All possible uses of words		Primary uses of words	
Adjective	25,547	(16.94%)	8,817	(22.97%)
Adverb	4,243	(2.81%)	–	
Noun	77,355	(51.31%)	22,061	(57.47%)
Verb	30,725	(20.38%)	6,333	(16.50%)
Conjunction	108	(0.07%)	–	
Interjection	352	(0.23%)	–	
Past Participle	5,939	(3.94%)	–	
Preposition	230	(0.15%)	–	
Pronoun	134	(0.09%)	–	
Other	6,136	(4.07%)	1,179	(3.07%)
Total	150,769	(100.00%)	38,390	(100.00%)

Table 1.1: Distribution of syntactic (part-of-speech) categories in the MRC Psycholinguistic Database. Given the large size of this database, the distribution given can be considered as representative of the one in the language at large. The first column considers all entries in the database, while the second counts only the primary uses of words as adjectives, nouns, verbs, or in another category (e.g, *freeze* is counted as a verb, but not as a noun), using a subset of the word list. Percentages do not sum to 100% because of rounding. The table is reproduced from the data in [Wilson 1987, pp. 8–9]; the categories used were drawn from the Shorter Oxford English Dictionary [Dolby *et al.* 1963].

- The class of nouns is the largest class of words in the English language, with roughly half the total number of distinct words. Table 1.1 shows the number of words of each syntactic class that occur in the MRC Psycholinguistic Database [Coltheart 1981; Wilson 1987], a large and representative machine-readable annotated word list. Nouns account for 51.31% of the 150,769 word forms for which part of speech information is available in this database. If we consider only primary uses of the words (e.g., omit the more esoteric sense of *freeze* as a noun), 57.47% of the 38,390 word forms in the subset of the database for which this information has been computed are nouns.

- Most of the words that are continuously added to the language (neologisms and specialized terms) are nouns. This property, in combination with the first one, makes nouns the category for which semantic information is most unlikely to be found in a dictionary, and therefore the category for which automatic knowledge acquisition methods are both most needed and expected to have the largest impact.

- One of the possible future extensions to the dissertation is to run the system on a collection of hospital discharge summaries, to identify semantic relationships between medical terms (cf. Section 11.2). This knowledge would increase the coverage of the lexicon of the MAGIC multimedia system [Dalal *et al.* 1996a; Dalal *et al.* 1996b; McKeown *et al.* 1997]. Such terms are in most cases expressed syntactically as nouns or noun phrases.

- Nouns closely interact with adjectives, which allows for sharing the extraction grammar rules and offers in the future the opportunity of experimenting with the *simultaneous* learning of the semantic properties of words from both classes, using learned generalizations about one category to refine the decisions made for the other in an iterative manner. I may use one of the *two-way* or *block clustering* methods proposed in [Good 1965; Hartigan 1972; McCormick *et al.* 1972; Hartigan 1975; Chanson and Pinson 1981; Kaufman and Plastria 1981] for this task.

On the other hand, parts of the problems that the thesis addresses apply to nouns to a lesser degree than to other syntactic classes. These parts consist of the further processing of semantic groups, namely the identification of scales, the separation and labeling of positive and negative terms, and the (eventual) extraction of semantic strength information. While nouns can have antonyms and even form scalar groups (e.g., {*cold*, *heat*}, {*midget*, *giant*}, and {*rain*, *sleet*, *snow*}[27]), these semantic relationships are not as frequent between nouns as they are between adjectives. Consequently, and in order to keep the scope of the thesis somewhat manageable, the implementation processes adjectives and nouns up to the formation of semantic groups and continues with the further analysis and subdivision of adjectival groups only.

[27]Thanks to Kathy McKeown for pointing out these examples to me.

Part I

Finding groups of semantically related words

The first stage of the system formulates groups of semantically related terms from an unordered collection of words. Chapter 2 describes the linguistic relationships and indicators on which the grouping is based, and the algorithms that are used to measure semantic similarity between words and to cluster the words into groups. Chapter 3 discusses the evaluation approach I adopted, presenting methods for converting a partition of words to a list of "yes"–"no" decisions and generalizing evaluation metrics from information retrieval so that they use sets of multiple reference models. Output from the system and the results of the evaluation process are given for several sets of adjectives and nouns, drawn from the general news and financial domains. I close this part by presenting in Chapter 5 a corpus-based analysis of the significance of linguistic knowledge in improving the performance of the base statistical grouping system.

Chapter 2

Linguistic indicators and statistical algorithms for the word grouping problem

My approach for classifying words into groups according to semantic relatedness is based on relevant *syntagmatic relationships* that are informative for this purpose and relatively frequent in the text. The term "syntagmatic relationship" (from the theory of structuralism[1] [de Saussure 1983]) refers to relationships between words that co-occur in a sentence and that fulfill particular roles in the structure of the sentence. For example, a modifier word and the word it modifies participate in a syntagmatic relationship, which may be expressed on the surface with one of several syntactic forms that further depend on the grammatical categories of the two words. An adjective and a noun can, for instance, fulfill the roles of modifier and modified word in the above example, and their deep syntagmatic relationship

[1] In the sense of European structuralism, not of American post-Bloomfieldian structuralism.

can be expressed on the surface in several different ways (e.g., *tall woman*; *that woman is tall*; *the woman, tall and slender, ...*). In Section 2.1 I present these syntagmatic relationships between words in the corpus and I then consider their realizations when particular grammatical categories are considered. In the next section, I assess each of these realizations in terms of its potential as an indicator of semantic relatedness and describe which of them I have incorporated into the system.

Once the sources of knowledge I rely on and their manifestations as syntactic relations have been identified, I turn to the actual algorithms that I use to produce semantic groups. In Section 2.3 I discuss the similarity metric that I adopted and its desirable properties for this task, contrasting it to alternative measures. This similarity computing module combines all information from positive association examples into a single number for each pair of clustered words, which is then transformed into a dissimilarity value between 0 and 1 as described in Section 2.4. A corresponding (but simpler) module combines negative examples for each pair of words to decide whether the two words are definitely semantically unrelated. This module is described in Section 2.5, followed by the presentation of the algorithm that puts together the information from positive and negative examples into a single overall dissimilarity measure for each pair of clustered words. The last component of this stage feeds the overall dissimilarities to a non-hierarchical clustering algorithm that produces the final groups. The clustering algorithm I chose and the reasons for doing so are presented in Section 2.7, along with a discussion of the computational complexity of the task and of the thorny problem of determining the optimal number of groups for a given collection of words. In that section, I also argue that although it might appear that overlapping classes would help for polysemous words, their practical usefulness is in practice limited, and the present approach of placing each word in a single group offers alternative advantages.

2.1 Syntagmatic relationships relevant to the grouping problem and their category-specific realizations

I have identified three types of generic syntagmatic relationships that correlate with the semantic relationship of "relatedness" between words from the same syntactic category.

1. A word X from a grammatical category A modifies, or takes as a syntactic argument, a word Y from a different grammatical category B.
2. A word X from a grammatical category A modifies, or takes as a syntactic argument, a word Y from the same grammatical category A.
3. Two words X and Y from the same grammatical category A simultaneously modify, or take as a syntactic argument to fulfill the same syntactic role, a word Z from any grammatical category, possibly even A.

Table 2.1 shows the actual realizations of the above generic relationships that are possible in English for the four open-class grammatical categories. Examples from the 1987 articles in the Wall Street Journal (WSJ) corpus (see Section A.2 in the Appendix) are also included in

Class	Realization	Example
1	adjective modifies noun	*hydraulic press*
	adverb modifies adjective	*highly speculative*
	adverb modifies verb	*previously reported*
	noun is subject of verb	*production declined*
	noun is direct object of transitive verb	*suspend operations*
	noun is indirect object of bitransitive verb	*gave voters (a list of issues)*
2	adjective modifies adjective	*alleged illicit*
	adverb modifies adverb	*unusually quickly*
	noun modifies noun	*unemployment compensation*
3	two adjectives modify a noun	*supplemental medical insurance*
	two adverbs modify an adjective	*chronically mentally ill*
	two adverbs modify a verb	*probably severely impair*
	two nouns modify a noun	*investment tax credit*

Table 2.1: Realizations of syntagmatic relationships for the four open classes of words.

the table. I have omitted verb-verb relationships where a verb appears in finite or nonfinite form as an argument of another verb, as these relationships seem to imply little or no semantic relationship between the participating verbs.[2] Furthermore, such relationships are syntactically quite complex and consequently hard to extract from the text with high accuracy.

Some other possible realizations of the generic syntagmatic relationships have not been included in the table because they are highly idiomatic, specialized, or involve a very small number of open-class words. Examples of such relationships are adverbs modifying prepositions (e.g. *well before*) or nouns (e.g. *the then chairman*); both of these cases are limited to very few adverbs. The specialized character of these relationships makes them unlikely candidates for general indicators of semantic relatedness, since even if they do correlate with

[2] Note that for other kinds of information such relationships may be valuable. As an example, consider the aspectual information extractable from the validity of *forced to run* when it is contrasted with **forced to resemble* [Dowty 1979; Klavans and Chodorow 1992].

semantic constraints, they apply to only a small subset of the words we may be interested in.

I have also omitted from the table two manifestations of the third generic class of syntagmatic relationships, namely those involving two adjectives modifying a third adjective or two adverbs modifying a third adverb. While such relationships and the corresponding syntactic patterns are possible in theory, I couldn't find any examples of them in the 21 million word WSJ corpus[3], which indicates that whenever three or more adjectives or adverbs co-occur they tend to modify some other word in the sentence, i.e., a noun or verb. In fact, I can think of somewhat contrived examples of such class 3 relationships involving adjectives or adverbs, for example *a greenish light red scarf* or *absolutely unequivocally positively*; but their scarcity in the corpus shows that their usefulness for the purposes of this thesis is extremely limited. On the other hand, sequences of more than two adjectives and adverbs (especially adjectives) that modify a previous or subsequent term are not uncommon; for example, *coherent domestic economic program* or *Too many of his attitudes, claims and complaints are careless, conflicting, dubious, inaccurate, mean, petty, simplistic, superficial, uninformed, and/or pointlessly biased* (Washington Post).[4]

Relationships belonging to the first class are likely to be useful as indirect positive examples of semantic relatedness for two words from category A that share a similar set of modified words (or arguments) from category B. This can be justified on distributional

[3] I searched the corpus for patterns of the form X_1 *(,)* X_2 X_3, where the X_i's are adjectives or adverbs and the comma is optional.

[4] Commas, however, as used in the last example, may negate the implication of semantic incompatibility. This point is further discussed in Section 1.7.5, and in Section A.4.1 in the Appendix.

properties of language; as Firth [1957] said, “words shall be known by the company they keep”. The fact that two words belong in the same semantic group implies that they describe the same property or have common semantic features. This in turn places semantic constraints on the words that can be modified or used as arguments by them. For example, temperature is normally defined for physical objects and we can expect to find that adjectives conveying different values of temperature will all modify physical objects; most likely, the set of nouns modified by such adjectives will be further constrained to physical objects that their temperature is likely to be talked about. The existence of these constraints has been recently[5] verified empirically by Justeson and Katz [1995a], who examined the constraints that modification of particular nouns places on the sense of the modifying adjectives. I exploit these constraints in reverse, by observing in the corpus patterns of common sets of modified words and using them to establish the semantic relatedness of their modifying words. Since the constraints go in both directions, we can also use common sets of modifiers from category A to establish the relatedness of words from category B. However, for these approaches to work, the relationship must have a high correlation with the semantic features of the words, something that may depend on the grammatical categories involved; it is possible, for example, that an adjective-noun modification relationship is semantically more restricted than an adverb-adjective modification relationship to the extent that the former is useful for our purposes while the latter not.

Relationships from the other two classes offer an altogether different kind of evidence. They involve two words from the same category A in a hypotactic relationship (class 2), or in

[5]Subsequent to the first publication of the arguments I present in this section.

a paratactic relationship where the two words from category A occupy the same semantic slot in the semantic description of the word from category B (class 3); see Table 2.1 on page 87 for actual examples from the various syntactic categories. In Section 1.7.5 I analyzed these two cases and showed that two words that occur in relationships of either type cannot be members of the same semantic group. Therefore, these relationships offer direct negative examples to the partitioning subsystem.

2.2 Usefulness of the realizations of syntagmatic relationships

In the previous section I discussed syntagmatic relationships that in principle can offer information about the semantic relatedness or unrelatedness of the words that participate in them. I also presented the more specific realizations of these relationships when particular grammatical categories are considered. In this section I evaluate each of these realizations as to how likely it is to actually provide information about semantic relatedness.

2.2.1 Indirect indicators of semantic relatedness

Three modification (adjective-noun, adverb-adjective, adverb-verb) as well as three head-argument (subject noun-verb, verb-direct object, verb-indirect object) relationships of this type were presented in Section 2.1. Of these, the adjective-noun and verb-direct object relationships most strongly express semantic correlation, through the use of semantic selectional restrictions. Subject-verb relationships are less constrained, and therefore I believe such co-occurrences to contain some information for the relatedness problem but not of the same quality as verb-object relationships. Semantic constraints between a bitransitive verb and its indirect object are minimal; usually simply that the indirect object is animate. Furthermore, proper nouns and pronouns are used as indirect objects quite frequently, more than for other noun phrase syntactic roles in the sentence. Because of these reasons, I do not expect the verb-indirect object relationship to be of value as an indicator of semantic relatedness.

Relationships involving adverbs are of intermediate value. While there are semantic constraints that an adverb and the item it modifies must satisfy, many adverbs are principally used as intensifiers (e.g. *largely*, *mainly*, *extremely*), at least when they modify adjectives. Two such intensifiers may be employed to heighten the effect, for example *very very good*. Consequently, many adjectives (and sometimes verbs) can co-occur with such intensifying adverbs and the co-occurrence information offers little knowledge about the adjective or verb beyond gradability.[6] This problem also exists with adjective-noun relationships but to a much smaller extent. Although some adjectives (e.g. *certain*, *slight*) typically function as intensifiers, they constitute only a small percentage of all adjectives. Furthermore, most actual uses of adjectives in the text are for the purpose of conveying additional semantic information about the modified item, and not for an intensifying effect.

[6]Gradability information is conveyed through the use of a scaling intensifier such as *extremely*, or, for a prototypical example, *very*.

2.2.2 Direct indicators of semantic unrelatedness

Turning to the direct negative indicators of Section 2.1 (types 2 and 3 of the syntagmatic relationships), I observe that for adjective-adjective and noun-noun pairs, the relationships offer strong evidence of incompatibility and therefore such pairs are likely to be useful for the grouping problem. On the other hand, adverb-adverb modification relationships (type 2) do not indicate semantic unrelatedness, since in such relationships the modifying adverb always acts as an intensifier [Quirk *et al.* 1985, p. 449] (e.g., *especially well*, *slightly faster*). Compare *psychologically crucial* where the adverb adds to the semantic content of *crucial* by appropriately restricting/specifying it, with **psychologically crucially* which would have been acceptable only if *psychologically* could function as an intensifier. When the modifying adverb already has such an intensifying or grading role, no problems arise when an adjective is replaced by the corresponding adverb; e.g., *unusually sharp* – *unusually sharply.*

Adverb-adverb relationships that belong to the third type of syntagmatic relationships (two adverbs that modify an adjective or verb) are not subject to this restriction and can therefore offer evidence of unrelatedness. However, the same problems that were mentioned in the previous subsection affect such relationships, namely the preponderance of intensifiers among adverbs and the possibility that two (or more) separate intensifiers are used simultaneously although they are not semantically incompatible. Consequently, while adjective-adjective and noun-noun relationships that indicate semantic incompatibility can be trusted to a high degree, our confidence in similar adverb-adverb relationships is much lower.

2.2.3 Selecting syntagmatic relationship realizations for inclusion in the system

In the previous two subsections I discussed the validity of the syntagmatic relationship realizations of Section 2.1 for the purposes of providing positive or negative information about semantic relatedness. Naturally, we want to include in the system the relationships that are likely to offer at least partially useful indications of relatedness.[7] But the relationships that are included in the system must satisfy two additional criteria:

1. It should be possible to extract instances of the relationship from the corpus automatically and with a relatively high precision. Accuracy is more important for the direct negative indicators than the indirect positive ones, since a single instance of the former can lead to the conclusion that two words are semantically unrelated.

2. Instances of the relationships should be frequent enough in the corpus, so that statistical methods applied to counts of such instances are meaningful. This is the classical sparse data problem which is a concern for all statistical learning systems. This requirement is more important for the indirect (positive) indicators, since for the negative indicators even a single occurrence can be convincing of unrelatedness, assuming that the indicator is strongly correlated with semantic incompatibility and that we can extract instances of the relationship from the corpus with high accuracy.

[7] We also want to weigh the indications of the various relationships according to their expected correlation with semantic relatedness. This is a topic I return to in Section 2.6.

The first requirement is satisfied by all relationships that were presented in Section 2.1; Sections A.3–A.5 in the Appendix give the details about an efficient and reasonably accurate extraction process that has been implemented. For the second requirement, I searched a portion of the Wall Street Journal corpus for the various patterns of Table 2.1 and manually analyzed the relationships between the words in the retrieved sequences. I observe that adjective-noun, subject-verb, and verb-object relationships are very frequent in the corpus, while adverb-adjective and adverb-verb relationships are of medium frequency and verb-indirect object relationships are of low frequency. Type 2 noun-noun relationships are also quite frequent, while type 2 adverb-adverb relationships are less frequent and type 2 adjective-adjective relationships are rare. For the third type of relationships, adjective-adjective relationships have a low frequency, while noun-noun relationships are rare; adverb-adverb relationships have a low frequency when the modified item is a verb, and a much lower frequency when the modified item is an adjective. Extensions of all the above relationships to sequences of more than two words from the same category are much less frequent than their basic counterparts, with their relative frequencies following the distribution of the relative frequencies of the basic two-word forms.[8] The above qualitative assessments of frequency are principally based on the empirical evidence collected from the corpus, and are supplemented by my own use of the language and experience with other corpora.

Table 2.2 presents the conclusions of the above discussion along with the results obtained in the previous two subsections. The decision of which syntagmatic relationship realizations to include in the system is also affected by the fact that both the type 2 and type 3 forms

[8] Allowing for quirks in the distribution due to collocations and other non-compositional constraints.

Relationship realization	Quality	Frequency
Positive indicators (type 1)		
Adjective-noun	High	High
Adverb-adjective	Low	Medium
Adverb-verb	Medium	Medium
Verb-subject	Medium	High
Verb-direct object	High	High
Verb-indirect object	Very low	Low
Negative indicators (type 2)		
Adjective-adjective	High	Very low
Adverb-adverb	Very low	Low
Noun-noun	High	High
Negative indicators (type 3)		
Adjective-adjective (modifying noun)	High	Low
Adverb-adverb (modifying adjective)	Low	Very low
Adverb-adverb (modifying verb)	Medium	Low
Noun-noun (modifying noun)	High	Very low
Combined negative indicators		
Adjective-adjective	High	Low
Adverb-adverb	Low	Low
Noun-noun	High	High

Table 2.2: Qualitative assessment of the syntagmatic relationship realizations under consideration, based on my experience with the corpora and my own use of the language. Characterizations of extractability have been omitted since all relationships can be retrieved reasonably efficiently and accurately.

of a relationship are expressed with identical syntactic constructs. For example, a sequence like *adjective1 adjective2 noun* may correspond to a type 2 underlying relationship (as in the case of *light blue scarf*) or to a type 3 relationship (as in the case of *confidential financial data*). This structural ambiguity of the English language, which makes the problem of deciding which word modifies what quite hard, forces us to combine the evidence from both types; without resorting to semantic analysis, we cannot determine the scope of modifiers and have no other choice. The combination of the two types of relationships implies that an information source of high quality can be contaminated from a source of lower quality.

This is not a problem for adjectives and nouns where both types offer reliable information in favor of the semantic unrelatedness of the modifiers. But for adverbs, where type 2 relationships do not imply semantic incompatibility, the inadvertent combination of types 2 (where one word modifies the next) and 3 (where all modifiers modify the final word in the sequence) lowers even more our confidence in that negative indicator.

According to this analysis, I have decided to include in the system the following relationships for adjectives: adjective-noun pairs as an indirect positive indicator and adjective-adjective pairs as a direct negative indicator. For nouns, I initially was planning to use adjective-noun, verb-direct object, and subject-verb pairs as positive indicators, and noun-noun pairs as a negative indicator. However, I found that using just the adjective-noun positive indicator, enough data was collected to form reasonably accurate semantic groups of nouns (see Section 4.2 in Chapter 4). The extension to nouns was made to the original adjective-based system to provide empirical support for the generality of the adopted approach; thus, achieving the highest possible performance for nouns was not the most critical goal. At the same time, pairs of syntactically related adjectives and nouns were already extracted by the finite-state grammar described in the Appendix, while retrieving verb-direct object and subject-verb relationships would require major extensions. Thus, I've decided to limit the current indicators for nouns to adjective-noun pairs as positive examples and noun-noun pairs as negative ones.

While I have not implemented the necessary extensions to the grouping subsystem so that it can cover verbs and adverbs (according to the arguments I gave in Section 1.11), both of these cases have no negative indicators, and the corresponding positive indicators for verbs would be subject-verb, verb-direct object, and verb-adverb pairs. The positive indicators for adverbs would be adverb-verb and adverb-adjective pairs. I am currently considering implementing some of these positive indicators for verbs, in the context of a system for the indirect resolution of sense ambiguities [Jing *et al.* 1997] outside the scope of this thesis.

The relationship between adverbs and adjectives, a positive indicator, is another one which may prove useful for the grouping of adverbs or adjectives[9], although I have argued above on its probably limited usefulness. I have decided not to include adverbs as a secondary indicator for the semantic relatedness of adjectives because of the prevalence of emphasizers among adverbs and the low correlation of adverbial modification with semantic content.

2.3 Estimating similarity values

After the completion of the first stage of the grouping subsystem, we have extracted from the corpus a large number of instances of binary syntagmatic relationships. This is an open set, potentially containing combinations of tens of thousands of distinct words (depending on the size of the corpus). Some of these word pairs constitute indirect positive evidence in favor of the semantic association of two words from a syntactic class A by showing similar

[9] Most likely more important for adverbs, since for adjectives we can rely on the more frequent and higher quality indicator of adjective-noun pairs.

Adjective	Noun					
	...	competition	economy	model	strategy	...
⋮		⋮	⋮	⋮	⋮	
global	...	18	32	2	17	...
international	...	31	34	0	6	...
⋮		⋮	⋮	⋮	⋮	
new	...	27	4	231	94	...
old	...	0	3	31	8	...
⋮		⋮	⋮	⋮	⋮	

Table 2.3: Sample of positive evidence for semantic association between adjectives (rows) on the basis of nouns (columns). The frequencies have been extracted from the 1987 WSJ corpus.

patterns of co-occurrence with words from another syntactic class B, while others indicate that two words from the same syntactic class are semantically unrelated. By combining pairs which involve the same two words, we obtain for the positive evidence a table of frequencies with the words from class A corresponding to rows and the words from class B corresponding to columns, and for the negative evidence a list of word pairs with their observed frequencies. Tables 2.3 and 2.4 present a small part of these tables of frequencies of adjective-noun and adjective-adjective pairs, obtained from the 1987 Wall Street Journal corpus (see Section A.2 in the Appendix).

As discussed in Section 2.1, the degree to which two different words from category A appear with the same words from category B in the table of positive evidence can be used to measure the semantic relatedness between them. That is, when we classify adjectives on the basis of nouns, the working hypothesis is that adjectives which tend to modify the same nouns are likely to be semantically related. I formalize this concept by viewing each row in this table as the empirical distribution of words from category B (e.g., nouns) for the

Pair of adjectives		Frequency of occurrence
⋮		⋮
annual	– current	22
annual	– high	13
annual	– small	3
big	– daily	2
big	– local	3
chief	– international	15
clean	– dry	1
current	– national	10
daily	– national	5
different	– local	1
difficult	– new	2
final	– possible	1
⋮		⋮

Table 2.4: Sample of negative evidence that can be used to establish semantic unrelatedness between adjectives. The frequencies have been extracted from the 1987 Wall Street Journal corpus.

particular word from category A (e.g., adjective) that corresponds to that row. Then, we can employ the isomorphism *word from category* $A \mapsto$ *random variable with the distribution of the corresponding row of the table of positive evidence* to define the similarity between any two words from category A as the similarity between the *distributions* of two random variables. These distributions are expressed by the two rows corresponding to the two words.

Several measures of association can be used for comparing two distributions, including mutual information [Fano 1961; Church and Hanks 1990], the cosine measure [Romesburg 1984], the (parametric) Pearson product moment correlation coefficient, Euclidean and average Euclidean distance [Sneath and Sokal 1973], Manhattan (L_1), generalized Minkowski (L_q), and Mahalanobis [1936] distances, and a variety of coefficients regularly used in psychology, biology, and the social sciences, such as the simple matching coefficient [Zubin 1938; Dumas 1955], the Jaccard coefficient [Jaccard 1908; Sneath 1957], and the Dice or Sorensen coefficient [Dice 1945; Sörensen 1948].[10] Statistical measures based on tests such as the Kolmogorov-Smirnov chi-square [Daniel 1990] and the normalized chi-square (ϕ) [Lindeman *et al.* 1980; Gale and Church 1991], and rank-invariant tests such as Spearman's rank correlation ρ [Spearman 1904; Spearman 1906] and Kendall's τ [Kendall 1938; Kendall 1975] are also potential choices. Some of the measures mentioned above operate by converting the frequencies to binary measurements (zero or non-zero), e.g., the simple matching, Jaccard, and Dice coefficients; others operate on the relative ranks of the frequencies (e.g., Kendall's and Spearman's coefficients); still others, including mutual information, the cosine measure, the Pearson product moment correlation coefficient, and many of the distance metrics, use the full quantitative information present in the frequency measurements.

I have implemented and experimented with several of the above measures, namely mutual information, the cosine measure, Euclidean and average Euclidean distances, and Kendall's τ. I have found Kendall's τ to outperform the other measures experimentally, and this difference in performance can also be explained on theoretical grounds. Kendall's τ compares the two random variables by repeatedly comparing two pairs of their corresponding observations. Formally, let (X_i, Y_i) and (X_j, Y_j) be two pairs of frequencies for the words X and Y that we are interested in grouping with the words i and j from the category that is used as evidence of association respectively. For example, when the similarity of

[10] I impose the requirement that the measure of association should be symmetric with regard to the two variables, so that $sim(X, Y) = sim(Y, X)$. This excludes some non-symmetric measures that have been used for this task, for example relative entropy (the Kullback-Leibler distance) [Cover and Thomas 1991].

Pair of nouns	Pair of adjectives					
	global–international	global–new	global–old	intern.–new	intern.–old	new–old
competition–economy	+	−	+	−	+	−
competition–model	+	−	−	−	−	+
competition–strategy	+	−	−	−	−	+
economy–model	+	−	−	−	−	+
economy–strategy	+	−	−	−	−	+
model–strategy	+	−	−	−	−	+
Total concordances	6	0	1	0	1	5
Total discordances	0	6	5	6	5	1

Table 2.5: Concordances and discordances for adjectives on the basis of nouns using the sample of adjective-noun pair frequencies given in Table 2.3. Concordant pairs are marked with a "+", while discordant ones with a "−".

adjectives is measured on the basis of modified nouns, X and Y are adjectives and i and j are nouns. I will call these pairs *concordant* if $X_i > X_j$ and $Y_i > Y_j$ or if $X_i < X_j$ and $Y_i < Y_j$. If $X_i > X_j$ but $Y_i < Y_j$, or if $X_i < X_j$ and $Y_i > Y_j$, the two pairs are *discordant*. Ties, that is pairs of observations where $X_i = X_j$ or $Y_i = Y_j$, are handled in a special way described below. As an example, considering the data of Table 2.3, I observe that the two nouns *competition* and *economy* are a concordant pair (a *concordance*) for the adjectives *global* and *international*, since *economy* appears more frequently than *competition* for both adjectives. The same nouns are a discordant pair (a *discordance*) when the adjectives *global* and *new* are compared, since *economy* appears more frequently than *competition* when the modifying adjective is *global*, but less frequently when the modifying adjective is *new*. Table 2.5 shows the results of these comparisons carried on all $\binom{4}{2} \cdot \binom{4}{2} = 36$ combinations of pairs of the four adjectives and pairs of the four nouns given in Table 2.3.

In general, if the distributions for the two random variables are similar, we expect a large number of concordances, and consequently a small number of discordances since the total number of pairs of observations is fixed for the two variables. This is justified by the fact that a large number of concordances indicates that when one of the variables takes a "large" value the other also takes a "large" value on the corresponding observation; of course, "large" is a term that is interpreted in a relative manner for each variable.

Kendall's τ is defined as

$$\tau = p_c - p_d$$

where p_c and p_d are the probabilities of observing a concordance or discordance respectively. τ ranges from -1 to $+1$, with $+1$ indicating complete concordance, -1 complete discordance, and 0 (i.e., $p_c = p_d$) no correlation between X and Y.

When no ties exist, an unbiased estimator of τ is the statistic

$$T = \frac{C - Q}{\binom{n}{2}} \tag{2.1}$$

where n is the number of paired observations in the sample and C and Q are the numbers of observed concordances and discordances respectively [Daniel 1990]. In the presence of ties, the value of T specified by equation (2.1) consistently underestimates the true value of τ. Kendall [1975, p. 35] provides an alternative estimator for τ when ties exist, given by

$$T_b = \frac{C - Q}{\sqrt{(\binom{n}{2} - t_X) \cdot (\binom{n}{2} - t_Y)}}$$

where t_X and t_Y are the numbers of tied pairs in the comparisons within X and Y values respectively. I use this revised estimator in the grouping subsystem.

Since the table of positive evidence is quite sparse, with many words from category B not having been observed with most of the words from category A, I further modify the definition of similarity on the basis of the statistic T_b so that in the calculation of the similarity between words X and Y, a pair of pairs (X_i, Y_i) and (X_j, Y_j) is not counted as a tie if all four frequencies X_i, Y_i, X_j, and Y_j are zero, but rather is ignored. This modification, which results in a modified statistic T^*, allows the informative part of the table of positive evidence (i.e., the pairs of words that actually co-occurred with at least one of the words we are grouping) to determine the value of T^*. Without this change, most of the observations would have been ties of zero frequencies, driving T_b close to 0. This change also makes T^* an *asymmetric* measure of association, one that gives more weight to the actually observed pairs. I have argued elsewhere [Smadja *et al.* 1996] that this asymmetric property is desirable for measuring lexical association, because it assigns a larger significance to the most important observations, the words that have been seen in the corpus together with the words of interest.

The adjusted statistic T^*, as well as the statistics T and T_b, present a computational disadvantage when compared with other similarity measures, since for n common observations between two words (e.g., n pairs of noun frequencies for two adjectives), all $\binom{n}{2} = \mathrm{O}(n^2)$ combinations of them must be considered. Consequently, measuring similarity with a straightforward implementation of Kendall's τ takes quadratic time in terms of the observations for each pair of words, contrasted with linear time for most other similarity measures. However, this disadvantage of Kendall's τ is offset by the following advantages that it offers:

- It requires no normalization for discrepancies in the magnitude of the frequencies of the two variables. This is not the case for other measures, for which adjustment of the frequencies to a common scale is necessary (e.g., the cosine measure and Euclidean and Manhattan distances).

- As a special but important case of the above, Kendall's τ is not sensitive to linear transformations of one or both of the variables, which correspond to a shift and/or scaling of the distribution but preserve the distribution's shape.[11] This is desirable behavior, as we do not want the similarity measure to be affected by such changes in the variables; the *shape* of the two distributions is exactly what we are interested in comparing. More generally, Kendall's τ is not changed by any isotonic (order-preserving) transformation of any of the variables. This is not true for other similarity measures, notably mutual information, which are quite sensitive to many isotonic transformations, including linear ones, sometimes with surprising negative effects [Smadja *et al.* 1996].

- Kendall's τ produces a measure of similarity in a bounded range (-1 to $+1$), unlike mutual information which is unbounded.[12] This property makes the interpretation of the estimated values of similarity and the combining of these values with similarity values produced by a different module more straightforward.

- With the further modification of ignoring zero-zero pairs, the adjusted statistic T^*

[11] This strictly applies to the unmodified T_b. The version I use will produce a different value if the transformation introduces non-zero values in some formerly zero-zero pairs (or vice versa). Note that this can happen only after a shift operation, not after a scaling one, and that the magnitude of the change will be small for such shift operations, as long as reasonable word frequency distributions are maintained.

[12] Of course, mutual information is bounded by the entropy of each variable [Cover and Thomas 1991], but this quantity in turn depends on the total number of observations as well as the distribution of the variables.

is an asymmetric measure of association, in which the presence of an observation is given more weight than its absence. This is desirable, since there are many possible pairs of words from the two categories and consequently the few of them that are actually observed are far more informative than the many pairs that are not. Mutual information on the other hand violates this desideratum. The original version of mutual information which is used in information theory (and sometimes called *average mutual information*) is completely symmetric, i.e., blind to a reversal of the set of pairs that occurred with its complement. The version of mutual information that is commonly used in computational linguistics[13] (e.g., [Calzolari and Bindi 1990; Church and Hanks 1990; Hindle 1990; Smadja 1993; Dagan *et al.* 1993; Daille 1994; Su *et al.* 1994; Chang *et al.* 1995; Fung 1995; Luk 1995; Zhou and Dapkus 1995; Haruno and Yamazaki 1996; Abraços and Lopes 1997; Buitelaar 1997]) is asymmetric, but its estimators are affected by the addition of non-observed pairs (columns in the table filled with zeros), since they are affected by the total size of the sample.

- Kendall's τ is a non-parametric statistical measure which has been extensively analyzed in the statistics literature. This in turn enables inference procedures such as the testing of hypotheses and the estimation of confidence intervals [Kendall 1975] without assuming particular parameterized distribution forms. This cannot be done for measures that are not based on statistical theory (including mutual information and the cosine measure), while many other measures make distributional assumptions, usually of normality, or assumptions of a linear relationship between the two variables

[13]The unweighted first term of the sum in the "standard", information-theoretic expression for mutual information, i.e., $\log \frac{f_{XY}}{f_X f_Y}$.

(e.g., the Pearson product moment correlation coefficient).

- Finally, Kendall's τ is a *robust* estimator of association. By virtue of its reliance on ranks and not on the actual frequencies, it is not affected by outliers and is much less influenced by observation noise caused by errors in the extraction process and in general by small random perturbations in the variables. In this way, τ also functions as an implicit smoother, addressing the sparse data problem which is manifested by the presence of many zeros in the table of positive evidence frequencies.

For the above reasons, I have selected Kendall's τ as the measure of similarity which is used to quantify indirect positive evidence for the semantic relatedness of two words. Despite the loss of information which takes place since only ranks (and not the actual values) are used, the measure seems to operate well in practice. Experimental comparisons with the other implemented measures (mutual information, the cosine measure, and Euclidean and average Euclidean distances) have confirmed the superiority of τ for this task.

2.3.1 An efficient algorithm for computing T^*

I noted earlier that a straightforward computation of T^* for two given variables X and Y examines all $\binom{n}{2} = \mathrm{O}(n^2)$ pairs of the n paired frequencies, where n is the number of words from category B which occur at least once with X, Y, or both. To compute the similarities between all pairings of m variables, we need time proportional to $\binom{m}{2} \cdot \mathrm{O}(n^2)$, or $\mathrm{O}(m^2 n^2)$. In order to increase the efficiency of the similarity computing process between any two variables X and Y, I sort the n pairs of observed frequencies (e.g., noun frequencies when computing similarities between adjectives) in ascending order according to each of the variables (corresponding to adjectives in this example). Let IX_i and IY_i be the ranks for a given word from category B in these two orderings. Then, ignoring for the moment ties within each variable, we observe that the total number of discordances between X and Y is given by

$$\sum_{i=1}^{n} \max(0,\ IY_i - IX_i) \tag{2.2}$$

or equivalently,

$$\sum_{i=1}^{n} \max(0,\ IX_i - IY_i) \tag{2.3}$$

We can derive expression (2.2) by conceptually sorting the paired frequencies in ascending order according to the frequencies for the variable X. If the pair of frequencies for a given word has rank l in this ordering, but its rank in the ordering for variable Y would be k, then the number of discordances this word participates in is $|k-l|$. Summing $|k-l|$ over all n words from category B results to twice the total number of discordances, as each discordance is counted twice for each of the two words from category B in it. Note that the sum of terms $k-l$ that are above zero is equal to the sum of the terms $k-l$ that are negative, resulting in expression (2.2). Expression (2.3) is derived in a similar manner by conceptually sorting the paired observations on the frequencies for variable Y.

Using expression (2.2) or (2.3) we can compute the total number of discordances for any two variables in linear time (not counting the cost of sorting). It is also possible to compute t_X, t_Y, and t_{XY} (the number of ties within variables X, Y, and both X and Y

simultaneously respectively) in a linear pass over the sorted lists of frequencies, and adjust the formulae given above for the total number of discordances for the more general case where ties are present. Then, the total number of concordances can also be computed in linear time by subtracting the sum of the total number of discordances and $t_X + t_Y - t_{XY}$ from $\binom{n}{2}$, the total number of pairs of words from category B that participate in T^*.

By carefully arranging the order of similarity computations across variables and storing the rankings of frequencies for each variable, we need to perform the sorting phase only m times for m variables, for a sorting time complexity of $\mathrm{O}(mn \log n)$. Thus, the total complexity of the similarity computation algorithm for all pairs of variables becomes

$$\mathrm{O}(mn \log n + \binom{m}{2} n) = \mathrm{O}(mn \log n + m^2 n)$$

which is both asymptotically and in practice significantly faster than the doubly quadratic time complexity of the straightforward computation of T^*.

2.4 Converting similarities to dissimilarities

Similarities are converted to dissimilarities because most clustering algorithms operate on the latter type of data rather than on the former. I use a simple linear transformation, which first normalizes similarities (which are assumed to be in the range -1 to $+1$, as produced by the Kendall's τ estimator) in the range 0 to 1, and then subtracts the resulting value from one; namely,

$$dissimilarity(X, Y) = 1 - \frac{similarity(X, Y) + 1}{2} \tag{2.4}$$

so that the produced dissimilarity value lies in the interval [0,1].

Although non-linear transformations based on raising the input similarity value to a power (typically 2 or $\frac{1}{2}$) can also be used, there doesn't seem to be a particular reason for doing so, and the relevant literature is generally in favor of a linear transformation. An experimental study by Lance and Williams [1979] on real data found that the linear transformation given by equation (2.4) was unequivocally the best when compared to other proposed transformations. Therefore, I have adopted this simple transformation for use by the system.

2.5 Identifying semantically unrelated words through negative examples

In addition to the positive examples that lead to the computation of similarity values according to Kendall's τ, I identified in Sections 2.1 and 2.2.2 syntagmatic relationships that indicate that two words are semantically unrelated. For the syntactic classes analyzed in this thesis, these relationships are expressed by adjective-adjective pairs for adjectives and noun-noun pairs for nouns. Using the finite state grammar described in the Appendix, the system extracts all such pairs from the corpus, forming a list of pairs with associated occurrence counts. A sample of this list for adjectives, drawn from the 1987 Wall Street Journal corpus, is shown in Table 2.4 on page 99.

Once this table of pairs and counts has been extracted from the corpus, we are faced with the task of deciding which of these cases correspond indeed to semantically unrelated pairs and which are spurious. We can use the frequency of occurrence of such pairs as a criterion for inclusion in the set of truly unrelated pairs; obviously, entries in the table with high frequencies are less likely to correspond to erroneously extracted pairs. The relative frequency of the pair compared to the individual frequencies of the two words can also be used, resulting in a relative rather than absolute threshold.

However, I have found that for this module placing a threshold does not seem to improve the overall performance. The collected examples are highly reliable, both because the underlying linguistic property is strongly indicative of semantic unrelatedness and because it is easy to extract such pairs with the finite-state grammar. Consequently, even a single example can lead to the conclusion of unrelatedness with high accuracy. On the other hand, the sparseness of the data means that raising the threshold results in the loss of valuable information. I do not therefore use a threshold on the frequency of the pair to decide if the two words are unrelated. The negative knowledge module of the grouping subsystem outputs an "unrelated" marker for all such pairs, with absolute certainty, although this happens only for a small fraction of all possible pairs (typically 2%–15%, depending on how frequent the words themselves are).

2.6 Combining evidence from different sources

When multiple modules produce independent suggestions for a particular decision, we need a method that combines them in a principled manner. This is the case for the grouping subsystem, where each possible pair of the words that are to be clustered receives a dissimilarity value from the indirect positive examples (Sections 2.3–2.4) and possibly a decision of semantic unrelatedness from direct negative examples (Section 2.5).

Because of the high reliability of the second source in this case, I have adopted a simple ad hoc algorithm for the combination of the two results. The input to the algorithm is $DSIM_{IN}(X, Y)$, a real-valued dissimilarity measure in the range 0 to 1 computed from the positive evidence as explained in Sections 2.3–2.4, and $COUNT(X, Y)$, a discrete count of the total number of examples indicating that the two words are semantically unrelated. The output of the algorithm is a new real-valued dissimilarity measure $DSIM_{OUT}(X, Y)$, which is defined as follows

if $COUNT(X, Y) = 0$, then

$$DSIM_{OUT}(X, Y) = DSIM_{IN}(X, Y)$$

else

$$DSIM_{OUT}(X, Y) = K$$

where K is a large constant relative to the maximum value of $DSIM_{IN}(X, Y)$ (currently, $K = 10$ while $\max(DSIM_{IN}(X, Y)) = 1$). In other words, the algorithm passes the input dissimilarity to the output unchanged when no negative examples have been seen; when even a single negative example has been extracted, the input dissimilarity is ignored and the dissimilarity between X and Y is set to the high value K. Additional negative examples do not affect the output, as argued in the previous section. Although $COUNT(X,Y)$ is an integer variable, it is effectively treated as a binary one (zero versus non-zero).

This simple algorithm operates quite well because of two reasons: First, the accuracy of the extraction process for the negative examples is high, and consequently the system rarely extracts a negative example by mistake. Second, cumulative negative examples are not in principle more informative than a single negative example; if we have seen two words in a pattern that implies unrelatedness we do not need to see this example more than once, except for the fact that multiple occurrences of the negative example may offset an extraction error.[14] These two reasons justify the implicit conversion of the negative evidence to a binary variable without significant loss of information. Since pairs of words participating in syntagmatic relationships that indicate incompatibility are not very frequent, the use of a threshold on the frequency of the negative examples results in a significant decrease of the negative evidence, without, as explained above, a counterbalancing increase in accuracy.

The above method works well because one of the knowledge sources is much more trustworthy, and because we already know the direction of the correlation of the output of the independent modules with the final decision. However, when we turn to other cases of combining evidence in the subsequent analysis of the word groups, we need a more general

[14] In an analysis of such pairs of adjectives, almost all of the few mistaken declarations of unrelatedness on the basis of the syntagmatic relationships of types 2 and 3 are due to polysemy, i.e., the two adjectives appear together with a given combination of senses, but they belong to the same group when a different sense is assigned to one or both of them. Taking the number of occurrences of the negative example into account does not help in this case unless the semantic group we want to form employs the most prevalent sense of the word, and that sense appears in the text far more frequently than the other senses of the word.

scheme for performing this task. Unlike the algorithm for the grouping of adjectives where only two sources of evidence come into play, we may have several independent variables that we want to combine.[15] In most cases, we won't have a linguistic analysis similar to the one given in the previous paragraph to guide us in determining which variable should be given more importance in the combined result. Some of the variables may not even be informative (i.e., have a very low correlation with the property we are studying), in which case they should be dropped from further consideration. Sometimes, we may believe that a variable is correlated with the property we are interested in but we may not know the sign (and not just the magnitude) of the correlation; for example, we may contemplate relative frequency or morphological productivity as an indicator of semantic markedness but not know if the more frequent or productive word is the marked or unmarked one.[16] Finally, even if we know which variables are the informative ones and the polarity of their correlation with the property, we still have to estimate the relative weight that each variable should be given in the overall measure. These issues necessitate the use of more general and formal learning methods; I discuss such methods as the need for them arises in Chapters 5–8.

[15]I refer to these variables as *independent* in the sense that they have been computed separately, and not in the statistical sense. Most likely, there will be statistical correlation between different predictors for the same property.

[16]Or more generally (and as in this example), we may suspect a particular direction for the correlation on the basis of linguistic theory, but we may want to confirm this conjecture empirically.

2.7 Clustering words on the basis of dissimilarities

Once dissimilarities have been computed between any two words that we are interested in grouping, I employ a *clustering* algorithm to automatically partition the words into groups of semantically related ones. Unlike supervised learning methods, clustering algorithms operate without access to preassigned class labels and without help from an external expert teacher. I found this approach more appropriate for the grouping task, since valid groups of words can be formed in many different ways, as evidenced from the disagreement between humans who are asked to perform the same task (see Sections 4.1.3 and 4.1.4). Furthermore, it doesn't seem likely that knowledge obtained from labeled samples (prior word groups) can be generalized to other samples involving different words. Of course, without guidance from a teacher, clustering methods can at best retrieve the structure present in the data, and may impose an artificial one if there are not significantly strong correlations between the clustered words. Nevertheless, I have found during the evaluation phase of the system (Sections 4.1.2–4.1.4) that the retrieved structure does not degenerate to artificial associations and is highly correlated with the target of groups of semantically related words.

In the next subsection I discuss the clustering method which I have implemented in the system. Then I turn to the important issues of how the number of clusters is determined and of the complexity of the task, and I close this section with a discussion of alternate clustering methods, comparing them to the method I use and explaining the advantages for doing so.

2.7.1 The exchange method

I have elected to use a non-hierarchical clustering algorithm, the *exchange method* [Späth 1985], which takes the number of the clusters to be produced as an input parameter.[17] The algorithm tries to produce a partition of the set of words as close as possible to the optimal one, i.e., the partition that minimizes the within-cluster dissimilarity (and consequently maximizes the between-clusters dissimilarity). This is accomplished by minimizing an *objective function* Φ which scores a partition $\mathcal{P}$. The objective function I use is

$$\Phi(\mathcal{P}) = \sum_{C \in \mathcal{P}} \left(\frac{1}{|C|} \sum_{X,Y \in C} d(X,Y) \right) \tag{2.5}$$

where $d(X,Y)$ is the dissimilarity between the words X and Y.

The algorithm starts by producing a random partition of the words, using the prespecified number of clusters. Then the Φ value of this partition is computed, and each word is considered in turn to determine the improvement in the objective function's value for every cluster where that word can be moved.[18] An optimized difference formula [Späth 1985] is employed to efficiently determine the change in Φ without recomputing its value from equation (2.5); note that just one computation of Φ from equation (2.5) requires time more than linear in the number of the words we are grouping. Once the move that results in the largest improvement of Φ is found, it is executed, and the algorithm proceeds to consider

[17] Actually, during the operation of the grouping system, clusterings are formed for different values of this parameter in a wide range of "reasonable" values. These clusterings are then available for inspection and evaluation.

[18] A word may not be moved from its current cluster if it is the sole member of the cluster and consequently the move would result in a partition with a smaller number of groups.

possible moves of the next word. The cycle continues repeatedly through the set of words until no more improvements to the value of Φ are possible. Note that in this way a word may be moved several times before its final group is determined.

This is a hill-climbing (gradient-descent) method and therefore is guaranteed to converge in finite time, but it may lead to a local minimum of Φ, inferior to the global minimum that corresponds to the optimal solution. To alleviate this problem, the partitioning algorithm is called repeatedly[19] with different random starting partitions and the best solution produced from these runs is kept.

2.7.2 Complexity of the task

As mentioned above, the exchange method is not guaranteed to find the global minimum of the objective function Φ. In fact, as with many practical optimization problems [Garey and Johnson 1979], computing the optimal solution is NP-complete. Brucker [1978] proves this claim of NP-completeness for a slightly different formulation of the objective function, but the small differences between his version and mine do not change the essence of the proof.

Nevertheless, if the problem's search space had been relatively small, then we could have computed the optimal partition by enumerating all possible solutions and keeping the best one. However, again as with many other practical optimization problems, the search space turns out to be intractably large. The number of possible partitions of n objects into m

[19] In the current implementation, 50 times for each value of the number of clusters parameter.

non-empty subsets with $m \leq n$ is equal to the corresponding Stirling number of the second kind [Knuth 1973; Kaufman and Rousseeuw 1990]:

$$\left\{ {n \atop m} \right\} = \mathrm{O}\left(\frac{m^n}{m!}\right)$$

which grows exponentially with n for all but trivial values of m. For example, for the smallest of the test sets of adjectives presented in Section 4.1.1 (Figure 4.1 on page 170), we have $n = 21$ and $m = 9$; the corresponding number of possible partitions, $\left\{ {21 \atop 9} \right\}$, is roughly $1.23 \cdot 10^{14}$.

2.7.3 Determining the number of clusters

Determining the "correct" number of clusters from the data alone is perhaps the hardest problem in cluster analysis in general [Kaufman and Rousseeuw 1990, p. 87]. Although an immediate suggestion would be to compare the values of the objective function for several different values of this parameter, it turns out that the objective function is monotonically decreasing as the number of clusters parameter increases, attaining the value 0 when each word is placed in a group by itself. This happens for any given set of dissimilarities $d(X, Y)$ between the words.[20] This behavior also seems to be a characteristic of all reasonable objective functions and not just of the one currently used.

[20] I have developed a formal inductive proof of this property, which is not included here in the interest of brevity.

Several heuristic methods have been proposed for the task. These methods, while relying on the objective function used during the clustering, include additional components so that they are more appropriate measures of the goodness of the clustering across different values of the number of clusters parameter. I have implemented one such method based on cluster *silhouettes* [Rousseeuw 1987; Kaufman and Rousseeuw 1990].

However, initial results do not seem to be satisfactory, as the heuristic measure seems to propose groupings with too few clusters. I have traced this underestimation of the proper number of clusters to the arbitrary handling of singleton clusters[21] with this method. As proposed in [Rousseeuw 1987; Kaufman and Rousseeuw 1990], the *average silhouette width* measure arbitrarily assigns a zero score to such clusters. While Kaufman and Rousseeuw [1990, p. 85] claim that this is a neutral value, I believe that it is actually an unfairly low value. Positive values of the silhouette width measure for a clustered object indicate that the object should have been placed in its assigned cluster, with increasing strength as the measure varies from 0 to 1. Negative values of the measure, on the other hand, indicate that another cluster seems to be more appropriate. In other words, a zero value is equivalent to saying that the assignment of the object to its current cluster has no support from the data and is on the edge between marginally appropriate and inappropriate assignments. Note that if we attempt to cluster the n vertices of a regular simplex in $n-1$ dimensions using Euclidean distance as the measure of dissimilarity, all formed clusters will receive an average silhouette width of 0 [Kaufman and Rousseeuw 1990, p. 98]. But this is an example where no structure in the data exists, so it is unfair for singleton clusters to receive the

[21] Clusters containing only one word.

same silhouette width value.

Because of this bias against singleton clusters, groupings that include many such clusters are scored lower than groupings with few singleton sets. But as the number of clusters increases, the average number of words per cluster decreases, and so does the proportion of non-singleton groups. Consequently, the average silhouette width and the *silhouette coefficient* that is used to rank clusterings decrease. However, singleton groups are not by definition bad groups, since some words may indeed not be semantically correlated with any of the other words we are clustering. I believe that the silhouettes method unfairly discriminates against such singleton groups, and I have considered modifications to the measure that will treat such groups in a more unbiased manner. One such modification is to assign a medium score value to such groups, for example 0.5, perhaps further depending on the number of clusters. I intend to experiment more with modifications of this type and reassess the results in the future, as the topic is interesting on its own right, beyond its application to the current clustering problem.

The failure of the current version of the silhouette coefficient measure to assign a correct value to the number of clusters parameter has not, however, been a major impediment in the work reported in this dissertation. The reason for this is partly that a significant range of values for this parameter is satisfactory for our purposes, since by varying the number of clusters we get different groupings at different levels of granularity. Furthermore, I have observed that a relatively good value for this parameter can be estimated from previous data sets of words that have been grouped, using the number of clusters that the human evaluators used in their models for these groups (see Sections 3.4 and 4.1.3). In

Adjective set	Number of words	Average number of clusters in human-constructed models	Ratio (average number of words per cluster)
1 (Figure 4.1)	21	8.56	2.453
2 (Figure 4.2)	43	15.33	2.805
3 (Figure 4.3)	62	23.12	2.682

Table 2.6: Relationship between word set size and average number of clusters in models constructed by humans.

Section 4.1.1 I present three sets of adjectives of different sizes and containing words of varying frequencies and degree of semantic specificity. Table 2.6 shows the size of these data sets and the average number of clusters employed by the human evaluators for each of them. I observe that the ratio of these two quantities (also shown in Table 2.6), i.e., the average number of words per cluster in the human models, remains relatively stable. This is an interesting observation that has psycholinguistic implications, notably that humans who are asked to perform classification tasks without explicit instructions on the level of detail tend to select approximately the same size of word groups independently of the total number of words. Therefore, a simple and yet reasonably accurate method for selecting the number of clusters is to divide the word set cardinality with approximately 2.4 to 2.8. The fourth set of adjectives presented in Section 4.1.1 (Figure 4.4) wasn't evaluated by humans because of its relatively large size (171 words). Yet, applying this method suggests 61 to 71 clusters, and the results of the clustering for this range of the number of clusters parameter seem satisfactory, indicating that we are viewing groups at more or less the proper level of detail. The heuristic method also gave good predictions for the second of the noun test sets presented in Section 4.2 and only slightly overestimated the average number of groups that the human evaluators used for the first test set of nouns.[22]

[22]It is possible that a different estimated number of words per cluster should be used in the prediction

2.7.4 Alternative clustering methods

A vast literature is associated with cluster analysis (e.g., [Anderberg 1973; Bock 1974; Hartigan 1975; Everitt 1980; Gordon 1981; Kohonen 1984; Romesburg 1984; Späth 1985; Kaufman and Rousseeuw 1990]), including numerous methods for performing the task. In this subsection, I review the most prominent methods in addition to the non-hierarchical exchange method that was described in Section 2.7.1, and I justify my choice of the latter.

The k-means method Another quite popular non-hierarchical clustering method is the *k-means* method [MacQueen 1967], which proceeds by randomly creating an initial partition, then computing the centroid of each cluster, and moving elements (i.e., words) to the cluster whose centroid they look more similar to. However, the method is not applicable in the case of this system since it requires the ability to compute centroids of groups, that is artificial elements that have average values for the features that define the dimensions of the element space. Subsequently, the distance between these artificial elements and the actual elements that are clustered must be computed. In our case, the element space is not explicitly defined through features, but rather implicitly on the basis of pairwise element dissimilarities. Consequently, neither the centroid of a cluster nor its distance from the words that are being grouped are defined.

A recently proposed robust alternative to the k-means method replaces the use of means with medians, resulting in the *k-medians* method [Massart *et al.* 1983; Klastorin 1985;

method for groupings of nouns than for groupings of adjectives; but more human-made groupings must be collected and analyzed before this question can be answered.

Kaufman and Rousseeuw 1987]. Using medians instead of means implies that the new center of the cluster (called now the *medoid*) will be one of the original elements. Consequently, both the medoids and their dissimilarities from other elements are well defined. The method offers two advantages that the exchange method does not, namely that a representative element is identified for each cluster and that the use of medians reduces the dependence on outliers. I did not consider this method when I initially adopted the exchange method, but I may do so in future work and compare the results. However, since I have found the performance of the exchange method satisfactory, I view this as an item of relatively low priority.

A point that is important when alternative non-hierarchical methods are considered concerns the mathematical restrictions placed on the dissimilarity measure. The method adopted for this system should not require that the measure $d(X, Y)$ that drives the clustering is a *distance metric*, i.e., that it satisfies the four metric properties

$$d(X,Y) \geq 0 \tag{2.6}$$

$$d(X,X) = 0 \tag{2.7}$$

$$d(X,Y) = d(Y,X) \tag{2.8}$$

$$d(X,Y) \leq d(X,Z) + d(Z,Y) \tag{2.9}$$

for any three elements X, Y, and Z. The dissimilarity metric that I use (cf. Sections 2.3 and 2.4) clearly satisfies the requirement of non-negative values (equation (2.6)) and the symmetry condition (equation (2.8)). In addition, since no explicit definition of the similarity of a word with itself was given in Section 2.3, we can assume that this similarity is perfect and the corresponding dissimilarity zero to satisfy equation (2.7).[23] Nevertheless, the triangle inequality that generally separates distance metrics from dissimilarity measures [Bock 1974, p. 25], i.e., equation (2.9), is not satisfied by the dissimilarity values I use. This violation is due to the incorporation of knowledge from multiple sources, in particular negative knowledge. Two words X and Y may have been observed in a syntagmatic relationship of type 2 or 3 that indicates their semantic unrelatedness, and therefore assigned a high dissimilarity value. But if neither X and Z nor Y and Z were observed in such a relationship, then it is quite possible that the sum of the dissimilarities computed for these pairs on the basis of positive evidence only is smaller than the dissimilarity between X and Y.

Hierarchical clustering methods Hierarchical clustering methods (e.g., [Florek *et al.* 1951; Sokal and Michener 1958; McQuitty 1960; Sokal and Sneath 1963; Ward 1963; Edwards and Cavalli-Sforza 1965; Lance and Williams 1966; Gower and Ross 1969; Hubert 1973]) differ from non-hierarchical ones in that they join or split groups sequentially, never revising previous decisions. They are the most popular ones[24] for a number of reasons:

- Since there is no revision of previous decisions, hierarchical algorithms run significantly faster than non-hierarchical methods. They are also significantly simpler and thus easier to implement.

---

[23] Such a choice for $d(X, X)$ is logical, and is actually implicitly made in the definition of the objective function (equation (2.5) on page 115) used in the clustering, if we don't explicitly disallow pairs involving the same word from entering the formula for the calculation of Φ.

[24] Romesburg [1984, pp. 2–3] retrieved several thousand abstracts of research papers from the late seventies and early eighties by searching for the key phrase "cluster analysis". In those papers, hierarchical approaches outnumbered non-hierarchical ones by more than 10 to 1. However, one might speculate that the increased availability of computing power during the last decade may have reduced the extent of this imbalance.

- The output is in the form of a tree (*dendrogram*) which lists all the various points where grouping decisions were made. Visual inspection of the tree makes possible an informal interpretation of the quality of the formed groups at various levels of the hierarchy.

- Since these methods produce a tree that incorporates partitions for all possible numbers of clusters, the problem of determining the value of this parameter does not come up immediately. Of course, if groups need to be explicitly formed, the problem reappears later, when we need to decide at which level to cut the tree.

- The inherent hierarchical structure means that smaller clusters are themselves grouped within other larger clusters. This is an advantage for many applications, including lexical classification tasks; in our specific application, for example, we would actually like to devise a semantic hierarchy of the words (similar to a thesaurus classification).

Non-hierarchical methods on the other hand offer greater precision in the construction of appropriate clusters, as the algorithm can reverse previous decisions that result in bad clusters when more words are considered. I chose a non-hierarchical method on the basis of this property, noting additionally that the performance penalty is not a prohibitive one. The relative low importance of performance costs is justified since we are dealing with a learning task that needs to be performed only once for a given set of words and domain, prior to the construction of the lexicon.

Among hierarchical algorithms, the *divisive* approach which proceeds by recursively partitioning the entire group into smaller and smaller clusters is expected to perform somewhat better than its alternative, the *agglomerative* approach that proceeds from singleton groups and repeatedly merges them [Kaufman and Rousseeuw 1990]. The reason for this is that methods in the former class have more data to look at at each decision step (the group that is to be split rather than isolated small subsets that are to be joined). Since efficient algorithms exist for the divisive approach that run almost as fast as the corresponding algorithms for the agglomerative approach [MacNaughton-Smith *et al.* 1964], I chose to implement and experiment with the former approach. An implementation of such a divisive method has been completed[25] and the analysis of the produced clusterings confirms the theoretically expected degradation of grouping quality when this hierarchical method is compared with the non-hierarchical exchange method.

Simulated annealing Simulated annealing is a clustering (and more generally, optimization) technique borrowed from statistical physics. The technique attempts to optimize (i.e., minimize) an objective function, as all non-hierarchical clustering methods do. However, to avoid being trapped in local minima, the algorithm allows the occasional choice of a move of an element from a cluster to another even if that move appears locally to make the objective function grow. Correspondence with physical reality is maintained by conceptually mapping the objective function to the role of energy, and the number of moves made so far to time. A third parameter, controlled by the clustering algorithm as a decreasing function of the number of moves (time), corresponds to temperature. Then, in analogy with the annealing of metals, we consider the various possible moves for each word in turn (see Section 2.7.1) and we immediately accept locally good moves (i.e., moves that reduce

[25] By Hsin Liu, a project student I supervised.

the value of the objective function, and conceptually move the system to a state of lower energy). But we also may accept locally bad moves that result in an *increase* ΔE of the objective function with probability

$$P(\Delta E) = A \cdot e^{-\frac{\Delta E}{kT}} \tag{2.10}$$

where k is an empirically chosen constant that corresponds to Boltzmann's constant in physics, T is the current (artificial) temperature, and A is a normalization constant that guarantees that the integral of the right side of equation (2.10) from 0 to $+\infty$ is 1. The temperature T is gradually decreased from an empirically specified initial value T_0 as a function of the number of moves t; the exact function that relates these two quantities can be specified in a variety of ways.

In other words, the simulated system goes eventually to states of low energy (i.e., partitions with low Φ values), but we hope that the occasional jolts that occur with the probability given by equation (2.10) will help the system move out of valleys containing local minima and plateaus of the objective function. Note that as time passes (i.e., the temperature decreases), the probability of taking a locally bad move also decreases with limit 0, which guarantees the convergence of the method in finite time with probability 1.[26]

Simulated annealing was initially greeted with much enthusiasm, and a lot of researchers studied optimization systems based on the basic Metropolis algorithm outlined above

---

[26]Which of course does not mean that the algorithm is guaranteed to converge in the worst case. However, I think that the difference between "convergence everywhere" and "convergence with probability 1" for stochastic processes [Papoulis 1984, pp. 189–190] is mainly of theoretical significance.

[Metropolis *et al.* 1953] and variations of it; see [Davis 1987; van Laarhoven 1988; Aarts and Korst 1989] for descriptions of several versions of simulated annealing algorithms. However, interest was considerably dampened when Geman and Geman proved that a simulated annealing-based optimization system using the Boltzmann (i.e., standard exponential) distribution for accepting a locally bad move (equation (2.10)) is guaranteed to find the globally optimal solution only if the rate of decrease of the temperature parameter is logarithmic in terms of the number of moves made [Geman and Geman 1984]. In other words, the artificial temperature T must vary with time at least as slowly as

$$T(t) = \frac{T_0}{\log(1+t)} \tag{2.11}$$

Such a decrease rate is of course equivalent to exponential time. To alleviate this obvious problem, alternative distributions have been proposed recently, e.g., the Cauchy distribution [Szu and Hartley 1987]. While the Cauchy distribution does not require an exponentially slow cooling rate analogous to equation (2.11), it suffers from other disadvantages originating from its non-finite variance.

To compare the performance of the grouping system when simulated annealing is used during the clustering phase, the Metropolis algorithm with a linear rate of decrease for the temperature was implemented, again by Hsin Liu in the project mentioned above. The analysis of the results showed essentially the same level of accuracy between the simulated annealing method and the exchange method that makes no locally bad choices but is restarted from a larger number of initial random configurations. The cost of the simulated annealing method in terms of running time was, however, significantly higher.

Fuzzy clustering techniques Another extension to clustering methods that is relevant to the word grouping task is the formation of "soft" clusters instead of the partitions described up to this point. Methods that produce such soft (or *fuzzy*) clusters do not assign each word (or, more generally, item that is to be clustered) to a unique group, but rather compute a *degree of membership* value, usually a real number between 0 and 1, for each word and group. The membership degrees of any particular word across all groups are frequently normalized to sum to 1, so that each can be interpreted as the probability that the word actually belongs to a given group. Several methods for fuzzy clustering exist, for example the *fuzzy k-means* method [Bezdek 1973; Dunn 1973] and the *MND2* method [Roubens 1978; Libert and Roubens 1982].

Fuzzy clustering methods seem especially useful for the grouping of words, since an alternative interpretation of the membership degree of a word in a particular group is as a measure of how well the word fits in the group.[27] This alternative interpretation is consistent with the knowledge that many words are polysemous, and that their different senses may well dictate conflicting choices for semantic groups and scales. For example, Table 2.7 shows some of the different senses of the adjective *hard* by the way of examples and corresponding semantic groups where this adjective could be assigned to depending on the sense that we consider most important. I note that no matter which choice is made, errors will occur as

[27]Under this alternative interpretation, the degrees of membership for each word are not expected to sum to 1 across groups and should not be interpreted as probabilities.

Semantic group (near synonyms and antonyms)	Example
easy, difficult	*hard problem*
dubious, questionable, flawless	*hard evidence*
kind, harsh, severe	*hard on the students*
mushy, soft, solid	*hard pillow*
strong, forceful	*hard push*
weak, strong, stable	*hard currency*
forgiving, resentful	*hard feelings*
unfeeling, compassionate	*hard heart*
arduous, strenuous	*hard work*
insistent, unyielding	*hard stare*
Other specialized senses: *hard drugs, hard liquor, hard water, hard blue, hard sound*	

Table 2.7: Some of the senses and corresponding partial groups of semantically related words for the adjective *hard*.

the lexeme *hard* corresponds to all the different senses of Table 2.7[28] and ideally should be placed in all groups, perhaps with different membership or association degrees.

The property of polysemy has consequently motivated researchers in computational linguistics to attempt the assignment of words to multiple senses or categories correlated with them, both in work in probabilistic sense disambiguation, and in work in word classification similar in some of its goals to the research described in this thesis [Pereira *et al.* 1993]. However, while I accept the need for a multiple classification of words as an ideal, I see two important points that I believe reduce its import in actual language learning systems. First, while several words are highly polysemous (for example, the COBUILD dictionary [Sinclair 1987b] gives 13 senses for *hard* as an adjective and WordNet [Miller *et al.* 1990] lists 41 senses for *run* as a verb), the phenomenon of polysemy is largely limited to the most frequent, core portion of the vocabulary. This observation has been made early on by Zipf

[28] As well as to other less frequent senses that are not mentioned in the table.

[1945], and was also corroborated in the empirical study I performed for the identification of semantic markedness [Hatzivassiloglou and McKeown 1995] (see Chapter 8). In other words, while some word types that represent many word tokens in the text have many senses, most word types represent few word tokens [Zipf 1945] and do so mostly unambiguously. For example, Leacock *et al.* [1993] found that each of the 100 most frequent nouns in the Brown corpus [Kučera and Francis 1967; Francis and Kučera 1982] has on average 5.15 senses listed in the WordNet lexical database [Miller *et al.* 1990]; only 8 of these nouns have a single possible sense in the database.[29] Yet, when we consider the entire collection of nouns appearing in WordNet irrespective of their frequency of actual use in the language, 86% of them have only one possible sense listed in WordNet.

Note that I claim that most words are unambiguous *in vacuum*, not because they are disambiguated according to contextual constraints. The correspondence of a single meaning to most words is reinforced when we are considering words from a particular domain rather than the language at large. Many polysemous words are polysemous because they have several senses in different domains, but only one sense in each domain; even when the word is polysemous in a given domain, one of the senses may prevail over the others to the point that, within the domain, the other senses can be ignored.

During the automatic learning of semantic classes we are more interested in the rarer (and less ambiguous) words, for which overlap between groups is not so much a concern. These words constitute the majority of the words we are dealing with (since we classify word types) and are the hardest to classify manually (since they are rare, specialized,

[29] And that within WordNet's coverage of senses, which is not necessarily complete.

and frequently domain-dependent). If the automatic method does not support overlapping groups, a human post-editor can correct the results for the relatively few frequent words that require such overlapping.

The point raised above justifies why I expect overlap between semantic groups to be of small value for a lexical acquisition system. However, the above discussion leaves the impression that, although the benefit from multiple classification may be small, such a classification technique nevertheless offers an improvement over non-overlapping classification methods for some of the words. This impression is mistaken, however, as it does not take into account the price paid for this generality in classification. A non-overlapping method that needs to assign n words to $m \leq n$ groups has to estimate an $n \times m$ matrix of zeros and ones constrained so that each row has exactly one "one"; equivalently, the method has to determine integer values in the range $\{1, \ldots, m\}$ for $n - m$ variables. On the other hand, a fuzzy clustering method has to estimate an $n \times m$ matrix of real numbers, possibly under the constraint that each row sums to one[30]; equivalently, the fuzzy method needs to estimate $n \cdot (m-1)$ real variables. I observe that the non-overlapping method has about m times less parameters that need to be estimated than the more general method. Since more parameters have to be estimated from the same amount of data, the non-overlapping method will perform considerably better than the more general method as long as its assumptions are not violated to a high extent. I explained above why I believe that these violations will be limited, and consequently I will not use a fuzzy clustering method for the word grouping task.

[30]This constraint applies if the first, probabilistic interpretation is used.

2.8 Summary

I have analyzed linguistic structural relationships between open-class words that provide either indirect positive or direct negative learning examples which help establish whether two words are semantically related (i.e., part of the same group or scale). I have identified the forms that these syntagmatic relationships take at the syntactic level, and discussed which of these forms are reasonably easy to extract and most informative for the task of word grouping. Then, I presented the similarity metric I use to measure the semantic association between any two words, and I discussed how negative evidence on the unrelatedness of two words is processed and combined with the positive co-occurrence data. Finally, I presented the non-hierarchical clustering algorithm that actually places the words into groups, and contrasted it with alternatives such as hierarchical and fuzzy clustering methods.

I now proceed, in the next chapter, with the description of the evaluation methods that score the automatically produced groupings of words. In Chapter 4 I return to the actual application of the algorithms discussed in the present chapter, and show results obtained for several collections of adjectives and nouns.

Chapter 3

Evaluation

A formal, quantitative evaluation methodology is necessary if we intend to be able to assess the quality of the system's output and compare between different versions of it (for example, when an alternative algorithm is used at some stage, or when a particular linguistic module is present or absent). I devote this chapter to a discussion of an evaluation approach appropriate for the four tasks this thesis addresses.

While this evaluation methodology is used throughout the four stages of the implemented system, it is discussed here because of its special relevance to the grouping subsystem: that subsystem's decisions for each word produce structured and interdependent results, for which no quantitative evaluation procedure has been proposed before. To numerically qualify the success of the system, I have adapted widely used metrics from information retrieval, generalizing them to multiple reference models and providing a procedure that converts structured results (such as the partitions produced from the word grouping component of the system) to a form suitable for the application of these metrics.

There are two major issues that need to be resolved during the design of an evaluation method. First, we need to decide whether the system's output will be evaluated solely on the basis of measurable properties it possesses, or whether an explicit correct answer should be provided for comparison with the system's output. I take up this question in the first section, where I argue that for the system described in this dissertation the availability of an objective standard is highly desirable. Second, the procedure that quantifies the discrepancies of the system's output from the ideal one should be specified. To address this issue, I show how solutions to the problems addressed in the thesis can be converted to lists of "yes"–"no" answers, so that evaluation methods developed in information retrieval for such lists can be employed. I then cover the related topic of how uncertainty about the correct solution can be handled by simultaneously using multiple models during evaluation. I also discuss issues related to meta-evaluation, i.e., to determining what the evaluation scores actually mean. I conclude the chapter by expounding on the meta-evaluation issues and showing procedures that transform the evaluation metrics, correcting for non-linearities in them due to the structure inherent in the evaluated solutions.

3.1 Defining the evaluation standard

The methods that have been proposed for selecting an evaluation standard for a learning task can be separated into two broad categories: *internal* and *external*. Internal evaluation methods use only data that was available during the learning phase, while external methods rely on information inaccessible to the learning component. The use of the *cophenetic matrix* and the associated *cophenetic correlation coefficient* for the evaluation of hierarchical groupings [Romesburg 1984] is an example of an internal method that scores trees according to the dissimilarities between the elements assigned to the various clusters; the same dissimilarities were available to and presumably have been used by the clustering algorithm that produced the tree. Evaluation of probabilistic language models on the basis of entropy (or equivalently, perplexity [Bahl *et al.* 1977]) [Brown *et al.* 1992b; Brown *et al.* 1993], which has recently been applied to evaluating parse trees as well [Lyon and Brown 1997], is another example of an internal evaluation method. While internal evaluation methods offer the simplicity of requiring no more data than was available for the learning phase, they suffer from two important deficiencies: They can only evaluate the performance of the system indirectly, within the confinements of the particular model adopted for the evaluation task; e.g., perplexity measures how well a given probabilistic model matches experimental data using a particular criterion for "matching", which may or may not correlate well with the accuracy of predictions based on the model. And, such methods ultimately depend on the initial data supplied to the component that produced the results under evaluation; e.g., the cophenetic correlation coefficient for evaluating hierarchical groupings assumes that the initial similarities or dissimilarities between the items that are clustered are the correct ones, as probabilistic language models do for their estimates of probabilities. In other words, the results of an internal evaluation method would be inaccurate if either the model employed or the data on which the system operates were found to be incorrect. In particular, when the system is itself responsible for the collection and/or synthesis of data on which later subsystems operate, as is the case with the system described in this thesis and with most learning systems, internal evaluation schemes fail to connect their measure of performance to actual performance in the real world.

Although internal evaluation methods can be useful as interim indicators during system development, the points mentioned above necessitate the adoption of an external evaluation standard for the final assessment of the quality of the produced output. Such standards can be supplied completely independently of the system, in the form of *model solutions* that are identical or equivalent in structure to the solutions produced by the system and have been generated by human experts (or other trusted systems). This evaluation approach assumes that the experts can be trusted to generate correct solutions, and that the system's performance can never exceed the experts'. If the latter were to happen the system would actually be *penalized*, as its performance is measured by how much its output differs from the models constructed by the humans.

The construction of such explicit models is expensive since it requires the co-operation of human experts (or, in any case, reasonable substitutes for them), and this methodology may not even be applicable to some tasks (e.g., machine translation), since it requires that the experts be able to construct correct solutions, and that a method for measuring differences across solutions exist. Fortunately, for the problems addressed by the thesis (i.e., partitioning words in groups of semantically related ones, deciding which groups are scales, separating antonyms from synonyms, and determining the semantic orientation of words) experts are usually able to form model solutions.[1] Furthermore, a procedure for measuring the discrepancy between two solutions has been developed for the problems covered in the thesis, and is described in the next two sections.

It should be noted here that a less expensive external evaluation method is to let the

[1]Although, as I discuss in Section 3.4, these solutions generally vary somewhat from expert to expert.

human expert inspect the solution produced by the system and grade its quality. In this way, the expert does not need to explicitly construct a model; rather, the model is implicitly constructed "on the fly" as the evaluation of the system's output is made. Although such an evaluation method still requires access to experts and additional work for the evaluation, it is attractive because checking an already formulated solution can be significantly easier than producing one from scratch. For example, it is relatively easy to check if the multiplication of all the numbers from a given collection of prime numbers results in a given product, but much harder to identify the prime factors of a given number.[2] Consequently, this evaluation methodology has been employed for many systems, including ones in computational linguistics. When there is only one solution to the problem that is evaluated[3] and this solution can be defined objectively, the method is equivalent to providing an independent model solution. However, many complex problems do not have a fully defined solution; rather, experts can construct "good" solutions and judge the quality of others, but not rationalize the process of solving particular instances of the problem. This is especially true for language problems, and holds for all the problems addressed in the thesis. In such cases, judging the system's output without an explicit model may result in the introduction of bias in the evaluation, with the experts over- or under-estimating the quality of the system's output depending on prejudices, human error, social factors such as the desire to please or discourage the system's developer, etc. I have therefore decided to adopt independently constructed models for evaluating the various stages of the system.

[2] In general, it is a defining characteristic of all NP-complete problems that it is easy to check a proposed solution although finding one is hard [Garey and Johnson 1979].

[3] Or a small number of solutions, assuming appropriate extensions to the method that uses the predefined solution(s).

Once it is decided that experts will be employed in the evaluation, the question of how these experts will be selected arises. Ideally, I would like to have professional lexicographers evaluate the system, but this policy has drawbacks in terms of the cost of the evaluation and access to evaluators. Therefore, I use instead people with a modicum of linguistics knowledge, more specifically faculty, research staff, and graduate students in computational linguistics here at Columbia University. As I explained in Section 1.9.3, I did not further restrict the experts to native speakers of English, both because that would severely limit their number and because I do not feel that knowledge of English as first language is necessary for constructing adequate models for the problems handled by the present system. The experts are given the same input as the system (list of words to be grouped, groups to be further separated, etc.), explicit instructions on the form that the output should have (e.g., non-overlapping partitions), and some description of the domain.[4] To avoid introducing bias in the evaluation, I do not provide low-level instructions on the specific desiderata of the models (such as the number of clusters to be employed in the grouping problem); rather, general instructions and examples are given to the experts. Figure 3.1 shows instructions of this type, which were actually used for the construction of model partitions into semantic groups for the adjective sets that are presented in Chapter 4.

[4]It would be impractical to provide them with a copy of the (very large) corpus used by the system.

Instructions:

You are requested to separate the adjectives appearing below into groups, so that adjectives with similar meaning (describing the same property) go to the same group and adjectives with dissimilar meaning go to different groups. The domain where the adjectives are used is the domain of financial reports about the stock market. You are free to select the appropriate number of groups, but the groups should not overlap.

Example:

4 adjectives are to be grouped: cold fast hot slow.

Appropriate grouping : 1. cold hot
2. fast slow

Adjectives to be grouped (given in alphabetical order):

⋮

The list of adjectives follows. Note that the four adjectives used in the example do not appear in the list.

Figure 3.1: Instructions given to human experts for the task of partitioning a set of words (adjectives) into semantically related groups.

3.2 Converting solutions to lists of "yes"–"no" answers

The problem of comparing two lists of "yes'–"no" answers for a common set of questions has been well studied in the information retrieval (IR) literature [Salton and McGill 1983; Frakes and Baeza-Yates 1992]. Consequently, if a method that converts other types of solutions to this type of lists is available and this method preserves the information in the original solutions, the information retrieval evaluation measures for such lists become applicable. I discuss in this section how the solutions produced by the system and the corresponding models for them that are constructed by the experts can be converted to such "yes"–"no" lists.

The system's output at the end of each of the four stages described in Section 1.9 can be characterized as having one of the following general structural types: a partition (of the entire set of words at the grouping stage or of a particular scale in two subgroups of opposite polarities), or a list of binary decisions (on whether each semantic group is actually a scale or on which side of each scale is the positive one). The partition of a scale into subscales is in fact equivalent to a binary decision because only two subgroups are employed. A partial ordering of scalar elements according to semantic strength, which I consider as a future extension to the thesis, is a third kind of structured result that can also be evaluated with the methods developed in this chapter.

Series of binary decisions are of course already in the form that we seek, i.e., lists of "yes"–"no" answers. More generally, if the system is allowed to be indecisive about some of the cases that it is unsure of, a three-way answer is possible: "yes", "no", and "don't know". Three-way decisions of this form can be evaluated by first measuring the percentage of cases where the system actually supplies an answer, and then using the techniques of the next two sections for evaluating the system's performance on these cases. As long as the system is more or less correct in identifying the cases where it has difficulty finding the proper response, a tradeoff between the measures of decisiveness and accuracy is expected.

Partitions of objects (words) into groups can be evaluated by defining a new binary relation between words, "is associated with", and assigning a "yes" value for this relation to all word pairs that are placed in the same group and a "no" value to all other word pairs. Similarly, partial orderings of words according to semantic strength can be converted to binary decisions by defining a binary relation "precedes in the ordering", with the value "yes" if the first word is considered less strong than the second in the solution, and "no" otherwise. As in the case of binary decisions, we can separate cases where the system has no opinion by measuring the percentage of these cases and then redefining the binary ordering relation only on the word pairs for which the system actually produced an answer.

These conversions fully preserve the information content of the initial structured solutions, since we can resynthesize the solutions from the corresponding sets of "yes"–"no" answers. Note, however, that the initial structure imposes constraints on which combinations of "yes"–"no" answers may represent solutions; not all such combinations have a corresponding structured solution. For example, in the case of partitions of words, if the association relation takes the value "yes" for the word pairs (X,Y) and (Y,Z), it must also take the value "yes" for the word pair (X,Z). These structural constraints have implications on the density of good solutions and consequently on the meaning of the evaluation scores. I return to this point in Sections 3.5.1 and 3.6.

3.3 Comparing lists of "yes"–"no" answers

Once we have both the solution being evaluated and the model that provides the standard in the form of lists of "yes"–"no" answers (either directly because the output of the system

	Answer should be Yes	Answer should be No
The system says Yes	a	b
The system says No	c	d

Table 3.1: Generic form of a 2×2 contingency table, contrasting the system's and model's answers to a binary classification problem.

or expert is in this form, or after the conversion procedures of the previous section have been applied), we can compare them quantitatively on the basis of well-developed information retrieval measures. First, I observe that as long as all questions to which answers are provided are considered of the same importance[5], the degree of similarity between the two lists can be captured in a 2×2 *contingency table* [Swets 1969]. Such a table, depicted in Table 3.1, separates the pairs of answers in the two lists into four categories, corresponding to the four possible combinations of the "yes"–"no" choices made in the two solutions that are contrasted.

A high degree of agreement between the two lists of decisions corresponds to high values for the quantities a ("yes"–"yes" matches) and d ("no"–"no" matches) of Table 3.1, and can be immediately measured by a quantity such as the *simple matching coefficient* (defined as $\frac{a+d}{a+b+c+d}$) [Zubin 1938; Dumas 1955; Romesburg 1984]. However, the discrepancies between the two solutions are of two types: cases where the evaluated system said "yes" but the correct answer was "no" (a false positive, or, using statistical terminology, a type II error), and cases where the evaluated system said "no" but the correct answer was "yes" (a false negative, or type I error). Depending on the particular problem to which solutions are

[5] A natural assumption in the absence of specific information about any particular question, and one that I will make during the evaluation process.

evaluated, one or the other type of errors may be more important. Another way to view these different evaluation aspects of the system's performance is to measure the percentage of correct "yes" answers given by the system among the total set of "yes" answers according to the model (*recall*, which will be high when there are few false negatives), and the percentage of correct "yes" answers given by the system among the total set of "yes" answers given by the system (*precision*, which will be high when there are few false positives). These two measures are widely used in information retrieval [Salton and McGill 1983; Frakes and Baeza-Yates 1992]. Using the designations of Table 3.1, they are formally defined as

$$\mathit{Recall} = \frac{a}{a+c} \cdot 100\% \tag{3.1}$$

$$\mathit{Precision} = \frac{a}{a+b} \cdot 100\% \tag{3.2}$$

An alternative measure used in information retrieval for measuring the rate of false positives is *fallout*, which is defined as

$$\mathit{Fallout} = \frac{b}{b+d} \cdot 100\% \tag{3.3}$$

i.e., as the percentage of wrong "yes" answers (false positives) compared to the total number of "no" answers in the model. This measure has analytical advantages over precision[6], and has been empirically observed to be a more accurate measure of the error rate due to false positives than precision [Lewis and Tong 1992]. Note that unlike recall and precision, good

[6] It doesn't become undefined unless there are no negative answers in the model, an unlikely situation. In contrast, precision becomes undefined if the system does not produce any positive answers.

values for fallout are values close to 0.[7]

Recall and precision (or recall and fallout) are frequently in an inverse relationship when one or more system parameters can be controlled to produce different solutions [Frakes and Baeza-Yates 1992; Chinchor 1992; Harman 1993]. Selecting values for these parameters near one end of their range results in high recall and low precision (or high recall and high fallout), while values near the other end result in low recall and high precision (equivalently, low recall and low fallout). Frequently, a *precision-recall graph* is used to show the tradeoff between precision and recall as some other system parameter varies; see for example Figure 2 in [Harman 1993, p. 65]. Generally, the more conservative the system is in producing positive answers, the lower its recall and fallout and (usually) the highest its precision. Taking this to extremes, a system that always says "yes" will have perfect (100%) recall and at the same time 100% fallout; a system that always says "no" will have 0% recall but also 0% fallout. A corresponding situation is realized in the grouping task where the controlling parameter is the number of clusters employed in the grouping. Placing all words in one cluster is equivalent to always saying "yes", while placing each word in its own cluster is equivalent to always saying "no". Intermediate values of the number of clusters parameter correspond to different tradeoffs between recall and fallout/precision.

Since recall and precision are inversely related, it makes sense to combine them in a single measure, as such a combined measure allows comparisons and rankings of different versions of the system using a single number. An immediate suggestion for the combination

[7]Another measure used in information retrieval, *overgeneration* [Chinchor 1992], is in all evaluation tasks addressed in the thesis always equal to $(100 - \mathit{precision})\%$.

of recall and precision would be to take their arithmetic average. But simply averaging the two measures does not indicate how far apart their values are. It can be argued that a desirable property for a solution is for it to be balanced in terms of the two types of errors: a system with 60% recall and 40% precision is more balanced and thus better than a system with 80% recall and 20% precision. A measure that allows an incorporation of balance considerations in the combined score is the *F-measure*, proposed in information retrieval [van Rijsbergen 1979] and used in (D)ARPA[8] evaluations of natural language systems [Sundheim 1992; Chinchor 1992; Lewis and Tong 1992]. This measure is defined as

$$\textit{F-measure} = \frac{(\beta^2 + 1) \cdot \textit{Precision} \cdot \textit{Recall}}{\beta^2 \cdot \textit{Precision} + \textit{Recall}} \tag{3.4}$$

where β is the weight of recall relative to precision.[9] Values for β that have been used in information retrieval include $\beta = 0.5$ (precision twice as important as recall), $\beta = 2.0$ (recall twice as important as precision), and $\beta = 1.0$ (recall and precision equally important). I will use the latter value of β in the evaluation of the system, as there is no particular reason to favor one aspect of the evaluation over the other. Then equation (3.4) takes the simplified form

$$\textit{F-measure} = \frac{2 \cdot \textit{Precision} \cdot \textit{Recall}}{\textit{Precision} + \textit{Recall}} \tag{3.5}$$

Independently of the value of β actually selected, it can be verified from equation (3.4) that the F-measure is always between the values of recall and precision, and it is larger

[8]The U.S. Defense Department's Advanced Research Projects Agency, a major funder of work in natural language processing in the United States.

[9]Note that for $\beta = 0$ the F-measure is equal to precision, while for $\beta \rightarrow +\infty$ the F-measure becomes equal to recall.

when these values are closer together, as desired.

3.4 Using multiple models

Due to the inexact definition of the tasks, the different interpretations of the words and different levels of granularity used by each evaluator, and even human error, the models produced by the humans for the same problem instance are not identical but rather contain considerable differences. Therefore, we end up having a *set* of different model solutions to the problem. Since our a priori assumptions place equal trust in each evaluator, we need a method that reconciles these differences and produces a revised single model. Alternatively, we can extend the comparison procedures of the previous two sections so that the system's output can be graded against all the human-prepared models, which are taken *collectively* to represent the "true" correct solution.

The prevalent approach when significant differences exist between the models is the first one, namely to extrapolate a single "correct" model from the set of disagreeing but equally important models. For the particular case of models consisting of lists of binary decisions, which are used for the evaluation of the components of this system, Gale *et al.* [1992a] have proposed a method for deriving a single model that agrees most with the original independent models. The method replaces the set of answers proposed for each decision by the multiple models with the answer that appears in the *majority* of the models for that decision. It can be shown that this procedure is optimal for the construction of a single model, in the sense that if we count the number of original models that agree with the constructed model at each decision point and sum these counts across all decision points, no other single model will have a higher value for this agreement measure.

However, the majority method described above results in loss of information, as the final single model fails to capture the degree of disagreement between the original models. Whether six or nine out of ten models agree on a particular question, the result would be identical for that decision point. If we assume that there is a single correct answer for each question (i.e., a single ideal model) of which the models at our disposal are but imperfect realizations, then the majority method is justified in attempting to reconstruct the ideal model as closely as possible. But in most problem solving tasks with imprecisely defined goals, it is more reasonable to assume that several possible solutions exist. In this case, a method that uses all the expert-prepared models simultaneously has the advantage that it directly incorporates their differences in the evaluation task. If the experts disagree a lot for a particular question, then our confidence in their ability to produce the correct answer for that question should be reduced. Consequently, the weight given to this question during the evaluation should also be reduced relatively to the weight given to questions with higher agreement among the experts.

Since the tasks that the system described here attempts to accomplish are certainly not fully well-defined (in the mathematical sense), and since I have observed considerable differences between the models already constructed by humans for the grouping task (see Section 4.1.3), I have decided to use an evaluation method that compares the system's output[10] to the entire set of model solutions. To this end, I have extended the information retrieval metrics discussed in the previous section (precision, recall, fallout, and F-measure) and provided generalized versions that compare one list of "yes"–"no" answers to a collection of models of "yes"–"no" answers. These extensions are based on the replacement of counts in the contingency table (Table 3.1) by sums of appropriate *fractional positive* or *negative* values.[11]

When only a single model is used during evaluation, the answer to each question should be either unequivocally positive ("yes") or unequivocally negative ("no"). Accordingly, the appropriate cell of the contingency table, in the row specified by the system's response and the column specified by the model's decision, is incremented by one. When multiple models are available and these models disagree, we cannot speak of completely positive or completely negative ideal answers; neither can we speak of "correct" or "incorrect" responses of the system, since the system's output will be correct according to some of the models and incorrect according to others. Rather, I define the *fractional positive value* of each particular question (i.e., decision point in the list) as the ratio of the number of models that contain a positive answer for this question to the total number of models. Similarly, the *fractional negative value* is defined for each question as the ratio of the number of models that propose a negative answer for that particular question to the total number of models. From the definitions, the properties below follow immediately:

[10] Or in general, any single solution, even one prepared by a human.

[11] In previous work [Hatzivassiloglou and McKeown 1993] I have referred to these values as *fractional correctness* and *incorrectness* values respectively, associating the positive response with correctness, since in the context of associations a positive answer means that the respective association should be correct (true). I revise this terminology since the system can be correct (i.e., in agreement with the models) even when it produces a negative response.

- Both the positive and negative values are between 0 and 1 for each response.
- The positive and negative values for each response sum to 1.
- A positive value of 1 (and thus a negative value of 0) is attained if and only if all the models propose a "yes" answer for that particular question. Similarly, a positive value of 0 (and thus a negative value of 1) is attained if and only if all the models propose a "no" answer for that particular question.
- A larger number of models offers higher resolution in both the positive and negative values. More specifically, with n models the minimum difference in such values is $\frac{1}{n}$.

The particular positive or negative values of each decision point in the list depend on the models only, and do not change as we evaluate different systems on the same models. These values essentially represent the popularity of each of the two possible answers to each question. They also represent the agreement present in the models for each question, with values closer to 0.5 indicating less agreement. For example, among the 9 models used during the evaluation of groupings for the set of adjectives of Figure 4.1 (Sections 4.1.1 and 4.1.3), the association between *foreign* and *international* received a positive value of 1, since all the human judges placed these two adjectives in the same group, while the association between *legal* and *severe* received a positive value of 0. The pair (*economic, political*) on the other hand received a positive value of 0.67, since two thirds of the judges placed the two adjectives in the same group.

Having defined the positive and negative value for each question, I revise the process for constructing the contingency table as follows: When the system says "yes" to a particular question, we increment *both* cells of the first row of Table 3.1 instead of just one; the first cell is incremented by the positive value of that question, and the second by the negative value of the question. When the system's response is "no", the same procedure is followed but using the two cells of the second row of Table 3.1. In this way, we replace the counts a and c with sums of positive values, and the counts b and d with sums of negative values. Formally, we have

$$a = \sum_{\text{question } i \text{ where the system says "yes"}} \text{positive value of question } i$$

$$b = \sum_{\text{question } i \text{ where the system says "yes"}} \text{negative value of question } i$$

$$c = \sum_{\text{question } i \text{ where the system says "no"}} \text{positive value of question } i$$

$$d = \sum_{\text{question } i \text{ where the system says "no"}} \text{negative value of question } i$$

Then the formulas for recall, precision, fallout, and F-measure which were given in Section 3.3 are applied to the redefined quantities a, b, c, and d, without change.

The generalized evaluation metrics described above represent a natural extension to the original ones. The new metrics retain the important properties of their simpler counterparts, and at the same time incorporate the disagreements among the models. In particular, it can be easily verified that the new metrics are identical to their original versions when there is only one model solution or when there is perfect agreement among the model solutions. I am interested in pursuing the exploration of the properties of these generalized metrics more in the future, and in comparing them with the majority method, possibly using simulation techniques under specific distributional assumptions for the system's responses and the models' specifications.

3.5 Meta-evaluation

While concern for a rigorous evaluation of natural language processing systems seems to be on the rise, in part due to emphasis placed on evaluation by (D)ARPA in the past five years, the problem of assessing the meaning of the evaluation scores is usually superficially addressed or not at all. For example, there has been increasing concern that the scoring methods used for evaluating the quality of parsers' output are producing values which seem extremely good. Researchers constructing parsers based on either the inside-outside algorithm (e.g., [Pereira and Schabes 1992; Magerman 1995]) or the transformational learning-by-example approach [Brill 1993] have reported scores in the 85–95% range, using the "crossing parentheses" measure proposed in [Black *et al.* 1991] and subsequently adopted by the parsing community. Yet the parse trees produced do not seem to be so satisfactory; the blame for this inflation of the scores may be attributable to a too lenient scoring technique, which essentially considers a tree fragment correct when it is a part of (although not exactly matching) the corresponding fragment in the model. This has motivated a more recent, information theory-based approach to the evaluation of parse trees [Lyon and Brown 1997]. For other tasks, such as the assignment of part-of-speech assignment tags to free text, the comparison techniques are sound, but very high levels of performance (around 90%) can be obtained by a system that simply assigns to each word the most frequent valid part-of-speech tag for it, irrespective of context. Clearly this makes the assessment of the significance of an improvement over the base line of the above simple algorithm much harder. In fact, even a program that assigns randomly legal tags to each word without any distributional knowledge will be accurate about 75% of the time, because many words have just one legal tag.

Yet the density of the solution space in "good" and "near-good" solutions is a factor that greatly influences the likelihood of obtaining a good score, and consequently should be extensively studied during evaluation. Even for the relatively conceptually simple general class of problems of binary classification, the a priori probabilities of seeing each of the two classes play a major role in determining how good a proposed method will look, if evaluated on measures such as accuracy of classification, recall, and precision. If two classifiers are compared on the same task, then these measures can provide an answer as to which performs the task better, although assessing the significance of any difference in the scores is difficult if we do not know how hard it is to improve performance beyond a certain point. But comparisons across tasks is not valid, because the percentage scores produced by the evaluation are not absolute. If one classifier operates with total accuracy of 70% on a problem where the most frequent class appears 60% of the time and the second classifier achieves 98% accuracy on a problem where the most frequent class appears 99% of the time, the first classifier is clearly doing a better job: at the very least, it offers improved performance over the simple method of always selecting the more frequent class. Consequently, an effort at providing at least an approximate measure of how difficult or easy it is to find good solutions to the problem and of how hard it is to progress in the evaluation score should accompany every objective evaluation effort.[12]

In the remainder of this section I address two issues of meta-evaluation in the context of the present system: structural constraints in the grouping solutions and the evaluation method that affect the density of the solution space in good solutions as measured by the evaluation metrics of the previous sections; and methods for empirically determining upper and lower bounds for the evaluation metrics.

3.5.1 Constraints relevant to the evaluation process

Estimating the density of the solution space in good solutions is very hard, especially as it involves a precise definition of the assumptions under which the solution space is searched: very different results will be obtained when a method with considerable relevant prior knowledge is used than with a blind search method. Since we are mostly interested in methods that take advantage of prior knowledge but we are unable to state the effects of such knowledge formally, the task of describing the density of the solution space quantitatively

[12]For the reader's amusement, I include the following excerpt from a negative review I received for a paper submitted in 1994: "... *78% performance in the case of* ... (the worst case for the evaluated system) *for a binary classification task doesn't seem satisfactory, which shows that you didn't look at the right kinds of data.*" The two classes were equiprobable (resulting in baseline classification accuracy of 50%) and no other algorithm for solving the problem had been reported previously.

seems intractable. However, we can at least try to identify structural constraints in the solution that affect the evaluation process, and constraints in the evaluation process itself. The process of converting a partition or ordering to a list of elementary binary decisions imposes such structural constraints, namely constraints of consistency among the individual binary decisions to which each higher level decision (group or ordering) is decomposed. When clusterings are evaluated, I observe that, first, the binary relation induced from the clustering is transitive, since we view the clustering as the collection of equivalence classes of the "is associated with" relation. If the system misses an association between two words X and Y that should have been placed in the same group, then it also has to miss all correct associations between the other words of that group and at least one of X or Y. Conversely, if the system erroneously identifies X and Y as semantically related, then it will also identify all words related to X as related to all words related to Y. Because each pair of words in a cluster is considered an observed association, the relationship between the number of associations produced by a cluster and the number of words in that cluster is not linear (a cluster with k words will produce $\binom{k}{2} = \mathrm{O}(k^2)$ associations). Consequently, the penalty in recall and precision for the wrong placement of a word is not constant (as it would be if the binary decisions were independent), but increases linearly with the size of the group where the error was made (the group to which the word was erroneously removed from for false negatives (recall) or attached to for false positives (precision)). Second, the system is constrained to produce a partition, meaning that errors will in most cases come in pairs: if a word is erroneously assigned to a group, we will have a number of false positives for that group (unless the group is a singleton) *and* a number of false negatives for the group that the word should have been assigned to (unless that group is a singleton). As

Model	Partition
1. A B	1. A B
2. C D	2. C D
3. E F G H I	3. E F G
	4. H I

Figure 3.2: A hypothetical model (left) where a small perturbation (right) leads to a recall score of 50%.

an example of the non-linearity of the evaluation measures when viewed as functions of the number of "elementary" mistakes, consider the hypothetical (single) model and partition of Figure 3.2. In this case, the partition differs from the model only in that the last cluster has been split into two, but the recall score abruptly falls to 50%. Similar observations hold for the collections of binary decisions to which a produced partial ordering is decomposed: transitivity holds since the relation is a partial order, and the antisymmetry constraint (i.e., that if X is semantically less strong than Y then Y cannot be semantically less strong than X, precluding the presence of both pairs (X,Y) and (Y,X) in the relation) forces a wrong decision to be counted as an error twice: for example, if X is incorrectly reported to be semantically less strong than Y, we get a false positive for the association (X,Y) *and* a false negative for the association (Y,X).

The use of multiple models in the evaluation is another cause for lower scores. As long as there is any disagreement between the models, no single partition can reach a perfect score. We can only hope that the system will select the combination of decisions that maximizes the score, but what the maximum score is is not immediately apparent.

In Section 3.6 I return to the topic of correcting the measures described in Section 3.4 for the over-penalizing effects of structural constraints and interdependence of false positive and false negative errors, after discussing empirical methods for estimating an upper and lower bound for the achievable scores. Having an upper bound on the system's score partly compensates for the third distorting effect caused by the presence of multiple reference models, as the score the system actually achieves can be compared and/or normalized with the upper bound.

3.5.2 Deriving bounds on the evaluation measures

In the previous subsection, I discussed the importance of assessing the meaning of the evaluation scores. Ideally, we would like to have a formal description of the variations in the goodness of solutions according to the method used and the problem instance; but since it is far from clear what the mathematical model for such a description would be, much less how we would actually compute the description for a given problem, we have to resort to less ambitious, partial indicators of the difficulty of the problem. To this end, I have developed methods for providing upper and lower bounds to the evaluation metrics for the various tasks described in this dissertation.

Estimating an upper bound For the upper bound, I use the evaluation methods defined above to evaluate each model provided by the human judges against all other models, and then I average the resulting scores. This is consistent with the a priori assumption that the system cannot be better than the models, which is partially justified from the fact that the humans in general produce results of satisfactory quality. Note that a slight asymmetry remains, namely that the system is evaluated against n models but each model is evaluated against $n-1$ other models. Since the disagreement between models increases with the number of models, this introduces a small[13] bias against the system. A possible extension to this method is to use a more complicated algorithm to remove this bias, obtaining the system's score by evaluating the system against each of the n subsets containing $n-1$ models each, and averaging the resulting average scores.

Technically, the method of the previous paragraph does not provide an upper bound on the performance of a single partition relative to the standard established by the models. Rather, it offers a means for comparing the performance of the system to the humans. Another kind of upper bound is to calculate the best score that *any* single-partition solution could ever achieve. That best solution would not, in general, be scored as perfect because, in the presence of disagreeing multiple models, the system can never receive a perfect score for a given decision point no matter what decision it makes as long as the fractional positive or negative values are not 0 or 1. A future extension that can provide a true upper bound on the *possible* score of any solution is to investigate techniques for determining the best decisions from the models. In particular, we can look for a formula that would relate the maximum score to the measures of agreement of the models on each possible association (i.e., the fractional positive and negative values defined in Section 3.4). If a closed formula for the maximum score cannot be found, one can use an approximation method which will

[13]Small for reasonable values of n, e.g., n close to 10, as used in the experiments reported in Sections 4.1.3–4.1.4 and 4.2.2.

locate local maxima of the evaluation score. Such an approximation method can be based on the clustering algorithm of Section 2.7.1 with an appropriate change of the objective function: the models will be made visible to the clustering component, and the objective function will be inversely related to the actual evaluation score that the currently considered partition would receive.

Estimating a lower bound In order to provide a lower bound for the evaluation metrics, we need to define a performance baseline that can be easily reached. When partitions or ordering decisions are evaluated, no simple algorithm that exploits some readily available knowledge is applicable, so the appropriate baseline is offered by a system which randomly partitions the words into groups or orders them; randomly in these cases means that all possible partitions or orderings are equiprobable. When the stages of the system that produce binary answers are evaluated, the situation is simpler since there are no consistency constraints between the various decisions. In this case, we can take advantage of previous knowledge about the distributions of the two answers, i.e., whether a "yes" or a "no" answer is more likely. Algorithms of this type that use minimal a priori distributional knowledge but do not take advantage of the context for each independent decision have been frequently used to supply an indication of baseline performance, e.g., for the problems of prepositional phrase attachment [Whittemore *et al.* 1990; Hindle and Rooth 1991] and word sense disambiguation [Dagan *et al.* 1991; Gale *et al.* 1992a; Dagan and Itai 1994]. But in all subproblems addressed in the thesis with answers of this type both outcomes are equiprobable, so these minimal-knowledge algorithms become equivalent to random decisions with equal probabilities for the two outcomes.

The expected performance of the random algorithms can in theory be determined from the models, according to the fractional positive and negative values of each decision and other factors that affect the structured solutions (groups and orderings) such as the number of clusters and the number of words that are clustered or ordered. However, determining these expected values analytically can be very complicated. In the simplest case, namely the problems with solutions in the form of independent binary decisions with no consistency constraints, we can easily estimate the quantities a, b, c, and d of the contingency table model (Table 3.1) on which the evaluation metrics are based. For example, since the probability of saying "yes" to each question is 0.5 and any given question has a known fractional positive value, the expected value (mean) of the quantity a for the contingency table model of Table 3.1 will be

$$\mathbf{E}[a] = 0.5 \cdot \sum_{\text{every question } i} \text{positive value of question } i$$

Similar formulas can also be derived for the quantities b, c, and d of the contingency table. But in order to compute the average value of precision, we need also to estimate the covariance between a and b, and the situation becomes even more complicated for the computation of the average F-measure value. When the decisions are structurally interdependent, as in the case of groupings and orderings, even the determination of the expected values of a, b, c, and d is difficult.

Since finding the expected performance scores analytically is very complicated if not practically impossible, I use instead a method based on Monte Carlo simulation [Rubinstein 1981]. The method consists of repeatedly creating random solutions according to the prior distributions discussed above and evaluating each of them. Then I estimate a smoothed probability density function for each metric from the resulting histograms, using non-parametric kernel estimates based on a Gaussian-shaped window [Silverman 1986]. Since a density function is obtained in this way, we can calculate any moment, including the mean and variance. Furthermore, we can perform the Monte Carlo simulation to any desired degree of accuracy by increasing the number of sample solutions used[14], and we can use non-parametric statistical tests (e.g., the sign test [Gibbons and Chakraborti 1992]) to assess the statistical significance of the difference between the system's and the random algorithm's performance. In this way, we can establish that the system does indeed perform significantly better than would be attributable to chance.

3.6 Adjusting the evaluation metrics to account for structural constraints, interdependence, and multiple models

Throughout this chapter, I emphasized that the generalized evaluation metrics of Section 3.4 are characterized by non-linearities, i.e., abrupt changes in the score caused by only small changes in the evaluated solution. These non-linearities result in much lower scores than would be the case if the decision points were independent or if a single reference model

[14] I currently run the simulation for 20,000 randomly drawn solutions for each problem instance that is analyzed.

were used. Although they do not affect our ability to make comparisons between different variants of the system since better performance will be associated with higher scores, they do impair the interpretation of *differences* in the evaluation measures, as humans tend to impose a linear scale on them. In other words, this consistent property of the evaluation metrics is not a problem of substance, but it can be a problem of perception.

I noted earlier (Section 3.5) that it is impossible to assess the meaning of an evaluation score without knowing at least some aspects of the distribution of the possible scores for the given problem, for example how hard it is to achieve this score with a minimal, baseline approach and how close this score is to the maximum score achievable because of constraints external to the evaluated system. Certainly, one cannot arbitrarily compare scores, even if the same evaluation approach is used (e.g., precision and recall for lists of "yes"–"no" decisions), across different tasks. Imagine the turmoil that would be caused if someone consistently predicted the outcome of a fair coin's toss with 51% accuracy; on the other hand, who in the computational linguistics community would be impressed with an English part-of-speech tagger with 90% precision? Or, for an example outside linguistics, consider a recent letter to the *New York Times* in favor of basic rights for primates because of, among other factors, their close genetic relatedness to humans ("more than 98%").[15] Yet, it is not mentioned than any random, lowly bacterium shares at least 90% of the DNA strand with humans.

Despite these observations, even otherwise knowledgeable people tend to interpret scores in an absolute manner, resulting in comments such as the one reproduced in footnote 12

[15]Paola Cavalieri, Letters, *New York Times*, February 10, 1997, p. A14.

on page 153.[16] Accordingly, more to account for this misconception than for any other significant reason, I have developed adjustments to the evaluation metrics of Section 3.4 that account for the identified non-linearities.

In Section 3.5.1 I mentioned that providing an upper bound for any given metric, based on the properties of a collection of reference models, is a way of addressing the low scores caused by disagreements among the multiple models. The score actually achieved by a system can be normalized by dividing it with this maximum value. In this section, I present adjustments to the evaluation metrics, and in particular to the quantities a, b, c, and d of Section 3.4, that correct for the other two causes of low scores: the abrupt jumps in the score traceable to the structure of the original solution before its conversion to a list of "yes"–"no" decisions, and the apparent doubling of errors because of the pairing of false positives and false negatives.

3.6.1 Adjusting the scores to account for structural constraints

When a partition produced by the system or the human evaluators is converted to a series of "yes"–"no" decisions, each element in each group is mapped to multiple such decisions. There is interdependence between these decisions, and a consequent drop in the evaluation metrics of Section 3.4 that can be traced to two reasons:

[16]Thus we reach the "80% rule": If your system achieves 80% "performance", no matter the metric employed, your paper merits publication.

- The transitivity of the "is associated with" relation of Section 3.2. I noted in Section 3.5.1 that a group with k elements introduces $O(k^2)$ interactions between its elements, and a corresponding number of association values participate in the computation of potential false positives. If the system erroneously associates X with Y and Z with X, it should not be penalized for the necessary association of X and Z if the latter is wrong, as that association follows from the structure of the solution and is not an independent choice made by the system.

- The fact that each clustered item (word) is placed into one and only one cluster. A word may be associated in the models with several words belonging to different groups, but the system can place it only in one "best" group. Each element that is misplaced out of its correct group results in potentially multiple false negatives for all the elements of the groups it is not moved into, and this problem is exacerbated as the number of groups grows.

The first phenomenon results in a large number of apparent false positives stemming from a single move of an element from one cluster to another. In order to correct for these abrupt changes in the evaluation metrics, we can adjust the values of the penalties that are added to the appropriate cells of the contingency table (cells b and c of Table 3.1 on page 142). Let the total penalty for false positives for a given word x be

$$\Delta b(x) = \sum_{\substack{y \in C_x \\ y \neq x}} \text{fractional negative value of the association } (x, y) \tag{3.6}$$

where C_x is the group that word x is placed in the evaluated partition (see Section 3.4). Clearly, the value of cell b in the contingency table without any adjustments is

$$b = \frac{1}{2} \cdot \sum_x \Delta b(x)$$

where the summation takes place over all words x in the set of words we cluster.[17]

For words that are placed in non-singleton groups, I replace the quantity $\Delta b(x)$ in equation (3.6) above with the revised value

$$\Delta b'(x) = \frac{\Delta b(x)}{|C_x| - 1} \tag{3.7}$$

Words that are placed in a group by themselves cannot generate false positive errors; thus, for these words,

$$\Delta b'(x) = \Delta b(x) = 0$$

Then I add these revised values $\Delta b'(x)$ across all words x, for a total penalty for false positives equal to

$$b' = \frac{1}{2} \cdot \sum_x \Delta b'(x)$$

For singleton groups, the revised penalty for false positives is identical to the original penalty; for groups with $|C_x| > 1$, there are $|C_x| - 1$ interactions of x with other words in the group, so this normalization corresponds to averaging and distributing the false positive

[17] The factor $\frac{1}{2}$ is due to the fact that the contribution of each word pair (x, y) is included twice in the sum, once for x and once for y.

errors associated with x across all members of its assigned group.

Multiple false negative errors stemming from a single wrong placement decision for a word are associated with the second type of structural constraint in the partition, namely that the word cannot belong to more than one group, although it may be associated with words placed in different groups in the clustering. To alleviate the effect this has on the total penalty for false negatives for a word, I replace the total penalty Δc that would have been added to cell c of Table 3.1 for a word x,

$$\Delta c(x) = \sum_{y \notin C_x} \text{fractional positive value of the association } (x, y) \tag{3.8}$$

with the revised value

$$\Delta c'(x) = \frac{\Delta c(x)}{m-1} \tag{3.9}$$

where m is the total number of groups in the evaluated partition.[18] The total revised penalty for false negatives for all words is obtained by summing $\Delta c'(x)$ across all words x in the collection of clustered words, i.e.,

$$c' = \frac{1}{2} \cdot \sum_{x} \Delta c'(x)$$

Then we can apply the definitions of generalized precision, recall, fallout, and F-measure given in Section 3.4 by substituting the normalized quantities b' and c' for b and c respec-

[18]I assume here that $m > 1$. If $m = 1$, all words are placed in the same group and there can be no false negatives. In that special case, I define $\Delta c'(x) = 0$ for all x.

tively. Note that since the cells a and d in the contingency table remain unchanged, and obviously $b' \leq b$ and $c' \leq c$, the normalized recall, precision, and F-measure will always be equal to or greater than their original versions, while the normalized fallout will be less than or equal to its original version (see equations (3.1)–(3.3) on page 143, and equation (3.4) on page 145).

3.6.2 Avoiding the double counting of false positive and false negative errors

The fact that a word cannot simultaneously be a member of two groups of a partition leads to a third kind of error overestimation: Whenever a word x is moved from its (non-singleton) correct group K_x to another (also non-singleton) group C_x, two errors are generated in parallel. The word is assigned a total penalty for false positives $\Delta b(x)$ as described by equation (3.6) *and* a total penalty for false negatives $\Delta c(x)$ as described by equation (3.8). Yet it can be argued that the application of both of these penalties is unfair, since one of them cannot be eliminated in favor of the other (as it would be the case if the decisions in the evaluated partition involving x and the words not in C_x were independent of the decisions involving x and the other members of C_x).

To reduce these penalties, we have to assign part of the blame to the evaluation metric for false positive errors and part to the evaluation metric for false negative errors (the distributed errors will be reintegrated during the computation of the F-measure). Because a part of the total penalty for false positives and negatives is counted twice, the total penalty should be $\max(\Delta b(x), \Delta c(x))$. I normalize the two penalties proportionally, replacing $\Delta b(x)$ with $\Delta b''(x)$ and $\Delta c(x)$ with $\Delta c''(x)$ where

$$\Delta b''(x) = \frac{\Delta b(x)}{\Delta b(x) + \Delta c(x)} \cdot \max(\Delta b(x), \Delta c(x)) \tag{3.10}$$

and

$$\Delta c''(x) = \frac{\Delta c(x)}{\Delta b(x) + \Delta c(x)} \cdot \max(\Delta b(x), \Delta c(x)) \tag{3.11}$$

If C_x is a singleton, then necessarily $\Delta b(x) = 0$, so $\Delta b''(x) = 0$ and $\Delta c''(x) = \Delta c(x)$, i.e., the original penalties remain unchanged. Similarly, no redistribution of the penalties takes place if K_x is a singleton in all the reference models (which implies $\Delta c(x) = 0$). In any case, the resulting total penalties for false positives and false negatives never increase, and consequently the values of the evaluation metrics either stay the same or improve. When $\Delta b(x) \approx \Delta c(x)$, the normalization reduces the penalties assigned to each error cell by a factor of almost two. We can also combine the normalizations of equations (3.10)–(3.11) with the normalizations addressing other structural constraints that were presented in the previous subsection, replacing $\Delta b(x)$ and $\Delta c(x)$ with $\Delta b'(x)$ and $\Delta c'(x)$ respectively, as the latter were defined in equations (3.7) and (3.9).

Chapter 4

Output from the word grouping system

The grouping stage of the present system was the one first implemented, initially classifying adjectives only. I have since extended that stage to nouns, but have performed most empirical studies of the system's performance with collections of adjectives. I thus devote the first and larger part of this chapter to the presentation of a part of these experiments, which show that the system is able to produce semantic groups of adjectives that are far better than random ones and reasonably close in quality to those produced by humans. In Section 4.1.1 I present several sets of adjectives drawn from general news and financial reports corpora that I used as input to the word clustering subsystem and explain how these particular adjectives were selected. Then, I give sample partitions produced by the system for these adjectives, sample models for the same adjectives that have been constructed by humans, and the results of the evaluation of the system's output according to these models, using the methodology of Sections 3.2–3.4. I also present the results of the various normalization operators of Section 3.6 when they are applied to the scores of the evaluation metrics for these partitions. I conclude Section 4.1 by presenting meta-evaluation results (cf. Section 3.5.2), which quantitatively establish the claims made above about the position of the system's performance in the range between random and human-prepared groupings.

In Section 4.2 I turn to experiments performed on collections of nouns, using data from the large Wall Street Journal corpus of articles from the financial domain. Again, I provide some sample solutions produced by the system and the results of the original and normalized evaluation metrics on the produced partitions. In the interest of brevity, I do not supply details of the evaluation models or extended output from the meta-evaluation procedure for these noun sets. These are completely analogous to the ones described for adjective sets in Sections 4.1.3 and 4.1.4.

4.1 Results on adjectives from general and financial corpora

4.1.1 Selecting the test sets of adjectives

The performance of the word grouping system depends naturally on the set of words[1] that are to be clustered, and several descriptive characteristics of this set can be specified and studied for their effect on the quality of the produced groupings. Presumably, variations in the adjective set can be modeled by several parameters, such as the size of the set, the

[1]In the sample results of this section, adjectives.

antitrust	global	mechanical	real	unexpected
big	international	new	serious	
economic	legal	old	severe	
financial	little	political	staggering	
foreign	major	potential	technical	

Figure 4.1: Adjective test set 1: Adjectives appearing in the Associated Press corpus strongly associated with the word *problem*.

number of "true" semantic groups in it, and the strength of semantic relatedness among its members. In addition, several parameters describing the properties of the adjectives in the set in isolation, such as frequency and specificity, may affect the quality of the clustering process. I have selected five adjective sets that represent several different points in the complex space of the parameters outlined above, and thus offer a basis for studying the word grouping subsystem in several quite different types of problem instances.

The system was originally tested on the set of adjectives of Figure 4.1. These adjectives were extracted from the corpus that I originally used, a collection of stock market reports from the Associated Press newswire (see Section A.2 in the Appendix for a description of this corpus). The criterion for the inclusion of any particular adjective was its degree of co-occurrence with a particular noun specified in advance, *problem*, as measured by the collocation extraction tool XTRACT [Smadja 1991; Smadja 1993]. In this way, I collected a set of adjectives that included words of varying frequency in the text, which nevertheless were semantically associated, since they all frequently modified the same noun.

The second and third sets of adjectives, depicted in Figures 4.2 and 4.3, were extracted from the corpus I used subsequently for most of the analyses reported in the thesis, the

annual	final	low	past	regional
big	future	military	positive	senior
chief	hard	modest	possible	significant
commercial	high	national	pre-tax	similar
current	important	negative	previous	small
daily	initial	net	private	strong
different	international	new	public	weak
difficult	likely	next	quarterly	
easy	local	old	recent	

Figure 4.2: Adjective test set 2: Adjectives from the 1987 Wall Street Journal corpus occurring more than 1,000 times.

abrupt	delicate	fragile	misleading	stringent
affluent	dirty	generous	multiple	stunning
affordable	disastrous	gloomy	outrageous	stupid
astonishing	dismal	gradual	plain	toxic
brilliant	dry	harmful	pleasant	turbulent
capitalist	dual	hazardous	prosperous	unfriendly
catastrophic	dumb	hostile	protracted	unreasonable
chaotic	endless	impoverished	rigid	unstable
clean	energetic	inexpensive	scant	vigorous
clever	exotic	insufficient	slim	wet
communist	fat	leftist	smart	
confusing	fatal	lenient	socialist	
deadly	flexible	meager	strict	

Figure 4.3: Adjective test set 3: Adjectives from the 1987 Wall Street Journal corpus occurring between 50 and 250 times.

1987 Wall Street Journal articles.[2] The words in these sets were selected to represent opposing points in the spectrum of word frequency and specificity. It has been known for quite some time [Zipf 1945] that there is an inverse correlation between the text frequency of a word and its specificity, i.e., more frequent words tend to be more polysemous. I have

[2]See Section A.2 in the Appendix for a description of this corpus and for a discussion of why I switched to it.

experimentally verified this association between frequency and specificity in the particular case of adjectives, by observing the candidate words for inclusion in these two sets. The words of Figure 4.2 were selected so that they all are quite frequent, with more than 1,000 occurrences each in the 21 million word corpus. The words of Figure 4.3 satisfy the opposite constraint: they occur with relatively low frequency, between 50 and 250 times in the 21 million word corpus.[3] In both cases, I further restricted the sets of eligible adjectives to maintain some semantic relatedness between subsets of them (e.g., including all synonyms or antonyms if possible) while keeping the number of words in each set at manageable levels.

The three adjective sets described above contain 21, 43, and 62 adjectives respectively, numbers that I felt were large enough to make the clustering task non-trivial and at the same time small enough to make the construction of models by humans practical. However, to demonstrate the ability of the system to function on larger collections of words, I also selected all adjectives appearing more than 200 times in the Associated Press corpus of 8.2 million words. This resulted in a set of 171 adjectives, given in Figure 4.4. Unlike the other three sets of adjectives, this set was not filtered to limit it to well-behaved or "interesting" cases only. I have left in it inflected forms of adjectives (comparative and superlative) if they passed the frequency threshold, e.g., *good–better–best* and *high–higher–highest*, and did not use the morphological reduction methods of Section A.5 in the Appendix on the members of this set during the processing of the extracted corpus data by the system. The set also contains some highly idiomatic words (e.g., *own*, as in the phrase *your own home*), and

[3] I did not consider words of truly low frequency (e.g., appearing once, twice, or even ten times in the corpus), because statistical methods on data that is so sparse can easily be misled. This problem can be rectified by using a larger corpus, and thus increasing the frequency of occurrence of any particular word that we want to study.

active	economic	institutional	outside	significant
additional	executive	international	outstanding	similar
annual	federal	jobless	over-the-counter	single
apparent	final	joint	overall	slight
approximate	financial	key	overseas	slow
average	foreign	large	own	sluggish
bad	former	late	past	small
best	fresh	latest	political	special
best-known	friendly	legal	poor	steep
better	full	light	positive	straight
big	further	little	possible	strong
blue	future	local	potential	substantial
brief	general	long	powerful	technical
broad	giant	low	precious	temporary
certain	global	lower	preferred	top
cheap	gold	main	preliminary	total
chief	good	major	presidential	traded
civilian	great	massive	previous	troy
commercial	gross	military	prime	unprecedented
common	healthy	moderate	principal	upward
complete	heavy	modest	private	various
composite	high	monetary	public	visible
congressional	higher	monthly	quarterly	volatile
consecutive	highest	mutual	real	weak
consolidated	historic	narrow	recent	weekly
continued	huge	national	regional	wholesale
corporate	immediate	nationwide	regulatory	wide
current	impending	negative	retail	widespread
daily	important	net	rival	wild
different	increased	new	senior	worldwide
domestic	independent	normal	separate	worst
downward	individual	nuclear	serious	
dramatic	industrial	odd	severe	
durable	inflationary	old	sharp	
early	initial	opposite	short	

Figure 4.4: Adjective test set 4: All adjectives appearing in the Associated Press corpus more than 200 times.

words that are very specialized to the domain (e.g., *inflationary*, *troy*[4] as in *U.S. dollars*

[4] A rather obscure adjective designating a British system of weights for precious metals based on a pound of 12 ounces and an ounce of 480 grains.

advance-purchase	convenient	foregone	noisy	self-help
anti-fraud	declined	hapless	non-performing	single
archaic	double-B-minus	high-flying	noticeable	six-year-old
best-performing	expanded	high-profile	old-fashioned	subordinate
bid-rigging	faint	initial	outspoken	sustainable
bogus	faithful	Japanese-made	Peruvian	twin-engine
bumpy	fashionable	less-developed	plain	unofficial
complacent	favorable	ludicrous	pre-crash	unpopular
computer-related	follow-up	murky	reputable	world-wide
contentious	forceful	nifty	rustic	worrisome

Figure 4.5: A random sample of 50 words from adjective test set 5: All 3,243 adjectives occuring in the 1987 Wall Street Journal corpus 20 times or more.

per troy ounce, and the compound *over-the-counter* as in *over-the-counter trading*).

A final test set of adjectives was drawn from the Wall Street Journal corpus and used to demonstrate the scalability of the approach. This adjective set contained 3,243 words, and included all adjectives appearing at least 20 times in the corpus. A random sample of 50 words from this set is shown in Figure 4.5. Note that the large size of this test set precludes the application of the formal evaluation methods of Chapter 3; it is hard to convince human volunteers to classify 3,243 words into semantic classes! In addition, some performance degradation is expected because of the lower frequencies of the words in this set, a condition that can be nevertheless amended if a larger input corpus is used. I used therefore this set of adjectives to show that the system can operate on truly large sets of words, and to study the practical time complexity of the adopted algorithms.

4.1.2 Partitions found for the test sets of adjectives

The word clustering system was run on the five adjective sets discussed above, automatically producing partitions of these sets for several different values of the number of clusters parameter. Initially, I used an external parser to identify and analyze noun phrases, plus a collection of tools I developed which extracted from the noun phrases adjective-noun and adjective-adjective pairs. The parser employed was FIDDITCH [Hindle 1983; Hindle 1994], a robust parser that is able to produce partial parse trees whenever it cannot fully analyze an input sentence. The tools that processed the noun phrases used a subset of the heuristics described in Section A.4.3 of the Appendix to collect appropriate instances of the syntagmatic relationships of Section 2.1, as further elaborated for the case of grouping adjectives in Section 2.2.3. The parser and the extracting tools were run on the 8.2 million word corpus of Associated Press newswire reports about the stock market (see Section A.2 in the Appendix).[5]

Although the FIDDITCH parser produces output of relatively high quality, it requires a significant amount of time to process a large amount of text, which becomes a more pressing concern when a very large corpus is used. Furthermore, the parser is not distributable outside AT&T, which greatly inconveniences the process of experimenting with new corpora in different domains, or with subsets of corpora of increasing sizes. For these reasons, I decided to switch from using FIDDITCH to a finite state parser I developed and which is described in detail in Sections A.4.2–A.4.4 of the Appendix. I also switched from the

[5] I want to thank Diane Litman and Donald Hindle, for arranging for and actually performing, respectively, the parsing of this corpus with FIDDITCH at AT&T Bell Laboratories.

	Initial version	Current version
Method for identifying syntagmatic relationships	FIDDITCH parser plus subset of heuristics of Section A.4.3 of the Appendix	Finite state parser (Section A.4.4 in the Appendix) plus full set of heuristics of Section A.4.3 of the Appendix
Types of adjective-noun modification extracted	Attributive only	Attributive, predicative (i.e., subject complement), object complement, adjective apposited to the noun (Section A.4.1 in the Appendix)
Negative examples (adjective-adjective pairs) used?	Yes	Yes
Morphological reduction used?	For nouns only	For both adjectives and nouns
Spell-checking used?	At the full word level only	Full words and separately for constituents of hyphenated compound words
Corpus source	1987–1988 Associated Press newswire	1987 Wall Street Journal articles
Corpus size	8.2 million words	21 million words
Corpus domain	Limited to stock market reports	Mostly financial, but also many articles of general interest
Grouping results presented	Adjective sets 1 (Figure 4.6) and 4 (Figure 4.10)	Adjective sets 1 (Figure 4.7), 2 (Figure 4.8), and 3 (Figure 4.9)

Table 4.1: The two variants of the grouping system that have been used for partitioning sets of adjectives from the Associated Press and Wall Street Journal corpora. The two versions are contrasted on their components and features, as well as on the corpora they use.

Associated Press newswire corpus to the cleaner, more general, more widely available, and larger Wall Street Journal corpus; a description of this corpus and a more detailed analysis of its advantages is given in Section A.2 in the Appendix. Consequently, two variants of the system have been used for the grouping task of adjectives from this domain. These two variants are contrasted in Table 4.1, according to the differences in their components and the corpora they employ.

1. foreign global international
2. old
3. potential
4. new real unexpected
5. little staggering
6. economic financial mechanical political technical
7. antitrust
8. big major serious severe
9. legal

Figure 4.6: Partition with 9 clusters produced by the system (initial version) for the adjective test set 1 of Figure 4.1.

1. potential
2. foreign mechanical
3. new old staggering
4. little
5. global international
6. big major serious severe
7. real unexpected
8. economic financial political technical
9. antitrust legal

Figure 4.7: Partition with 9 clusters produced by the system (current version) for the adjective test set 1 of Figure 4.1.

Figures 4.6 and 4.7 show the partitions produced by the initial and current versions of the grouping subsystem respectively for the first set of adjectives (Figure 4.1). Similarly, Figures 4.8 and 4.9 show the partitions produced for adjective sets 2 and 3 (Figures 4.2

1. private public senior	9. negative positive
2. strong weak	10. difficult important likely possible similar
3. annual quarterly	11. chief
4. international local national regional	12. current future initial past previous
5. high low pre-tax	13. big modest significant small
6. commercial	14. military net
7. final next recent	15. daily
8. different easy hard new old	

Figure 4.8: Partition with 15 clusters produced by the system (current version) for the adjective test set 2 of Figure 4.2.

1. affluent impoverished prosperous	13. plain
2. catastrophic disastrous harmful	14. abrupt gradual stunning
3. capitalist socialist	15. dirty endless protracted
4. affordable inexpensive	16. fat slim
5. communist leftist	17. dry exotic wet
6. clean dumb smart stupid	18. dual multiple pleasant
7. generous outrageous unreasonable	19. hazardous toxic
8. flexible lenient rigid strict stringent	20. deadly fatal
9. brilliant clever energetic	21. hostile unfriendly
10. chaotic delicate fragile turbulent unstable	22. dismal gloomy
11. astonishing meager vigorous	23. confusing misleading
12. insufficient scant	

Figure 4.9: Partition with 23 clusters produced by the system (current version) for the adjective test set 3 of Figure 4.3.

and 4.3) respectively, using the current version of the system, and Figure 4.10 gives the partition obtained by the initial version of the system for adjective set 4 (Figure 4.4). Since it is impractical to show partitions for many different values of the number of clusters parameter, for the first three sets I present the partition found when this parameter was set to the average number of clusters used by the human evaluators for the corresponding set of words (9, 15, and 23 clusters respectively). However, I also give in Figures 4.11 and 4.12 the partitions obtained by the current version of the system for the first (smallest) test set when 8 or 10 clusters were employed instead of 9. For the fourth, larger set of words, I looked at clusterings within the range of the number of clusters parameter suggested by the analysis of Section 2.7.3, i.e., 61 to 71 clusters, and near the endpoints of that range. The partition with 73 clusters included several interesting placements of words in groups, so I display that one in Figure 4.10; other partitions with similar numbers of clusters were comparable in quality to the one shown. Results from the fifth set of 3,243 words are not shown because of the very large size of that set.

Before proceeding with the formal evaluation of these results in the next subsection, it is worth noting several points that become apparent by observing the associations between words that the system identifies (and thinking about the ones it fails to identify). First, the system generally identifies both groups of complementary non-gradable adjectives and groups of gradable adjectives which fall in the same scale. The former case is exemplified by groups such as {*current*, *future*, *initial*, *past*, *previous*} (group 12 in Figure 4.8), {*domestic*, *global*, *overseas*, *worldwide*} (group 30 in Figure 4.10), and {*annual*, *daily*, *monthly*, *quarterly*, *weekly*} (group 72 in Figure 4.10). The scalar groups can be further subdivided into

1. normal	25. negative positive
2. big dramatic sharp slight steep	26. serious
3. local regional	27. historic unprecedented
4. heavy light	28. full wide
5. downward upward	29. early initial jobless late
6. best-known important key main	30. domestic global overseas worldwide
7. special wild	31. final preliminary
8. industrial over-the-counter	32. temporary
9. immediate	33. common preferred
10. congressional federal joint troy various	34. cheap total
11. presidential	35. healthy slow sluggish strong weak
12. inflationary	36. latest narrow
13. moderate modest overall	37. best better good poor
14. public volatile	38. legal nuclear
15. own retail wholesale	39. huge large net small
16. active certain precious	40. different opposite
17. approximate significant substantial	41. average consolidated
18. additional fresh gross	42. major powerful principal
19. economic technical	43. blue separate
20. national nationwide	44. mutual real
21. giant	45. high higher highest low lower
22. financial independent outside	46. widespread
23. foreign individual institutional	47. great
24. corporate odd single	48. past previous

Figure 4.10: Partition with 73 clusters produced by the system (initial version) for the adjective test set 4 of Figure 4.4. Continued on the next page.

49. possible potential prime	62. apparent
50. complete	63. brief
51. international outstanding traded	64. friendly new private rival
52. consecutive recent straight	65. continued little
53. bad	66. current monetary
54. commercial	67. chief senior top
55. massive	68. civilian durable military
56. severe worst	69. future impending
57. general gold	70. further similar visible
58. increased	71. executive former
59. broad composite	72. annual daily monthly quarterly weekly
60. political regulatory	73. long short
61. old	

Figure 4.10: Partition with 73 clusters produced by the system (initial version) for the adjective test set 4 of Figure 4.4. Continued from the previous page.

the following categories:

- Groups that contain near synonyms, e.g., {*affordable*, *inexpensive*} and {*deadly*, *fatal*} (groups 4 and 20, respectively, in Figure 4.9).

- Groups that consist of a pair of antonymous words, e.g., {*negative*, *positive*} (group 9 in Figure 4.8), {*heavy*, *light*} (group 4 in Figure 4.10) and {*final*, *preliminary*} (group 31 in Figure 4.10).[6]

[6]Note that *final* is not gradable, and yet it participates in an opposition relationship with the gradable adjective *preliminary*. This phenomenon, which also extends to verbs (e.g., *judge–misjudge*), is further discussed in Section 6.1.

1. foreign mechanical
2. big major serious severe
3. antitrust legal
4. economic financial political technical
5. little real staggering
6. new old unexpected
7. global international
8. potential

Figure 4.11: Partition with 8 clusters, one less than the average number used by humans, produced by the system (current version) for the adjective test set 1 of Figure 4.1. Compare with Figures 4.7 and 4.12.

---

1. antitrust legal
2. real unexpected
3. economic financial political technical
4. new old staggering
5. global international
6. potential
7. big major severe
8. serious
9. foreign mechanical
10. little

Figure 4.12: Partition with 10 clusters, one more than the average number used by humans, produced by the system (current version) for the adjective test set 1 of Figure 4.1. Compare with Figures 4.7 and 4.11.

- Groups that are one-sided subscales, e.g., {*catastrophic*, *disastrous*, *harmful*} (group 2 in Figure 4.9) which can be ordered as *harmful* $\prec$ *disastrous* $\prec$ *catastrophic* using the relation "is semantically less strong" ($\prec$) of Section 1.7.

- Groups that are full scales and include words of both positive and negative orientations, e.g., {*big*, *modest*, *significant*, *small*} (group 13 in Figure 4.8).

Second, the produced groups depend on the domain. This is apparent from groups such as {*common*, *preferred*} and {*healthy*, *slow*, *sluggish*, *strong*, *weak*} (groups 33 and 35, respectively, in Figure 4.10). These groups may initially seem wrong but are justified when we take into account the specific domain for which the grouping was made: In a corpus of stock market reports, it is most likely that *common* and *preferred* refer to types of stocks, while the words of the second group given above refer to different states of the economy. Note that the human evaluators may be unaware of these fine specializations of the word meanings according to the domain, and thus penalize the system for actually discovering domain-specific semantic relationships.

Another interesting observation is that the individual placements of words in clusters also depend on the overall set of words that are analyzed. Depending on the possible semantic "neighbors" that are available for inclusion in a group, the system may prefer a particular interpretation for a polysemous word over another. As an example, consider the adjective *big*, which is highly polysemous. When it is clustered together with adjectives selected for their close association with the noun *problem*, it is placed in the group {*big*, *major*, *serious*, *severe*} (group 6 in Figure 4.7). This is quite appropriate for describing one aspect (magnitude) of problems, since, for problems, magnitude can be equated with severity. However, when *big* is clustered with adjectives selected according to more general criteria, it is placed in the cluster {*big*, *modest*, *significant*, *small*} (group 13 in Figure 4.8). In this case, the cluster produced is more general in meaning, as it is no longer valid to consider something *big* as *severe*. Note that both of these groupings were produced using the same variant of the system and the same corpus, the only difference being the adjective set on which the system operated.

Finally, by looking at cases where the system fails to produce correct output, I note that sometimes an inappropriate word is inserted into an otherwise valid cluster (e.g., *pleasant* in {*dual*, *multiple*, *pleasant*}, group 18 in Figure 4.9), or a word is missing from its correct group (e.g., *presidential* in Figure 4.10, which is placed in group 12 by itself instead of with the closely associated words *congressional* and *federal*).[7] I have noticed that words that are semantically "empty", i.e., are frequently used in a very general way without significantly constraining the meaning of the modified term, are particularly prone to this type of wrong association or omission. Moreover, such words often move a lot from cluster to cluster as the number of clusters in the partition changes; consider, for example, the movements of *little* and *real* in the partitions given in Figures 4.7, 4.11, and 4.12. The instability of these words as the number of clusters varies may offer a means for identifying them, and thus enable a separate, more accurate grouping of the remaining, more informative words.

[7]As noted in Section 3.5.1, these two types of errors will most of the time come in pairs.

1. antitrust legal	1. big little major
2. economic financial	2. antitrust
3. foreign global international	3. economic financial
4. big little major	4. foreign global international
5. new old	5. political legal
6. potential unexpected	6. mechanical
7. real serious severe staggering	7. new old
8. technical mechanical political	8. potential real
	9. serious severe
	10. technical
	11. staggering unexpected

Figure 4.13: Two of the nine human-prepared models that were used for the evaluation of adjective test set 1 (Figure 4.1).

4.1.3 Results of the evaluation process

For the formal evaluation of the system output that was presented in the previous subsection, I solicited human-constructed models for each problem instance. Faculty, research scientists, and graduate students in computational linguistics, as well as other students who had taken at least an introductory course in natural language processing, provided between 8 and 9 models for each of the three adjective test sets of Figures 4.1–4.3.[8] Two of the models used during the evaluation of the first adjective test set are shown in Figure 4.13. As is

[8]I want to thank the people who at various times during the development of the system constructed one or more such models: Michael Elhadad, Pascale Fung, Chih-Peng Hung, Diane Litman, Hsin Liu, Kathy McKeown, Tom O'Donnell, Rebecca Passonneau, Dragomir Radev, Jacques Robin, James Shaw, Eric Siegel, Frank Smadja, David Torok, Tony Weida, David Worenklein, and David Yang.

System version	Adjective test set	Number of clusters	Recall	Precision	Fallout	F-measure
Initial	Test set 1	8	37.31	38.10	6.89	37.70
		9	49.74	46.38	6.54	48.00
		10	35.23	41.98	5.54	38.31
Current	Test set 1	8	39.90	40.74	6.60	40.31
		9	38.34	43.27	5.72	40.66
		10	31.61	42.36	4.89	36.20
	Test set 2	14	42.46	45.24	3.64	43.80
		15	42.46	46.06	3.52	44.19
		16	42.46	50.67	2.92	46.20
	Test set 3	22	46.59	50.18	1.86	48.32
		23	40.27	46.83	1.84	43.30
		24	42.32	52.54	1.54	46.88

Table 4.2: Evaluation scores for the adjective groupings produced by the system that were listed in Figures 4.6–4.9 and 4.11–4.12, as well as other groupings on the same adjective sets but with slightly different values for the number of clusters parameter. None of the score adjustments of Section 3.6 has been applied to the listed scores.

evident from the figure, there are considerable differences between the solutions that humans construct for the same problem instance, which justifies the use of several models during the evaluation (cf. Section 3.4).

The evaluation methods of Sections 3.2–3.4 were applied to the system output, using the collected models as the reference standard. The resulting evaluation scores (without any of the normalization steps described in Chapter 3) are given in Table 4.2 for the groupings produced by the system that were listed in Figures 4.6–4.9 and 4.11–4.12, as well as similar groupings produced for the same test sets but with slightly different values of the number of clusters parameter. Table 4.3 presents the normalized F-measure scores for the same clusterings according to several combinations of the methods of Section 3.6.

In the case of adjective test set 4, I did not apply the full evaluation method because

System version	Adjective test set	Number of clusters	Base score	I	II	III	I and III	I, II, and III
Initial	Test set 1	8	37.70	69.84	66.89	47.45	78.18	83.96
		9	48.00	88.92	78.38	55.98	92.24	95.33
		10	38.31	70.97	70.02	47.22	77.81	86.59
Current	Test set 1	8	40.31	74.68	68.52	51.25	84.45	85.64
		9	40.66	75.32	69.31	50.25	82.80	85.68
		10	36.20	67.06	65.95	43.88	72.30	81.67
	Test set 2	14	43.80	68.15	79.07	51.88	76.98	86.84
		15	44.19	68.76	80.57	51.88	76.98	88.07
		16	46.20	71.88	81.75	54.29	80.56	89.15
	Test set 3	22	48.32	76.61	81.69	56.73	82.55	91.21
		23	43.30	68.65	78.58	51.67	75.19	88.09
		24	46.88	74.33	81.19	54.87	79.85	90.70
Humans	Test set 1	8.56	53.98	100.00	82.25	60.69	100.00	100.00
	Test set 2	15.33	64.27	100.00	93.34	67.39	100.00	100.00
	Test set 3	23.12	63.07	100.00	90.24	68.72	100.00	100.00

Table 4.3: Normalized F-measure scores for the adjective groupings for which unnormalized scores were given in Table 4.2. The evaluated partitions include those listed in Figures 4.6–4.9 and 4.11–4.12, as well as other groupings on the same adjective sets but with slightly different values for the number of clusters parameter. The following normalization procedures were applied: (I) Dividing with the average score achieved on the same problem instance by the humans, which, as discussed in Section 3.6, corrects for discrepancies among the models; (II) applying the procedure of Section 3.6.1 to correct for multiple false positives and false negatives resulting from one wrong decision; and (III) applying the procedure of Section 3.6.2 to avoid counting paired false positives and false negatives twice. The normalized average scores of the human-prepared groupings, when each of them is evaluated against the other human-prepared groupings, are included for comparison; the number of clusters listed in these rows is the average value across all human-made models for the current adjective test set. Note that any column involving normalization (I) has necessarily a value of 100% for the human-constructed models.

its relatively large size (171 words) makes the construction of models by humans difficult. Rather, I evaluated the produced partition by considering each of the associations proposed in it and deciding whether it was a valid semantic association in general, whether it could be construed as a valid association in some contexts, or whether it was a completely invalid association. Since this evaluation method only measures the precision of the system (i.e., the method doesn't penalize the system if it *doesn't* propose the correct association

Association grade	First evaluator		Second evaluator		Both evaluators		Average across evaluators	
Clearly valid association	84	(52.17%)	105	(65.22%)	56	(65.88%)	94.5	(58.70%)
Association valid in some contexts	11	(6.83%)	19	(11.80%)	2	(2.35%)	15.0	(9.32%)
Invalid association	66	(40.99%)	37	(22.98%)	27	(31.76%)	51.5	(31.99%)
Association sometimes or always valid	95	(59.01%)	124	(77.02%)	58	(68.24%)	109.5	(68.01%)
Total	161	(100.00%)	161	(100.00%)	85	(100.00%)	161.0	(100.00%)

Table 4.4: Evaluation of associations in the partition of adjective test set 4 produced by the system (Figure 4.10). The number of associations falling under each category are given, along with the category's percentages relative to the total number of associations in the produced partition (161 associations). The percentages in the column labeled "both evaluators" are relative to the total number of associations on which the two evaluators produced the same answer (85 associations, or 52.8% of the total number of associations). Percentages do not always sum to 100.00% because of rounding.

between two words), I also evaluated each of the proposed groups in terms of both the internal associations among its members and missed associations with words that were placed in other groups. With these criteria, clusters were classified as excellent (all words that should be in the group were there and no other words), good (the group contained valid associations but one word was missing or one extraneous word was added to it), questionable (the group contained valid associations but some of them were borderline), or poor (the group contained numerous errors or omissions). Another evaluator[9] independently made similar assignments to the associations and clusters of Figure 4.10. The results of these two evaluation experiments are shown in Tables 4.4 and 4.5.

[9]Thanks to Kathy McKeown for serving in this role.

Cluster grade	First evaluator	Second evaluator	Both evaluators	Average across evaluators
Excellent	26 (35.62%)	30 (41.10%)	23 (42.59%)	28.0 (38.36%)
Good	14 (19.18%)	9 (12.33%)	7 (12.96%)	11.5 (15.75%)
Questionable	8 (10.96%)	5 (6.85%)	4 (7.41%)	6.5 (8.90%)
Poor	25 (34.25%)	29 (39.73%)	20 (37.04%)	27.0 (36.99%)
Excellent or good	40 (54.79%)	39 (53.42%)	30 (55.56%)	39.5 (54.11%)
Total	73 (100.00%)	73 (100.00%)	54 (100.00%)	73.0 (100.00%)

Table 4.5: Evaluation of clusters in the partition of adjective test set 4 produced by the system (Figure 4.10). The number of clusters falling under each category are given, along with the category's percentage relative to the total number of clusters in the produced partition (73 clusters). The percentages in the column labeled "both evaluators" are relative to the total number of clusters on which the two evaluators produced the same answer (54 clusters, or 73.97% of the total number of clusters). Percentages do not always sum to 100.00% because of rounding.

4.1.4 Assessing the evaluation scores

A comparison between the evaluation results given in the previous subsection for different sets of adjectives already illustrates some of the points made in Section 3.5.1, namely that the evaluation of the system's output according to independently prepared models results in scores that should not be interpreted on an absolute scale. Although bias in favor of the system may be present in the evaluation of adjective test set 4 (Tables 4.4 and 4.5) which was done by grading the output actually produced by the system, the results of this evaluation suggest that about 68% of the associations that the system identifies are accurate ones. By visually contrasting the partition obtained for test set 4 (Figure 4.10) with the one obtained for test set 3 (Figure 4.9), it seems that the latter is of a clearly superior quality (e.g., most clusters in the latter partition are valid and complete, whereas the evaluators found only about 42% of the clusters in Figure 4.10 to be of unquestionably high quality).

	Adjective set 1 (9 clusters)	Adjective set 2 (15 clusters)	Adjective set 3 (23 clusters)
Random partitions	9.61	6.20	3.80
System (current version)	40.66	44.19	43.30
Humans	53.98	64.27	63.07

Table 4.6: Lower (random classifier) and upper (humans) bounds on the F-measure score. The scores obtained by the current version of the system are included for comparison. Only the F-measure score is given in the interest of brevity; the other evaluation measures convey a similar picture. None of the score adjustments of Section 3.6 has been applied to the listed F-measure scores.

Yet the partition obtained for test set 3 is given a precision score of only 47% according to the reference models, although it looks more accurate than the partition that was given a precision score of 68% when evaluated by judges who looked at the system's output. This is further supported by the considerable increase in the evaluation scores when normalization procedures that account for the structural constraints in the partitions are employed (cf. Table 4.3). Clearly, we need to further study the range of possible scores for the evaluation method that is based on multiple independent models.

With this goal in mind, I evaluated each of the human-prepared models for each adjective set against the other models, and averaged the results (cf. Section 3.5.2). I also performed a Monte Carlo simulation of the clustering process for each adjective test set and several values of the number of clusters parameter, under the assumption that all possible clusterings were equiprobable, and evaluated the produced partitions against the human-prepared models, in the same way that the system itself was evaluated. Table 4.6 presents the F-measure scores obtained from these analyses, before any of the transformations of Section 3.6 are applied. This measure was selected as the most representative (see Section 3.3), and the other evaluation metrics show a similar picture. The corresponding F-measure scores of the (current version of the) system in the corresponding problem instances are also included in the table for comparison.

Insight into two important aspects of the clustering process when done by humans can be obtained from Table 4.6, as well as by contrasting the humans' performance with the system's as the size and the specificity of the test set varies. The human evaluators can easily produce a solution for the clustering problem as long as the size of the set of clustered words remains small. However, they have problems when the number of words increases. They also find it much more difficult to solve problem instances where the words are ambiguous or polysemous; for example, several evaluators reported to me that they found it more difficult to produce a grouping for test set 2 (43 ambiguous and polysemous adjectives) than for test set 3 (62 less ambiguous adjectives). Note that since test set 3 is about 1.44 times larger than test set 2, a considerably greater number of possible associations between words exists (approximately 2.09 times more such associations). This number increases with the number of words much faster than the number of actual correct associations, so, if all other parameters that affect the difficulty of the task remained constant, we would expect the larger set to be much more difficult to cluster than the smaller set. This is corroborated by the performance of the random classifier (see Table 4.6), which decreases monotonically with the size of the set that is clustered. The fact that the humans and the system obtain essentially the same scores with test sets 2 and 3 is contrary to this expectation, particularly when we consider that the system has actually *less* data (adjective-noun and adjective-adjective pairs) for doing the clustering in the case of test set 3, since the words in that set are significantly less frequent than the words of test set 2. Note that while the F-measure scores remain essentially at a constant level for these two test sets, the fallout score for the system (Table 4.2) falls from an average of 3.36 in test set 2 to an average of 1.75 in test set 3. Since fallout represents the ratio of the errors (false positives) the system reports to the total number of possible errors of this type, the relationship between the F-measure and fallout scores for test sets 2 and 3 establishes that the system has about equivalent overall performance in both cases but to achieve this, it is much more effective in avoiding errors in the latter case. When we consider the F-measure scores of humans, their practically equal inter-agreement on test sets 2 and 3 despite the expected increased difficulty for the latter set may have some interesting psycholinguistic implications.

In terms of speed, the humans outperform the system for small test sets, since the latter has to collect its linguistic knowledge (i.e., the associations between adjectives and nouns) from the corpus, while the humans have ready access to their form of linguistic knowledge about the semantics of the clustered words. However, as the size of the set of clustered words increases, the system spends only a little additional time for collecting the data, and most of its time is spent on the computation of similarities and the clustering itself. According to responses from the human evaluators, it seems that the difference in the time the humans need to solve a larger instance of the problem is significantly larger than the corresponding increase in the time the system needs. Consequently, at some point the system catches up with the humans, and at the size of test set 3 (62 words) it is already somewhat faster than the humans. This provides another justification for using an automatic method for this task, especially if we expect to repeatedly perform it, for example as we move between

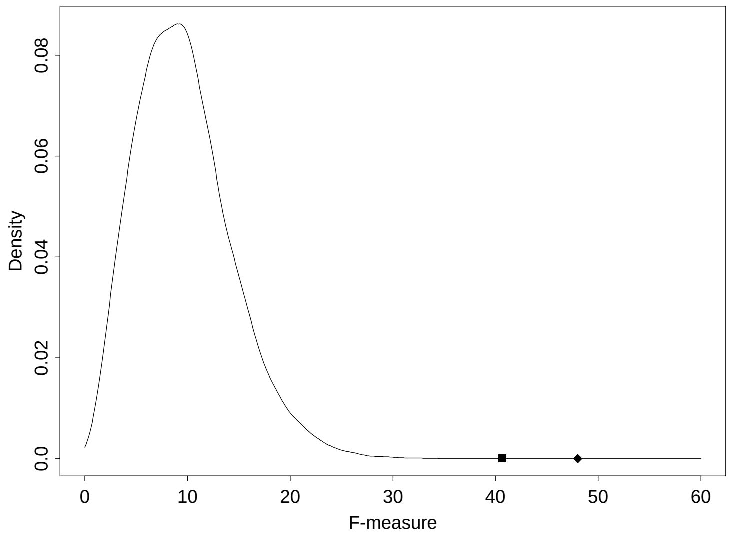


Figure 4.14: Estimated probability density for the F-measure score of random partitions of adjective test set 1 (Figure 4.1) into 9 clusters. The diamond and the square in the figure mark the scores obtained for this clustering task by the initial and current version of the system respectively. The F-measure scores do not incorporate any of the adjustments of Section 3.6.

different domains.

Finally, it should be noted that the results of the Monte Carlo simulation are not just the average scores used to establish a lower bound on the system's performance and reported in Table 4.6. Rather, an entire probability density function is estimated for each adjective set and evaluation metric I use. Figure 4.14 shows the smoothed density function that was obtained for the F-measure score of partitions of test set 1 into 9 clusters, without applying any of the score adjustments discussed in Section 3.6[10]; a diamond and a square in the

[10] Note the resemblance of this empirical density function to a gamma distribution [Johnson and Kotz 1970], or more particularly to a chi-square distribution with around 12 degrees of freedom. This is a topic I may explore more in the future, e.g., by making quantitative comparisons between the empirical distribution and a parametric gamma or chi-square distribution or by seeking reasons for this particular shape of the empirical distribution.

figure mark the corresponding F-measure scores of the initial and current versions of the system respectively. The estimated density function confirms the existence of constraints on the evaluation metric (cf. Section 3.5.1) due to the structure imposed by the clustering, as the F-measure score tends to peak at some point and then fall rapidly.

The availability of a density allows us to perform statistical tests to determine the significance of the observed difference in performance between the system and the random classifier. Alternatively, a sign test [Gibbons and Chakraborti 1992] can be used for this purpose, without making any distributional assumptions. The sign test statistic is in this case equal to the number of trials where the random classifier outperformed the system. Under the null hypothesis that the system performs equally well as chance, the test statistic follows a binomial distribution with parameter $p = 0.5$. When I carry this test for the scores obtained on the first test set by either version of the system, I find that *none* of the 20,000 random trials produced an F-measure score better than 40.66 or 48, the values obtained by the system. The probability of this happening under the null hypothesis is 0.5^{20000}, or approximately $2.5 \cdot 10^{-6021}$! Consequently, I reject the null hypothesis and conclude that the system clearly outperforms random classification.

4.2 Results on nouns from the Wall Street Journal corpus

I have extended the experiments on adjectives described in the previous section to several collections of nouns. Following the organization of the presentation of grouping results for the adjective test sets (Section 4.1), I first discuss how the collections of nouns on which the system was tested were selected. Then, I show actual partitions of these sets according to semantic relatedness between the words, constructed by the first stage of the system. I conclude by presenting the results of the evaluation procedure of Sections 3.2–3.4 and 3.6 when the latter is applied to the retrieved semantic groups of nouns.

4.2.1 Selecting the tests sets of nouns

To construct the first test set of nouns, I started with the list of nouns that appear in the 1987 Wall Street Journal corpus (see Section A.2 in the Appendix) 300 times or more. I selected from this list 44 nouns that I felt were organized in clearly defined semantic groups. In this way, the collection of selected nouns exhibits more semantic coherence than a random collection of nouns. Needless to say, the operation of choosing which nouns to include in this set was performed before the clustering experiments involving these nouns. The 44 nouns in the first test set are listed in Figure 4.15.

The second test set of nouns was constructed with the opposite approach: instead of manually selecting an interesting subset of frequent words, I included in this set the 50 most frequent nouns from the 1987 Wall Street Journal corpus.[11] This second test set is shown in Figure 4.16. There is significant overlap between these two test sets, as they share 22 words (half of the words present in the smaller of the two sets). Note that there is significantly

[11] The less frequent word in this set was *change*, appearing 2,664 times in the corpus.

acquisition	decline	gain	officer	share
analysis	director	growth	official	stock
business	earning	judge	president	study
company	employee	jury	profit	success
corporation	executive	law	purchase	supplier
cost	expense	liability	quarter	week
court	failure	loss	regulation	worker
day	fee	manufacturer	report	year
debt	firm	month	sale	

Figure 4.15: Noun test set 1: Nouns from the 1987 Wall Street Journal corpus occuring 300 times or more.

agreement	earning	issue	people	sale
analyst	executive	law	plan	service
bank	firm	level	policy	share
business	fund	loss	president	stock
change	gain	manager	price	system
company	group	market	problem	thing
concern	growth	month	product	time
cost	income	officer	profit	way
day	interest	official	program	week
debt	investor	operation	rate	year

Figure 4.16: Noun test set 2: The 50 most frequent nouns in the 1987 Wall Street Journal corpus.

less semantic coherence between the members of this set, when compared to the first one; this characteristic, which is expected given the selection criteria for the two sets, is reflected on the increased difficulty and reduced agreement that both the system and the human evaluators have when clustering the words in the second set.

Finally, I selected two additional collections of nouns, again using frequency criteria and drawing the words from the 1987 Wall Street Journal corpus. The first of these additional test sets contains all nouns appearing 200 times or more in the corpus; 811 nouns satisfy

adjustment	delay	justice	reaction	suit
advantage	demand	lawmaker	reduction	swing
answer	difficulty	leadership	regime	tax
attorney	document	letter	relationship	tool
burden	drive	limit	restaurant	town
carrier	freedom	method	role	trend
choice	ground	night	ruling	venture
communication	help	planner	salary	version
control	hope	question	secretary	war
couple	interpretation	ratio	spending	wave

Figure 4.17: A sample of 50 words from noun test set 3: All 811 nouns occuring in the 1987 Wall Street Journal corpus 200 times or more. Nouns that also belong to test sets 1 or 2 were excluded from the sampling.

assignment	dose	liberalization	pipeline	resignation
backing	drama	lobbyist	pool	sex
booster	factory	merit	port	singer
circuit	fix	momentum	presentation	spread
coal	flier	museum	pricing	strife
commander	grip	negotiator	probe	tariff
connection	headline	neighborhood	proof	territory
constituency	illness	objective	rank	thrust
contest	ingredient	overhaul	reality	tomorrow
credibility	leeway	pill	reporting	yard

Figure 4.18: A sample of 50 words from noun test set 4: All 1,893 nouns occuring in the 1987 Wall Street Journal corpus 50 times or more. Nouns that also belong to test sets 1, 2, or 3 were excluded from the sampling.

this criterion. I lowered the frequency threshold to 50 for the fourth test set, collecting 1,893 nouns. It necessarily follows from the construction criteria that test set 3 is included in test set 4; test set 2 is also included in test set 3, as the least frequent noun in set 2 (*change*) occurs more than 200 times in the corpus. Figures 4.17 and 4.18 list 50 randomly selected words from each of test sets 3 and 4 respectively, with the constraint that members of the included test sets were not considered when sampling from a larger test set.

1. acquisition purchase sale	7. business failure success
2. director executive judge officer official president	8. debt liability
3. cost expense fee	9. decline earning gain growth loss profit
4. court jury share stock	10. law regulation
5. day month quarter week year	11. employee worker
6. analysis report study	12. company corporation firm manufacturer supplier

Figure 4.19: Partition with 12 clusters produced by the system for the noun test set 1 of Figure 4.15.

4.2.2 Results and evaluation on the noun test sets

I applied the algorithms of Chapter 2 to the four collections of nouns described above. This procedure generated multiple classifications of the nouns in each set into semantic groups, according to how many groups were specified for each partition. Figure 4.19 lists the partition obtained when the words of the first test set were separated into 12 groups, while Figures 4.20 and 4.21 show the results for the second test set of nouns at two different levels of detail, when 14 and 18 semantic groups are produced, respectively. I do not list results for the third and fourth test sets of 811 and 1,893 nouns respectively, mainly because of the large size of these results. The available data is also more sparse for these nouns, and, as a consequence, they are harder for the system to classify; furthermore, it is not possible to find volunteer evaluators who would create model partitions for these large word sets.

I asked several colleagues at Columbia University to independently construct their own

1. change problem	8. law policy
2. earning gain growth income loss profit sale	9. business debt interest operation
3. fund product	10. cost level price rate
4. bank company concern firm	11. service system
5. agreement plan program	12. thing way
6. analyst investor market	13. executive group manager officer official people president
7. issue share stock	14. day month time week year

Figure 4.20: Partition with 14 clusters produced by the system for the noun test set 2 of Figure 4.16.

1. bank company concern firm	11. analyst investor market
2. law policy	12. month week year
3. product	13. group president
4. service system	14. cost level price rate
5. thing way	15. day time
6. agreement plan program	16. executive manager officer official
7. business interest operation	17. share stock
8. people	18. earning gain growth income loss profit sale
9. change problem	
10. debt fund issue	

Figure 4.21: Partition with 18 clusters produced by the system for the noun test set 2 of Figure 4.16.

semantic groups for the nouns in the first two test sets.[12] In this way, I obtained 12 models

[12] I thank David Evans, Hongyan Jing, Min-Yen Kan, Judith Klavans, Kathy McKeown, Shimei Pan, Dragomir Radev, Carl Sable, Barry Schiffman, James Shaw, and Eric Siegel, who provided such models of

1. day month quarter week year
2. director employee executive judge officer official president worker
3. acquisition analysis business company corporation court failure firm jury law manufacturer regulation report share stock study supplier
4. cost debt decline earning expense fee gain growth liability loss profit purchase sale success

Figure 4.22: Partition with 4 clusters produced by the system for the noun test set 1 of Figure 4.15. Compare with Figure 4.19, where the same words are placed into three times as many clusters and thus more detailed divisions between the semantic groups become apparent.

---

for each of these test sets. The models are used for the evaluation of the system's results, and also provide some guidance about the appropriate number of semantic groups in each test set. The average number of clusters in the human-prepared groupings was 12.42 for the first test set and 18.67 for the second test set, while the heuristics of Section 2.7.3 suggest for this number ranges of 15 to 18, and 17 to 21, respectively. Thus, it seems that the humans use somewhat fewer classes when they separate the first test set of nouns in semantic groups than when they process collections of adjectives of similar size. But this observation is based on limited data and may be due to the words in that particular test set. I have elected to present automatically constructed groupings with approximately as many semantic groups as the average in the human-prepared models; in any case, groupings with widely different numbers of classes represent different organizations of the words where the granularity of the distinctions between groups varies. To further support this point, I show in Figure 4.22 the automatically retrieved classification of the words from test set 1 into just four classes (i.e., at a high level of generality). The groups of Figure 4.22 aggregate

---

semantic affinity for nouns.

closely related groups of more detailed partitions, and correspond to very broad divisions of the words. A comparison between Figures 4.19 and 4.22 indicates that cluster 5 in the former figure (nouns describing periods of time) is the same as cluster 1 in the latter figure; clusters 2 and 11 in Figure 4.19 (nouns describing people) are joined together in cluster 2 of Figure 4.22; clusters 4, 6, 10, and 12 in Figure 4.19 (nouns describing mostly types of companies and reports) are aggregated into cluster 3 of Figure 4.22; clusters 3, 8, and 9 of Figure 4.19 (nouns describing changes in income or size) are incorporated into cluster 4 of Figure 4.22; and only the six members of clusters 1 and 7 in Figure 4.19 are distributed between groups 3 and 4 in Figure 4.22.

I applied the evaluation methodology of Sections 3.2–3.4 to the automatically constructed partitions of the words shown in Figures 4.19–4.21, and to similar partitions produced for the same words with different values of the number of groups parameter. In all cases, the human-prepared groupings of the words were used collectively as the reference standard. The resulting scores for the various evaluation metrics of Section 3.3 before any normalization operations are given in Table 4.7, while Table 4.8 shows the modified scores after the score transformations of Section 3.6. In order to interpret these scores, I have estimated upper bounds for the system's performance by computing the average value of the evaluation measures when each human-prepared model is scored against the other human-prepared models, in a manner similar to what I did for adjectives in Section 4.1.4. These upper bounds are also listed in Table 4.7.

From Tables 4.7 and 4.8 I observe that the performance of the automatic word grouping subsystem is high, and close to the performance of individual humans when the latter are

	Noun test set	Number of clusters	Recall	Precision	Fallout	F-measure
System	Test set 1	10	59.80	49.18	5.31	53.97
		11	59.47	56.33	3.96	57.85
		12	57.46	60.56	3.21	58.97
		13	52.56	63.44	2.60	57.49
		14	50.89	65.66	2.29	57.34
	Test set 2	12	49.80	41.67	5.17	45.37
		14	45.96	46.23	3.96	46.09
		16	32.64	40.56	3.54	36.17
		18	31.36	43.44	3.02	36.43
		20	31.46	45.83	2.75	37.31
Humans	Test set 1	12.42	65.31	66.78	3.00	64.88
	Test set 2	18.67	48.59	52.06	3.85	47.76

Table 4.7: Evaluation scores for the noun groupings produced by the system that were listed in Figures 4.19–4.21, as well as other groupings on the same noun sets but with different values for the number of clusters parameter. The average scores of the human-prepared models, when each of them is evaluated against the other human-prepared models, are included for comparison. The average number of clusters in the 12 human-constructed models for each noun test set is listed in the "number of clusters" column for the rows evaluating human-made groupings. None of the score adjustments of Section 3.6 has been applied to the listed scores.

compared to other humans. This applies to both test sets, but especially to the second one. In fact, when the words in the second test set are classified, the system outperforms (before score normalizations are considered) several of the judges when both are compared to the remaining judges. The system achieves a maximum F-measure score of 46.74 on this test set, when nine clusters are used. Five of the twelve human-constructed model solutions (employing between 8 and 25 numbers of clusters) receive agreement scores with the remaining eleven models smaller than 46.74.

The second test set was constructed by taking the 50 most frequent nouns in the corpus, and thus exhibits less semantic coherence than the first test set where the semantic

	Noun test set	Number of clusters	Base score	I	II	III	I and III	I, II, and III
System	Test set 1	10	53.97	83.18	87.57	61.74	87.04	96.67
		11	57.85	89.16	88.44	66.52	93.78	97.44
		12	58.97	90.89	88.53	67.81	95.60	97.55
		13	57.49	88.61	87.72	66.01	93.06	96.99
		14	57.34	88.38	88.53	65.57	92.44	97.62
	Test set 2	12	45.37	95.00	81.47	55.92	*	96.30
		14	46.09	96.50	80.72	56.97	*	95.33
		16	36.17	75.73	74.49	45.84	84.89	89.09
		18	36.43	76.28	75.27	44.82	83.00	89.43
		20	37.31	78.12	77.51	44.93	83.20	91.48
Humans	Test set 1	12.42	64.88	100.00	91.40	70.92	100.00	100.00
	Test set 2	18.67	47.76	100.00	85.98	54.00	100.00	100.00

Table 4.8: Normalized F-measure scores for the noun groupings for which unnormalized scores were given in Table 4.7. The evaluated partitions include those listed in Figures 4.19–4.21, as well as other groupings on the same noun sets but with different values for the number of clusters parameter. The following normalization procedures were applied: (I) Dividing with the average score achieved on the same problem instance by the humans, which, as discussed in Section 3.6, corrects for discrepancies among the models; (II) applying the procedure of Section 3.6.1 to correct for multiple false positives and false negatives resulting from one wrong decision; and (III) applying the procedure of Section 3.6.2 to avoid counting paired false positives and false negatives twice. The normalized average scores of the human-prepared groupings, when each of them is evaluated against the other human-prepared groupings, are included for comparison; the number of clusters listed in these rows is the average value across all 12 human-made models for the current noun test set. Note that any column involving normalization (I) has necessarily a value of 100% for the human-constructed models. Transformation (II) may raise the F-measure score above the average obtained by the humans when the evaluated partition has fewer clusters than the human-made models; this produces a score above 100% if transformation (I) is subsequently applied. Such cases where the normalized score is higher than that of the humans are marked with * in the table.

relationships between its members were a consideration during its construction. The difference in semantic coherence is reflected in the lower scores achieved by both the system and the humans on this second test, which in turn indicate significantly less agreement between the human evaluators. Not only do the semantic relationships identified by the human evaluators differ significantly from model to model for the second test set, but also does the number of clusters that each model employs. The human-prepared models have between 8 and 17 groups for the first test set of nouns and between 8 and 28 groups for the second test set. The ratio of standard deviation (square root of variance[13]) to the mean for the numbers of groups in the human models is $\sqrt{6.9924}/12.4167 = 0.2130$ for the first test set, but $\sqrt{41.8788}/18.6667 = 0.3467$ for the second test set.[14] Several of the evaluators expressed their difficulty with clustering the words in the second test set compared to the same task for the first test set, a situation parallel to what was observed for adjective test sets 2 and 3, where the former test set contained more ambiguous words (see Sections 4.1.1 and 4.1.4).

[13] I use an unbiased estimator of population variance, namely $\mathrm{var}(\mathbf{x}) = \sum_i (\mathbf{x}_i - \overline{\mathbf{x}})^2/(|\mathbf{x}|-1)$, rather than the biased maximum likelihood estimator; see [Huber 1981].

[14] The ratio of the square root of variance to the mean is an objective measure of variability in the data, as the variance itself is linearly related to the square of the mean; multiplying all data values by α multiplies the variance by α^2.

Chapter 5

The importance of linguistic knowledge in the statistical word grouping system

Throughout the previous three chapters I presented the algorithms used in the first stage of the system and results from its operation. This first stage separates words into semantically related groups on the basis of statistical co-occurrence evidence, but linguistic knowledge is used to increase the quantity and improve the quality of the extracted evidence, to filter and transform this evidence, and to supplement it with linguistically based negative examples. This principle of cooperation between statistical learning techniques and shallow linguistic knowledge applies to the stages that subsequently analyze the semantic groups as well. The remaining stages of the system, which are described in detail in Chapters 6–8, operate more or less on the basis of relationships extracted from the text, whose relevance to the problem we are solving is linguistically justified; but again, statistical learning techniques are used to establish this relevance quantitatively and to appropriately weigh the contributions of different sources to the overall result.

Historically, pure statistical, corpus-based approaches were the ones first applied to natural language processing problems in general and lexical knowledge acquisition problems in particular. Such efforts took place very early: in the 50's and early 60's a considerable amount of research work was under way addressing problems such as machine translation and lexicon/thesaurus construction with purely statistical methods. However, the lack of sophisticated statistical models and powerful hardware led to rather disappointing results, and the statistical approach was largely abandoned in the 70's in favor of knowledge-based, AI approaches.

Yet, pure knowledge-based approaches, which use knowledge collected by human experts and entered in the system as an external source, also seem insufficient for providing an adequate solution to natural language problems. As discussed in Section 1.4, such systems are effective when the domain and the vocabulary are tightly controlled, but fail to scale up in more general settings where knowledge acquisition becomes a major bottleneck. When this limitation of current knowledge-based systems became apparent, interest in statistical methods was renewed in the mid- and late 80's, and continues unabated until now. This renaissance of statistical methods was particularly helped by recent successes of such "knowledge-poor" methods in problems such as part-of-speech tagging [Church 1988; Kupiec 1992; Cutting *et al.* 1992; Charniak *et al.* 1993; Merialdo 1994] and speech recognition [Waibel and Lee 1990].

Once more, however, systems based on statistical methods alone have not been totally successful in solving most of the natural language processing problems they addressed. Machine translation is a good example of an area where pure statistical methods were initially adopted enthusiastically as sufficient for the task, but have not yet produced a fully adequate solution. While they were viewed as the means for the full solution to the problem in 1989, five years later some of the pioneers of statistical work on this problem viewed them as useful aids for human translators, a considerable downgrading of the claim.

Consequently, a number of researchers have turned to possible ways of combining linguistic, hand-encoded knowledge with statistical techniques as a means to improve overall performance, since it is reasonable to expect that the combined approach will potentially offer significantly better performance over either methodology alone. The interaction of statistical and linguistics-based components in a system is a topic of considerable current interest; a recent ACL workshop ("The Balancing Act", July 1994) and subsequent book [Klavans and Resnik 1996] were devoted exclusively to this topic. However, when we consider such a hybrid system, several questions arise. Perhaps the most pressing ones are whether the linguistic knowledge actually helps in improving performance, and if so, whether the improvement is worth the effort needed to implement the linguistic modules and incorporate them in the system. In most cases answers to these questions, if supplied at all, are derived from intuitive beliefs and anecdotal evidence, rather than from a rigorous, quantitative comparison.

To answer these two questions in the context of the particular system described in the thesis[1], I considered several different types of linguistic knowledge which were introduced in

[1] But see Section 5.5 for a discussion of why the results obtained with this system should carry over to

the first stage of the system, and examined less sophisticated alternatives for them. I view the word grouping subsystem as a basic statistical system plus a collection of linguistics-based *modules*. These modules can be ignored during system operation, or replaced with others that use less sophisticated linguistic models or do not access any form of shallow linguistic knowledge. In this way, I can contrast the performance of the system with and without each such module and thus quantitatively determine (using the evaluation methodology of Chapter 3) whether each module contributes to the system's performance in a positive or negative direction and how important each such contribution is. The word grouping system is used as the testbed for these comparative experiments; several test sets of adjectives (see Section 4.1) were analyzed across the various combinations of the basic statistical system and linguistic modules.

In what follows, I present several different types of shallow linguistic knowledge that can be, and have been, efficiently introduced into the word grouping system. I evaluate the performance of the system with and without each such feature, obtaining an estimate of each feature's positive or negative contribution to the overall performance. By matching cases where all system parameters are the same except for one feature, I assess the statistical significance of the differences found. Furthermore, a statistical model of the system's performance in terms of the active features for each run offers a view of the contributions of features from a different angle, contrasting the significance of linguistic features (or other modeled system parameters) against each other. I also assess the overall effect linguistic knowledge has on the word grouping system. The results of these experiments and the

---

many other statistical language learning systems.

associated analyses of statistical significance are presented in Section 5.3. I then proceed to discuss the cost of incorporating the linguistic knowledge modules into the statistical system, and present arguments in favor of the relevance of these results to statistical approaches for other natural language processing problems. I conclude this chapter by summarizing the conclusions that can be drawn from this empirical, corpus-based comparative study.

5.1 Linguistic features and alternative values for them

I have identified several sources of linguistic knowledge that can be incorporated in the word grouping system, augmenting the basic statistical component. Each such source represents a parameter of the system, i.e., a feature that can be present or absent or, more generally, can take a value from a predefined set. I selected features that can be efficiently computed in a completely automatic way for unrestricted text and do not require extensive amounts of knowledge to be available to the system. Almost all of these features can be generalized to other applications as well, as I discuss in Section 5.5. In this section I present first one of these parameters that can take several values, namely the method of extracting data from the corpus, and then several other binary-valued features.

5.1.1 Extracting data from the corpus

When the word clustering system partitions adjectives in groups of semantically related ones, it determines the distribution of related (modified) nouns for each adjective and eventually the similarity between adjectives from pairs of the form ⟨adjective, modified noun⟩ that have been observed in the corpus. Direct information about semantically unrelated adjectives (in the form of appropriate adjective-adjective pairs) can also be collected from the corpus. Therefore, a first parameter of the system and a possible dimension for comparisons is the method employed to identify such pairs in free text. This is hardly a unique feature of the present system: all word-based statistical systems must first collect data from the corpus about the words of interest, on which the subsequent statistics operate.[2]

There are several alternate models for this task of data collection, with different degrees of linguistic sophistication. A first model is to use no linguistic knowledge at all[3]: we collect for each adjective of interest all words that fall within a window of some predetermined size. Naturally, no negative data (adjective-adjective pairs) can be collected with this method. However, the method can be implemented easily and does not require the identification of any linguistic constraints so it is completely general. It has been used for diverse problems such as machine translation and sense disambiguation [Gale *et al.* 1992b; Schütze 1992].

A second model is to restrict the words collected to the same sentence as the adjective of interest and to word class(es) that we expect on linguistic grounds to be relevant

---

[2] Although frequently details of the statistical model employed receive more consideration.

[3] Aside from the concept of a word, which is usually approximated by defining any string of characters separated by white space as a word.

to adjectives. Since nouns are expected to be the only other such relevant class of major importance for adjectives (Section 2.2.3), we collect all nouns in the vicinity of an adjective without leaving the current sentence. We assume that these nouns have some relationship with the adjective and that semantically different adjectives will exhibit different collections of such nouns. This model requires only part-of-speech information (to identify nouns) and a method of detecting sentence boundaries. It uses a window of fixed length to define the neighborhood of each adjective. Such a model incorporates minimal linguistic knowledge, namely in determining what constitutes the informative class(es) of words collected (nouns in our problem). Again, negative knowledge such as pairs of semantically unrelated adjectives cannot be collected with this model. Nevertheless, it has also been widely used, e.g., for collocation extraction [Smadja 1993] and sense disambiguation [Liddy and Paik 1992].

A third model uses a simple linguistic rule to identify pairs of interest that is even more restrictive and informative than the "nouns in vicinity" approach. Since I am interested in nouns modified by adjectives, such a rule is to collect a noun whenever it immediately follows an adjective, assuming that this implies a modification relationship. Pairs of consecutive adjectives can also be collected.

Up to this point I have successively restricted the collected pairs on linguistic grounds, so that less but cleaner data is collected. For the fourth model, I extend the simple rule given above, using linguistic information to catch more valid pairs without sacrificing accuracy. I employ a pattern matcher that retrieves any sequence of one or more adjectives followed by any sequence of zero or more nouns. These sequences are then analyzed with heuristics based on linguistics (see Section A.4.3 in the Appendix) to obtain pairs. For example, it can be shown that all adjectives in such a sequence must be semantically unrelated, and that it is best to attach all the adjectives to the final noun.

The regular expression and pattern matching rules of the previous model can be extended further, forming a grammar for the constructs of interest. This approach can detect more pairs, and at the same time address known problematic cases not detected by the previous models.

I implemented the above five data extraction models, using typical window sizes for the first two methods (50 and 5 on each side of the window respectively) which have been found useful in other problems before. For the fifth model, I developed a finite-state grammar for noun phrases which is able to handle both predicative and attributive modification of nouns, conjunctions of adjectives, adverbial modification of adjectives, quantifiers, and apposition of adjectives to nouns or other adjectives.[4] A detailed description of this grammar and its implementation is presented in Sections A.4.2–A.4.4 of the Appendix.

Unfortunately, the resources required to perform the comparative tests for the first model were too great (e.g., 12,287,320 word pairs in a 151 MB file were extracted for the 21 adjectives in the smallest test set I use in this comparative analysis) so I dropped that model from further consideration and use the second model as the baseline of minimal linguistic knowledge. Other researchers have also reported similar problems of excessive resource demands with the "collect all neighbors" model [Gale *et al.* 1992b].

[4] For efficiency reasons I did not consider a more powerful formalism, such as a context-free grammar augmented with features.

5.1.2 Other linguistic features

In addition to the data extraction method, I identified three other areas where linguistic knowledge can be introduced in the system. First, we can employ morphology to convert plural nouns to the corresponding singular ones and adjectives in comparative or superlative degree to their base form. Almost all adjectives and nouns that appear in multiple forms have no semantic difference from their base form except for the number or degree feature. This conversion combines counts of similar pairs, thus raising the expected and observed frequencies of each pair in any statistical model. I developed a morphology component that produces the singular form of nouns using rules plus a large table of exceptions. For adjectives, a set of rules is again employed but because of the vowel in the suffix *-er* or *-est*, many base forms look plausible without a lexicon (e.g., *bigger* could have been produced from *big*, *bigg*, or *bigge*). I solve this problem by counting the occurrences of each candidate form in the corpus and selecting the one with the highest frequency.[5] Section A.5 in the Appendix gives more details about the morphology module along with the specialized rules and exceptions used during the conversion process.

Another potential application of linguistic knowledge is the use of a spelling checking procedure, combined with a word list, to eliminate typographical errors from the corpus. Such errors can produce wrong estimates for the frequencies of modified nouns for an adjective, but most importantly introduce "unique" nouns appearing only with one adjective, skewing the comparison of noun distributions. I implemented this component using the

[5]Typically, only one of the candidate forms will have non-zero frequency; an exception to this may occur because of typographical errors.

UNIX *spell* program and associated word list, with extensions for hyphenated compounds; see Section A.5 in the Appendix for a detailed description. Unfortunately, since a fixed and domain-independent lexicon is used during this process, some valid but overspecialized words may be discarded too.

Finally, we can use additional sources of knowledge which supplement the primary similarity relationships and are justified on linguistic grounds. One such knowledge source, namely instances of syntagmatic relationships that indicate semantic unrelatedness (types 2 and 3 in Section 2.1), is relevant to the grouping task and has been incorporated in the system. In this comparison study I consider the availability of the negative examples offered by these adjective-adjective pairs as an additional linguistic feature.

5.2 The comparison experiments

In the previous section I identified four parameters of the system, the effects of which we want to analyze. But in addition to these parameters that can be varied in a directly controlled manner and have predetermined possible values, several other variables can affect the performance of the system.

First, the system's performance depends on the adjective set that is to be clustered. Presumably, variations in the adjective set can be modeled by several parameters, such as the size of the set, the number of semantic groups in it, and the strength of semantic relatedness among its members, plus several parameters describing the properties of the adjectives in the set in isolation, such as frequency, specificity, etc.

A second variable that affects the clustering is the corpus that is used as the main knowledge source, through the observed co-occurrence patterns. Again the effects of different corpora can be separated into several factors, e.g., the size of the corpus, its generality, the genre of the texts, etc.

Since in these experiments I am interested in quantifying the effect of the linguistic knowledge in the system, or more precisely of the linguistic knowledge that I can explicitly control through the four parameters discussed in the previous section, I did not attempt to model in detail the various factors entering the system as a result of the choice of adjective set and corpus. However, I am interested in measuring the effects of the linguistic parameters in a wide range of contexts, and in correlating these effects with variables originating from the choice of corpus and adjective set. For example, I would want to be able to detect that the linguistic parameter "morphology" is significant for small corpora but not for large ones, if that were the case. Therefore, I included in the model two additional parameters, representing the corpus and the adjective set used.

I used the 1987 Wall Street Journal articles from the ACL Data Collection Initiative as the corpus (see Section A.2 in the Appendix). I selected four sub-corpora to study the relationship of corpus size with linguistic feature effects: sub-corpora of 330,000 words, 1 million words, 7 million words, and 21 million words (the latter consisting of the entire 1987 corpus) were selected as representative. Each selected sub-corpus contained the selected sub-corpora of smaller sizes, and was constructed by sampling across the whole range of the entire corpus at regular intervals. Since I use subsets of the same corpus, I am essentially modeling the corpus size parameter only.

For each corpus, I analyzed three different sets of adjectives, listed in Figures 4.1–4.3 on pages 170–171. As discussed in Section 4.1.1, the first of these adjective test sets was selected from a similar corpus and contains 21 words of varying frequencies that all associate strongly with a particular noun (*problem*). The second set (43 adjectives) was selected with the constraint that it contain high frequency adjectives (more than 1,000 occurrences in the 21 million word corpus). The third set (62 adjectives) satisfies the opposite constraint containing adjectives of relatively low frequency (between 50 and 250). Taken together, these three sets exemplify several different choices for the parameters of problem instance size, semantic relatedness between the words, and frequency and specificity of the clustered words.

By including the parameters "corpus size" and "adjective set", I have six parameters that I can vary in the experiments. Any remaining factors affecting the performance of the system are modeled as random noise[6], so statistical methods are used to evaluate the effects of the selected parameters. The six chosen parameters are completely orthogonal, with the exception that parameter "negative knowledge" must have the value "not used" when parameter "extraction model" has the value "nouns in vicinity". In order to avoid introducing imbalance in the analysis, I constructed a complete designed experiment [Hicks

[6]Including some limited in extent but truly random effects from the non-deterministic clustering algorithm of Section 2.7.1.

1982] for all their $(4 \times 2 - 1) \times 2 \times 2 \times 4 \times 3 = 336$ valid combinations.[7]

For each experimental run, corresponding to a particular combination of the six modeled parameters, I evaluated the partitions produced by the system using the methodology of Chapter 3. In order to be able to compare different versions of the system (with different combinations of linguistic modules active), I use the F-measure score as the evaluation metric, without applying any of the score adjustments of Section 3.6 to account for structural constraints and the interdependence of the decision points in the evaluated partitions. This measure, defined in equation (3.5) in Section 3.3 (page 145), combines precision and recall in a single number with equal weight. In addition, since the correct number of groupings is something that the system cannot yet determine (see the discussion in Section 2.7.3), and, incidentally, something that human evaluators disagree about, I run the system for the five cases in the range -2 to $+2$ around the average number of clusters employed by the humans and average the results. This smoothing operation prevents an accidental high or low score being reported when a small variation in the number of clusters produces very different scores.

As noted in Section 3.5, the scores reported should not be interpreted as linear percentages. In other words, a score of 40 is not just twice as good as a score of 20, and going from 30 to 40 is much harder than going from 20 to 30. The latter is true for most applications, but the problem of interpreting the scores is exacerbated in our context because of the structural constraints imposed by the clustering and the presence of multiple models.

[7]Recall that a designed experiment is *complete* when at least one trial, or *run*, is performed for every valid combination of the modeled predictors.

Even the best clustering that could be produced would not receive a score of 100, because of disagreement among humans on what is the correct answer. Having cautioned the reader on this characteristic of the evaluation scores, I do not apply to them the normalization transformations of Section 3.6, since for the purpose of comparing across system variants the original F-measure scores serve just as well. However, to clarify the meaning of the scores, I accompany them with lower and upper bounds for each adjective set I examine. These bounds are obtained by the performance of a system that creates random groupings (averaged over many thousands of runs) and by the average score of the human-produced partitions when evaluated against the other human models respectively (cf. Sections 3.5.2 and 4.1.4). In particular, applying the same evaluation method to score each model constructed by humans for the three adjective sets used in the comparative study against the other human-constructed models leads to an average score of 60.44 for the human evaluators.

5.3 Experimental results

5.3.1 Average effect of each linguistic parameter

Presenting the scores obtained in each of the 336 individual experiments performed, which correspond to all valid combinations of the six modeled parameters, is both too demanding in space and not especially illuminating. Instead, I present several summary measures. I measured the effect of each particular setting of each linguistic parameter of Section 5.1 by

Parameter	Value	Average score
Extraction model	Parsing	30.29
	Pattern matching	28.88
	Observed pairs	27.87
	Nouns in vicinity	22.36
Morphology	Yes	28.60
	No	27.53
Spell checking	Yes	28.12
	No	28.00
Use of negative knowledge	Yes	29.40
	No	28.63

Table 5.1: Average F-measure scores (without any normalization) for each value of each linguistic feature.

averaging the scores obtained in all experiments where that particular parameter had that particular value. In this way, Table 5.1 summarizes the differences in the performance of the system caused by each parameter. Because of the complete design of the experiment, each value in Table 5.1 is obtained from a collection of runs that are identical to the runs used for estimating the other values of the same parameter except for the difference in the parameter itself.[8]

Table 5.1 shows that there is indeed improvement with the introduction of any of the proposed linguistic features, or with the use of a linguistically more sophisticated extraction model. To assess the statistical significance of these differences, I compared each run for a particular value of a parameter to the corresponding identical (except for that parameter) run for a different value of the parameter. Each pair of values for a parameter produces in this way a set of paired observations. On each of these sets, I performed a sign test

[8] The slight asymmetry in parameters "extraction model" and "negative knowledge" is accounted for by leaving out non-matching runs.

Parameter tested	Test: First value	Test: Second value	Comparisons	First value better than second	Probability
Extraction model	Parsing	Pattern matching	96	64	0.0014
	Parsing	Observed pairs	96	66	0.0003
	Parsing	Nouns in vicinity	48	42	10^{-7}
	Pattern matching	Observed pairs	96	61	0.0104
	Pattern matching	Nouns in vicinity	48	41	$6.24 \cdot 10^{-7}$
	Observed pairs	Nouns in vicinity	48	36	0.0007
Morphology	Used	Not used	168	107	0.0005
Spell checking	Used	Not used	168	94	0.1425
Negative knowledge	Used	Not used	144	97	$3.76 \cdot 10^{-5}$

Table 5.2: Statistical significance analysis of the difference in performance offered by each linguistic feature. The rightmost column gives the probability that the same or more extreme results would have been obtained by chance if a change between the two contrasted parameter values had no effect on system performance.

[Gibbons and Chakraborti 1992] of the null hypothesis that there is no real difference in the system's performance between the two values, i.e., that any observed difference is due to chance. I counted the number of times that the first of the two compared values led to superior performance relative to the second, distributing ties equally between the two cases following the standard practice in classifier induction and evaluation [Duda and Hart 1973]. Under the null hypothesis, the number of times that the first value leads to better performance follows the binomial distribution with parameter $p = 0.5$. Table 5.2 gives the results of these tests along with the probabilities that the same or more extreme results would be encountered by chance. We can see from the table that all types of linguistic knowledge except spell checking have a beneficial effect that is statistically significant at, or below, the 0.1% level.

5.3.2 Relative importance of the linguistic features

In order to measure the significance of the contribution of each linguistic feature relative to the other linguistic features, I fitted a linear regression model [Draper and Smith 1981] to the data. I use the six parameters of the experiments as the predictors, and the F-measure score of the corresponding clustering[9] as the response variable. In such a model the response R is assumed to be a linear function (weighted sum) of the predictors V_i, i.e.,

$$R = \beta_0 + \sum_{i=1}^{n} \beta_i V_i \tag{5.1}$$

where V_i is the i-th predictor and β_i is its corresponding weight.[10]

We can estimate appropriate values for the coefficients (weights) β_i from a set of samples where each sample S_j, $j = 1, 2, \ldots, m$, consists of a vector of the predictor variables $(V_{1j}, V_{2j}, \ldots, V_{nj})$ and the associated response R_j. Then, the linear model for R specifies that for each sample S_j

$$R_j = \left(\beta_0 + \sum_{i=1}^{n} \beta_i V_{ij}\right) + \varepsilon_j$$

[9] Averaged over five adjacent values of the number of clusters parameter, but without any of the adjustments of Section 3.6, as explained in Section 5.2.

[10] Such a model is appropriate for comparative purposes, although extrapolating response values for prediction outside the range of predictor values used in the fitting may give incorrect results. For example, the coefficients in Table 5.3 cannot be used to predict the score when the corpus is significantly smaller than 330,000 words or larger than 21 million words.

where ε_j is a random variable that models the estimation error when we predict R from the V_i's for sample S_j using the estimator $\hat{R}$ given by equation (5.1). In order to select appropriate values for the coefficients β_i, I define the *sum of squared residuals* (RSS) quantity,

$$\begin{aligned} RSS &= \sum_{j=1}^{m} \left(R_j - \hat{R}_j\right)^2 = \\ &= \sum_{j=1}^{m} \left(R_j - \left(\beta_0 + \sum_{i=1}^{n} \beta_i V_{ij}\right)\right)^2 \end{aligned}$$

and select the values of the coefficients that minimize the above expression. The problem of selecting these coefficients in a way that minimizes RSS (i.e., the problem of determining the *least squares fit*) has been well studied in statistics, and there exist efficient and accurate methods for this task.[11] The benefit of choosing the minimization of RSS as the criterion for determining the β_i's becomes apparent if we make the reasonable distributional assumption that the error terms ε_j are identically distributed independent random variables following the normal distribution with mean 0 and the same (finite but unspecified) variance. In a linear model, both the coefficients β_i and the predictions for the response on new data R are linear combinations of these error terms and the values of the predictors and the response used for the estimation of the β_i's. Consequently, these random variables are normally distributed and both the estimates for the coefficients $\hat{\beta}_i$ and for the response $\hat{R}$ are unbiased estimators of the corresponding random variables.[12]

Table 5.3 shows the weights found by this fitting process from the experimental data

[11]Briefly, the problem essentially involves the computation of an orthogonal basis for the matrix of observed values for the variables V_i, which in turn can be performed with standard methods such as the Householder decomposition of the matrix of predictors [Golub and van Loan 1989]. Alternatively, other methods, such as singular value decomposition, can be used.

[12]Recall that an estimator $\hat{X}$ for a random variable X is *unbiased* if $\mathbf{E}[\hat{X}] = \mathbf{E}[X]$.

Variable	Weight
Intercept	18.7997
Corpus size (in millions of words)	0.9417
Extraction method (Pairs versus Nouns in vicinity)	5.1307
Extraction method (Sequences versus Nouns in vicinity)	6.1418
Extraction method (Parser versus Nouns in vicinity)	7.5423
Morphology	0.5371
Spell checking	0.0589
Adjective Set (2 versus 1)	2.5996
Adjective Set (3 versus 1)	−11.4882
Use of negative knowledge	0.3838

Table 5.3: Fitted coefficients for the linear regression model that contrasts the effects of various parameters on overall performance for the word grouping subsystem. The given weights for the categorical variables "extraction method" and "adjective set" are contrasts with the base cases of the "noun-in-vicinity" method and adjective set 1 respectively.

collected for all valid combinations of the six parameters that I model. These weights indicate by their absolute magnitude and sign how important each predictor is and whether it contributes positively or negatively to the final result.[13] Numerical values such as the corpus size enter formula (5.1) directly as predictors, so Table 5.3 indicates that each additional million words of training text increases the performance of the system by 0.9417 on the average. For binary features, the weights in Table 5.3 indicate the increase in the system's performance when the feature is present, so the introduction of morphology improves the system's performance by 0.5371 on the average.

Categorical variables that can take a value from a finite (usually small) predefined set are treated in a special manner. Although a superficial way to represent their values is to assign to them successive integer values, this approach is not suitable for direct incorpora-

[13]The scale of values of the predictor variables should also be taken into account when comparisons of their contributions to the overall performance score are made.

tion in the model; the value encoded with "3" is not necessarily three times more significant than the value encoded with "1". In such cases, we encode the different values as *contrasts*, and we include several terms for the variable in the model. In this way the term $\beta_i V_i$ in equation (5.1) corresponding to a discrete variable with r distinct values will be replaced by $r-1$ terms corresponding to the product of the (extended) vector of coefficients $(1, \beta_{i1}, \beta_{i2}, \ldots, \beta_{i(r-1)})$ and an $r \times (r-1)$ matrix encoding the contrasts. Each contrast represents the effect that the encoded value has in the overall response.

Several methods exist for encoding such contrasts: for example, we may use the orthogonal Helmert contrasts, which contrast the second value with the first, the third with the average of the first two, and so on. Or we may sacrifice orthogonality in favor of simplicity, selecting one value as the default or standard one and contrasting all other values with it. In that case, each of the $r-1$ coefficients corresponds to one of the other values and simply gives the effect in the response if the default value is replaced with that value. I adopt this second method for encoding contrasts, because it allows the resulting coefficients to be interpreted clearly and easily.[14]

Consequently, for the two categorical variables "adjective set" and "extraction model" the weights show the change in score for the indicated value in contrast to the base case (adjective set 1 and the case of minimal linguistic knowledge represented by extraction model "nouns in vicinity" respectively). For example, using the finite-state parser instead of the "nouns in vicinity" model improves the score by 7.5423 on the average, while going

[14]Binary variables are a special case of categorical variables for which no explicit contrasts are needed; encoding one value as "1" and the other as "0" and directly using these encodings in equation (5.1) is equivalent to both of the general encoding methods described above.

from adjective set 2 to adjective set 3 decreases the score by $-(-2.5996-11.4882) = 14.0878$ on the average. Finally, the intercept β_0 gives a baseline performance of a minimal system that uses the base case for each parameter; the effects of corpus size are to be added to this minimal system.

From Table 5.3 we can see that the data extraction model has a significant effect on the quality of the produced clustering, and among the linguistic parameters is the most important one. Increasing the size of the corpus also significantly increases the score. The adjective set that is clustered also has a major influence on the score, with rarer adjectives leading to worse clusterings.[15] The two linguistic features "morphology" and "negative knowledge" have less pronounced although still significant effects, while spell-checking offers minimal improvement that probably does not justify the effort of implementing the module and the cost of activating it at run-time.

5.3.3 Overall effect of linguistic knowledge

Up to this point I have described averages of scores or of score differences, taken over many combinations of features that are orthogonal to the one studied. These averages are good for describing the *existence* of a performance difference caused by the different values of each feature, across all possible combinations of the other features. They are

[15] Note that these are average effects, taken over a wide range of different settings for the system. In particular, while the system produces bad partitions of adjective set 3 when the corpus is small, when the largest corpus (21 million words) is used the partitions produced for test set 3 are equal in quality or better to the partitions produced for the other two sets with the same corpus.

	Adjective set 1		Adjective set 2		Adjective set 3	
Random partitions	9.66	(17.90%)	6.21	(9.66%)	3.80	(6.03%)
No linguistic components active	24.51	(45.41%)	38.51	(59.92%)	33.21	(52.66%)
All linguistic components active	39.06	(72.36%)	44.73	(69.60%)	46.17	(73.20%)
Humans	53.98		64.27		63.07	

Table 5.4: Overall effect of linguistic knowledge on the evaluation score. Two versions of the system (with all or none of the linguistic modules active) are contrasted to the performance of a random classifier and of the humans. The scores were obtained on the 21 million word corpus, using a smoothing window of 3 adjacent values of the number of clusters parameter centered at the average value for that parameter in the human-prepared models (9, 15, and 23 clusters for the three adjective sets analyzed). I also show the percentage of the score of the humans that is attained by the random classifier and each version of the system.

---

not, however, representative of the performance of the system in a particular setting of parameters, nor are they suitable for describing the difference in features quantitatively, since they are averages taken over widely differing settings of the system's parameters. In particular, the inclusion of very small corpora drives the average scores down, as I have confirmed by computing averages separately for each value of the corpus size parameter. To give a feeling of how important the introduction of linguistic knowledge is *quantitatively*, I compare in Table 5.4 the results obtained for the full corpus of 21 million words for the two cases of having all or none of the linguistic components active. This table is similar to an augmented version of Table 4.6 on page 190. The scores obtained by a random system that produces partitions of the adjectives with no knowledge except the number of groups are included as a lower bound. These estimates are obtained after averaging the scores of 20,000 such random partitions for each adjective set. The average scores that each human model receives when compared to all the other human models are also included, as an estimate of the maximum score that can be achieved by the system. As discussed in Section 3.5.2, this maximum depends on the disagreement between models for each adjective set. For these

1. deadly fatal	13. communist leftist
2. capitalist socialist	14. astonishing meager vigorous
3. clean dirty dumb	15. catastrophic disastrous harmful
4. hazardous toxic	16. dry exotic wet
5. insufficient scant	17. chaotic turbulent
6. generous outrageous unreasonable	18. confusing misleading
7. endless protracted	19. dismal gloomy
8. plain	20. dual multiple pleasant
9. hostile unfriendly	21. fat slim
10. delicate fragile unstable	22. affordable inexpensive
11. affluent impoverished prosperous	23. abrupt gradual stunning
12. brilliant clever energetic smart stupid	24. flexible lenient rigid strict stringent

Figure 5.1: Partition with 24 clusters produced by the system for the adjective test set 3 of Figure 4.3 using the entire 21 million word corpus and all linguistic modules.

measurements I use a smaller smoothing window of size 3 instead of 5, which is fairer to the system when its performance is compared to the humans. I also show in Figures 5.1 and 5.2 the groupings produced by the system using the entire 21 million word corpus when the words of adjective set 3 are partitioned in 24 groups with all linguistic components active and with no linguistic components active respectively. These two partitions should also be compared with the partition with 23 groups obtained for the same test set and corpus using all linguistic modules, which is shown in Figure 4.9 on page 178.

1. catastrophic harmful	13. brilliant energetic
2. dry wet	14. dual multiple stupid
3. lenient rigid strict stringent	15. hazardous toxic unreasonable unstable
4. communist leftist	16. plain
5. flexible hostile protracted unfriendly	17. confusing
6. abrupt chaotic disastrous gradual turbulent vigorous	18. clever
7. affluent affordable inexpensive prosperous	19. endless
8. outrageous	20. clean dirty impoverished
9. capitalist socialist	21. deadly fatal
10. dismal gloomy pleasant	22. astonishing misleading stunning
11. generous insufficient meager scant slim	23. dumb fat smart
12. delicate fragile	24. exotic

Figure 5.2: Partition with 24 clusters produced by the system for the adjective test set 3 of Figure 4.3 using the entire 21 million word corpus but no linguistic modules.

5.4 Cost of incorporating the linguistic knowledge in the system

The cost of incorporating the linguistics-based modules in the system is not prohibitive. The effort needed to implement all the linguistic modules was about 5 person-months, in contrast with 8 person-months needed to develop the basic statistical system. Most of this time was spent in designing and implementing the finite state grammar that is used for extracting adjective-noun and adjective-adjective pairs (cf. Sections A.4.2–A.4.4 in the

Module	Run-time overhead
Word tokenizer	0 min 58 sec per million words
Part of speech tagger	7 min 56 sec per million words
Finite-state parser	2 min 43 sec per million words
Morphological analysis	3 min 07 sec per million pairs
Spell-checking	3 min 23 sec per million pairs
Using negative examples	1 min 20 sec per million pairs
Statistical system	approximately 15 min to classify 40 adjectives

Table 5.5: Run-time overhead of each module of the word grouping system on a Sun SparcStation 10. The times listed are averages over several different corpora of different sizes (parts of the 1987 Wall Street Journal corpus) and over the three adjective test sets used in the comparative analysis (for modules that depend on the problem instance been analyzed).

Appendix).

Furthermore, the run-time overhead caused by the linguistic modules is not significant. In Table 5.5 I present the average time that each module of the word grouping system takes for processing one million entries (words or pairs of words, as appropriate for the module). As we can see from the table, each linguistic module takes from one to eight minutes on a Sun SparcStation 10 to process a million items, and all except the negative knowledge module need process a corpus only once, reusing the same information for different problem instances (word sets). This should be compared to the approximately 15 minutes needed by the statistical component for grouping about 40 adjectives.

5.5 Generalizing to other applications

In Section 5.3 I showed that the introduction of linguistic knowledge in the word grouping system produces a performance difference, which is not only statistically observable but also quantitatively significant (cf. Table 5.4). I believe that these positive results should also apply to other corpus-based natural language processing systems that employ statistical methods. Many of the linguistic components of the grouping system, including the extraction model that was shown to be the most important linguistic parameter, are not specific to the word grouping problem. They can thus be directly incorporated in systems designed for other problems but essentially following the same basic architecture as the testbed system.

Many statistical approaches share the same basic methodology with the present system: a set of words is preselected, related words are identified in a corpus, the frequencies of words and of pairs of related words are estimated, and a statistical model is used to make predictions for the original words. Across applications, there are differences in what words are selected, how related words are defined, and what kind of predictions is made. Nevertheless, the basic components stay the same. For example, in the adjective grouping application the original words are the adjectives and the predictions are their groups; in machine translation, the predictions are the translations of the words in the source language text; in sense disambiguation, the predictions are the senses assigned to the words of interest; in part-of-speech tagging or in classification, the predictions are the tags or classes assigned to each word. Because of this underlying similarity, the comparative analysis presented in this chapter is relevant to all these problems.

For a concrete example, consider the case of collocation extraction that has been addressed with statistical methods in the past. Smadja [1991; 1993] describes XTRACT, a system that initially uses the "nouns in vicinity" extraction model to collect co-occurrence information about words, and then identifies collocations on the basis of distributional criteria. A later component filters the retrieved collocations, removing the ones where the participating words are not used consistently in the same syntactic relationship. This post-processing stage doubles the precision of the system. I believe that using from the start a more sophisticated extraction model to collect these pairs of related words will have similar positive effects. In this way, we can benefit from linguistic knowledge without having to use a separate filtering process after expending the effort to collect the collocations. Other linguistic components, such as a morphology module that combines frequency counts, should also improve XTRACT's performance.

Similarly, the sense disambiguation problem is typically attacked by comparing the distribution of the neighbors of a word's occurrence to prototypical distributions associated with each of the word's senses [Gale *et al.* 1992b; Schütze 1992]. In many implemented disambiguation systems, no explicit linguistic knowledge is used in defining these neighbors, which are taken as all words appearing within a window of fixed width centered at the word being disambiguated. Many words unrelated to the word of interest are collected in this way. In contrast, identifying appropriate word classes that can be expected on linguistic grounds to convey significant information about the original word should increase the performance of the disambiguation system. Such classes might be modified nouns for adjectives, nouns in a subject or object position for verbs, etc. As I have shown in Section 5.3, less but cleaner information increases the quality of the results.

An interesting topic is the identification of parallels of the linguistic modules that have been designed with the present thesis in mind for these applications, at least for those modules which, unlike morphology, are not ubiquitous. Negative knowledge, for example, improves the performance of the grouping system, supplementing the positive information provided by adjective-noun pairs. It could be useful for other systems as well if an appropriate application-dependent method of defining and extracting such information is identified.

5.6 Lessons learned from the comparative study

I have presented results from an experimental study which showed that shallow linguistic knowledge, in the form of syntactic and morphological rules and negative learning examples, is effective in increasing system performance over pure statistical methods. This analysis established that all linguistic modules offered an improvement in performance, which was statistically significant at the 0.1% level for all modules except for one (spell checking). Furthermore, the contribution of the linguistic modules was also quantitatively significant[16] (see Section 5.3.3). Their incorporation in the system raised the performance score by a percentage between 16.15% and 59.36%, depending on the particular instance of the

[16]The results showed only a slight improvement for the spell checking module. I attribute this to the fact that this module also inadvertently removes valid words that are too specialized or rare and do not appear in its word list.

problem that was analyzed. And because the linguistic modules were selected to be general and exploit only shallow knowledge, their design, implementation, and interfacing with the statistical system took only slightly more than half the time that was needed for the development of the basic statistical system.

In addition, preliminary results from a separate study that analyzes the *trends* in the evaluation scores as the corpus size grows[17] indicate that asymptotic projections for the performance of the word grouping system when the latter utilizes linguistic knowledge are superior to similar projections for the basic statistical system. If this is indeed the case, it appears that linguistic knowledge is necessary for the system to reach the increased performance levels; pure statistical methods cannot compensate for it by simply using a larger training corpus.

Evidence for these claims was presented in several different forms, including analyses of the quantitative and statistical significance of the observed differences in performance. Furthermore, the results can be explained on linguistic grounds. Words and word relations are not all distributed identically; in particular, a naive method for extracting information about words may be systematically biased against some words or relations that have quite skewed distributions. An example from the problem addressed by the word grouping system is adjectives that only appear in a predicative position (e.g., *asleep*). A method that uses the most simple linguistic rule for detecting modification of nouns by adjectives can never collect enough data for such words, since it will not recognize the syntactic patterns in which they appear. On the other hand, the extraction method which collects *every* noun

[17]Obtained by fitting parametric approximating functions to the observed performance scores.

in the vicinity of an adjective does not have the ability to differentiate between valid and invalid associations. If occurrences of a noun correlate with occurrences of an adjective, that method will assume that a semantic relationship exists, which may not be the case when there is no syntactic link. Consequently, both of these methods will tend to systematically produce more biased estimates for learned parameters related to such words, no matter how much training text is supplied.

Of course, these results would be of limited value if they applied only to the specific system which was used in the experiments. However, the system that I employed exemplifies statistical learning systems for lexical knowledge acquisition, and the linguistic modules I considered are not limited to the particular task addressed by that system. As I explained in Section 5.5, many other statistical learning systems for natural language processing applications share the same basic methodology with the present system, even if their tasks are different. Even the use of adjective-adjective pairs as a source of negative examples, which is the most specialized linguistic feature used in the word grouping system, is a manifestation of the general principle of using multiple knowledge sources, and in particular sources of both positive and negative learning examples. While this feature may not be directly transferable to, say, machine translation, it can be argued that a different surface manifestation of the same principle could be identified and exploited for that problem. Furthermore, the features I selected do not require extensive amounts of knowledge to be accessed by the system, and can be efficiently computed in a completely automatic way for unrestricted text. Consequently, I expect the results of this analysis to be relevant to a significant number of other statistical natural language learning applications.

I close this discussion by noting that several other interactions between the parameters modeled in these experiments are worth studying. Although the extraction method and the corpus size are the parameters that most influence the overall performance of the system, it would be interesting to analyze any trends in the magnitude of the effects of the remaining linguistic modules as other parameters vary in a controlled manner, in particular in relation to changes in the corpus size and adjective set. I have thus far collected preliminary data for such an analysis as part of the comparative experiments reported here, but in the future I plan to stratify the experimental data and explore the influence that the corpus size and the characteristics of the problem instance exert on both the overall performance of the system and the contribution of individual linguistic modules. In addition, future experiments with a finer grid of corpus sizes, especially larger ones, would be useful for checking how well the predictions of the present study compare with new observations.

Part II

Analysis of the semantic structure of the automatically identified word groups

I have presented so far a collection of automated methods for measuring semantic similarity between words and constructing groups of closely semantically related adjectives and nouns. In the next four chapters I describe methods for the further analysis of these groups: First, I show how to use gradability indicators to separate scalar words from non-scalar ones, in order to limit the further processing to the scalar word groups for which properties such as semantic orientation and antonymy apply. In Chapter 7 I present a method that measures the extent any two words share the same semantic orientation and consequently separates the members of any particular group into subgroups of different orientations. Chapter 8 contains an analysis of text-level indicators of the linguistic property of semantic markedness, which can be successfully used on the identified subgroups to determine which of them contain semantically positive and which semantically negative terms. Finally, Chapter 9 puts together the various components of the thesis on a case study of the Deese antonyms, showing how the system can start from an unmarked list of the words in this set of adjectives and end up with their classification as pairs of antonyms.

The general methodology for addressing these problems is similar (see Section 1.9.2): Linguistic properties that bear on each problem are initially identified; these properties may have been explicitly studied (and named) in the linguistics literature, or may be phenomena that have not been studied in the past. Subsequently, indicators about these properties that can be automatically extracted from the text are identified, and methods for their extraction are discussed. On the basis of these indicators, an independent determination of the applicability of each property to each particular word or pair of words is made. When more than one property or indicators are relevant for one of the tasks, their indications are combined by a general learning algorithm. Also, if local decisions can produce conflicts, the conflicts are resolved to produce the final global decisions for each word or word pair. Figure 1.5 on page 69 depicts this general architecture model for the three components that operate on semantic groups.

In the subsequent three chapters, I address the issues listed above in more detail and separately for each of the three subproblems. I first present the linguistic properties that I have identified as relevant to each task, and then I discuss how these properties manifest themselves at the text level, corresponding to the syntagmatic relationships of Chapter 2. The details of the extraction process that actually retrieves instances of these indicators automatically from the corpus are discussed in the Appendix, along with the related issues of preprocessing the corpus and postprocessing the extracted information.

In addition to the case study of Chapter 9, sample output and results from the formal evaluation of each component are presented in the corresponding chapter. All these experiments were performed with various collections of adjectives as noted in Section 1.11, although the methods are directly transferable to other open-class words.

Chapter 6

Identifying scalar word groups

The subsystems that further analyze the produced semantic groups address the problems of determining differences in orientation and labeling words according to it; a future extension deals with the ordering of words on the basis of relative semantic strength. All these concepts apply to scalar words, which can be ordered according to semantic strength or degree of informativeness, and not to words that merely express alternate values of some underlying property. For example, we can partition the set of scalar adjectives describing temperature to a positive (e.g., {*warm*, *hot*}) and a negative (e.g., {*cool*, *cold*}) subscale; we can state that *cold* is an antonym of *hot* while both *warm* and *hot* indicate the same basic orientation of the property (i.e., existence of "warmness" rather than lack of it); and we can consider *hot* as a stronger indicator of temperature than *warm*. None of these distinctions is possible within a group of words that are semantically related through their association with a common property but do not participate in an order relation (e.g., {*economic*, *financial*, *political*, *technical*}, which have been identified by the present system as descriptors of different types of problems). This remains true even if the words in the group are mutually exclusive (i.e., paradigmatically complementary), as is the case with {*male*, *female*}[1] or with adjectives of color such as {*blue*, *green*, *red*}. We want to apply the methods that retrieve additional structure from the semantic groups (subdividing groups according to orientation, labeling subgroups, and (in the future) extracting order information) only to the scalar groups, for which such structure exists and the methods are valid. Therefore, we need an additional stage that separates scales from the non-structured non-scalar groups.

It is hard to measure directly whether a particular word is scalar in a quantitative manner using observable text properties, as I argued when I discussed related linguistic tests in Section 1.8. However, the property of gradability, i.e., whether a word can be modified by intensifiers or by formal suffixes to express the same concept in a different degree, is closely correlated with participation in scalar groups. I present arguments for this correlation in Section 6.1, where I also analyze the limited exceptions to the association between gradable and scalar words. Having decided to use gradability as the criterion for determining whether a word is scalar, I discuss in Section 6.2 how gradability information can be extracted from the text in an automated manner, and describe in Section 6.3 a learning algorithm that combines modification and morphology observations from the corpus to determine gradability at the word level and subsequently at the word group level. A formal evaluation of results produced by this method is given in Section 6.4. This evaluation is performed at two levels: first at the level of single words, using a large collection of 453 adjectives from the Wall Street Journal corpus; and secondly, at the level of word groups produced by the system (cf. Section 4.1.2) or present in corresponding reference word groupings constructed

---

[1] Contrast with the related set {*masculine*, *feminine*} where antonymy and gradation in degree, and consequently difference in orientation, exist.

by humans, where the gradability decisions made individually for each word are aggregated over each group.

6.1 Gradability as an alternative criterion for scalar words

Participation in a scale implies that the words express different degrees of the same property. The semantic strength of each word is relative to the norm for the scale, which can be modified by the surrounding context. Consequently, the meaning of a scalar word is not absolute, but is interpreted relative to its context and relative to pragmatic factors associated with the utterance (cf. the distinction between *absolute* and *relative* adjectives made by Katz [1972, p. 254]). For example, the modification of a noun by the scalar term *small* provides only the information that the object denotated by the noun is smaller that the typical object of its category. Modification by *small* cannot be mapped to an absolute interpretation of magnitude, as is evident from the very different dimensions of a *small insect*, a *small house*, and a *small planet*: a typical *small planet* will be much larger than a typical *large house*.[2] The relative character of the interpretation of scalar terms also blocks inferences which we would otherwise expect to be valid: *planet* is a hyponym of *object*, but we cannot (truthfully) say *This is a small planet, therefore it is a small object.* The last example should be contrasted with the validity of the same inference procedure when the scalar term is replaced with a non-scalar one; for example, we can say *This is a red car, therefore it is a red object.*

[2] Saint-Exupery's planets notwithstanding.

This relativism in the interpretation of a scalar term implies that it should be possible for the term to be explicitly modified by expressions that have an intensifying or diminishing effect. Such expressions can specify the meaning of the term more precisely (within the constraints placed by the norm that is explicitly or implicitly supplied). In fact, such a mechanism of variation of the semantic strength of the term may be necessary to cover *lexical gaps*, i.e., semantic concepts for which no single lexeme exists in the language and which must be expressed periphrastically. For example, *very small* can be used to describe something that is smaller than *small* but not as small as *tiny*; similarly, *extremely tiny* may be used to describe something that is so small that no corresponding single-word expression exists.

In addition to the above observations that indicate that a scalar word should accept intensifiers and diminishers, I note that the existence of the order relation $\prec$ in the scale implies that scalar words participate in implicit comparisons (e.g., an object that is described as *small* has less of the associated property (size) than an object of the same type that is described as *big*). Since implicit comparisons involving scalar terms are valid, explicit ones should be valid too.

Gradability (or *grading*) [Sapir 1944; Lyons 1977, p. 271] is the semantic property that enables a word to participate in comparative constructs and to accept modifying expressions that act as intensifiers or diminishers. The arguments presented above indicate that scalarity (i.e., the property that a word is scalar) should be strongly correlated with gradability. Given the lack of other direct indicators for selecting scalar terms, I have decided to use gradability as the sole criterion for the determination of whether a word is scalar or not.

The association of gradability with scalarity is strong but unfortunately not perfect. Most scalar words are gradable, so that expressions like *very small* or *bigger* are valid. On the other hand, most non-scalar words are not gradable, resulting in the unacceptability of expressions like **as financial as possible* and **more female*.[3] However, there are exceptions of two kinds:

- Some scalar words are not subject to interpretation according to a norm, but rather contain the norm as part of their semantic content. Since the norm and the meaning of the word is fixed, these words cannot be graded. Typically, the scales these words belong to contain just two antonymous terms, e.g., *alive–dead*. Because of the existence of such groups, some authors have specifically included gradability as part of the definition of scales [Lehrer 1974, p. 29], and used the term *ranks* to refer to scales containing non-gradable lexemes.

  The rare words that lexicalize the zero (neutral element) of a scale, (e.g., *medium* in {*small*, *medium*, *large*}) are also not gradable, although they are part of a scale. We can view the phenomenon of grading as similar to number multiplication, where the unmodified word provides a base value (according to the norm currently in effect) and the intensifier or diminisher provides a multiplier. But multiplying zero with any number yields always zero, so there is no point in attempting to explicitly or implicitly multiply a zero value. Consequently, the words that lexicalize zero values (such as *medium*) cannot be graded.

- Sometimes, normally non-gradable words appear within the scope of a grading con-

[3] Contrast **more female* with *more feminine*.

struct. For example, we may refer to someone or something as being *very dead*, or describe an object as being *redder* than another. These uses, however, are rather atypical and do not negate the basic lack of gradability of the word. *Very dead* does not imply that the person or object in question is more dead than what the unmodified word would signify. Rather, the intensifier *very* is employed to add emphasis to the utterance. In the same vein, *redder* implies a higher concentration of color or a higher similarity to the prototypical color "red", not a different color that is more "red" than the unmodified *red*. An example of the first interpretation is the phrase *Her face became redder and redder*, where the comparative is used mainly to express a progressive feature. The second interpretation is exemplified by the following sentence from the 1987 Wall Street Journal corpus: *Now, however, growers and plant breeders are stepping up efforts to develop hardy tomato varieties that can be shipped riper and naturally redder.*

The existence of the above exceptions shows that gradability is not always a perfect indicator that a word is scalar. Nevertheless, the number of non-gradable scalar words is very low, and by using a threshold on the relative frequency of gradable versus non-gradable occurrences of a word we can identify exceptional cases where non-gradable words are used as if they were gradable ones. Consequently, I will base the decision of whether a particular word is part of a scale on its apparent gradability in the corpus. However, there is an important subclass of semantic groups that involve non-gradable words and yet contain pairs of antonymous words. Such groups typically have just two members in an antonymous relationship, and one of the two words is derived from the other through the addition of a negative prefix, e.g., *constitutional–unconstitutional*, or, for an example involving verbs, *judge–misjudge*. Note that *constitutional* (or, similarly, *judge*) is not gradable, as something either is constitutional or is not; yet, we would like to be able to extract the semantic opposition between *constitutional* and *unconstitutional*. Consequently, I consider as scalar and leave for further processing by the subsequent components of the system those semantic groups that either contain gradable words or fall in the special case of groups that contain only two words one of which is derived from the other by the addition of a negative prefix.

6.2 Text-level indicators of gradability

Gradability of adjectives is indicated through the use of explicit or implicit comparative forms, and through modification by grading adverbials.[4] The last method is also employed to grade verbs. Comparative forms include single-word inflected forms (e.g., *big–bigger–biggest*) which indicate a higher semantic strength than what corresponds to the unmodified word, and periphrastic forms (e.g., *appropriate–more appropriate–most appropriate–less appropriate–least appropriate*) where the comparative form can be semantically stronger or weaker than the base form.[5] They also include constructs that indicate equivalence between two phrases (e.g., *equally big*, *as big as* ...). Several qualifiers and adverbs (e.g.,

[4]The linguistic properties discussed in this and the next two chapters, and to a larger degree their indicators in the text, depend to some extent on the particular syntactic class of the words being analyzed. Since I focus on the analysis of adjectival scales (see Section 1.11) which is also the prototypical class of scales, I address mostly indicators applicable to adjectives. However, I mention in several cases how the indicators would be employed for the retrieval of semantic information for verbs (the only other class of words for which the concepts of this chapter apply to a significant degree).

[5]Thus the two indicators overlap for polysyllabic adjectives with periphrastic comparative and superlative forms.

very, *so*, *extremely*) and noun phrases functioning as adverbs (e.g., *a bit*, *a little*) indicate gradability when they modify adjectives. In fact, the most common use of an adverb that modifies an adjective is to function as an intensifier or diminisher [Quirk *et al.* 1985, p. 445]; modification by an adverb is also the main method for intensifying or diminishing a verb.

To identify occurrences of adjectives inflected for degree, I use an algorithm that analyzes each adjective morphologically; this is an alternative to the necessarily non-exhaustive approach of storing inflected forms in a word list. The program has a small list of irregular comparative and superlative forms (such as *better* and *furthest*) which are matched directly with each candidate inflected adjective. To determine whether other adjectives are inflected for degree, I look at the ending characters, which should match *-er* or *-est*. For the adjectives matching this criterion, rules that derive the original form under the assumption that the present form is truly inflected for degree are applied; these rules are the reverse of the forward rules that produce the inflected form, and have been compiled from the forward rules listed in [Quirk *et al.* 1985, pp. 458–461]. However, since several original word forms can correspond to the same inflected form (for example, *big*, *bigg*, and *bigge* would all produce the comparative form *bigger*), the application of the reverse morphology rules typically results in multiple candidate solutions. I thus measure the frequency of each candidate solution in the corpus, and select as the base form the word that has the highest frequency and exceeds a predetermined frequency threshold. Typically, at most one candidate solution will exceed the threshold (other candidates will have zero frequency in the corpus, except for occasional typographical errors); the threshold guards against mistakenly identifying as inflected words that end in *-er* or *-est* but are not inflected (for example, *silver* or *earnest*).

For such cases, no candidate solution's frequency will surpass the threshold. The threshold is also useful in addressing the occasional exceptional uses of non-gradable words in graded constructs, which I analyzed in the previous section. The value of the threshold represents a tradeoff between precision and recall; using a higher threshold, the algorithm will avoid misidentification errors resulting from accumulated typographical errors but will also miss valid inflected words with an infrequent base form. More details on this algorithm and the rules employed for deriving candidate base forms are given in Section A.5 of the Appendix.

For the second indicator of gradability, I extract pairs of adjectives and modifying adverbs from the corpus using the finite state grammar of Section A.4.2. However, modifying adverbs can also add to the semantic content of the adjectival or verb phrase instead of providing a grading effect (e.g., *immediately available*, *politically vulnerable*), or function as emphasizers, adding to the force of the base adjective and not to its degree (e.g., *virtually impossible*; compare **very impossible*). Therefore, we cannot assume that all adjectives modified by adverbs are gradable ones. Instead, I use a fixed list of common adverbs and other modifiers that are practically always used as intensifiers or diminishers. This list, which is given in Figure 6.1, was constructed from analyzing the function of adverbs in adverb-adjective pairs collected from a part of the 1987 Wall Street Journal corpus (see Section A.2 in the Appendix), and was augmented with comparable characterizations of adverbs in [Quirk *et al.* 1985, pp. 445–448 and 589–591].

a bit	disproportionately	hardly	much†	singularly
a little	doubly	heavily	notoriously	slightly
absolutely	dramatically	highly	obscenely	so
absurdly	drastically	immeasurably	outrageously	somewhat
acutely	enormously	immensely	overly	spectacularly
alarmingly	enough	increasingly†	overwhelmingly	strongly
amazingly	equally	incredibly	pretty	terribly
as	exceedingly	infinitely	progressively†	too
astonishingly	exceptionally	least	proportionately†	tremendously
astronomically	excessively	less	quite	unbelievably
awfully	extraordinarily	minimally	rather	unusually
barely	extremely	moderately	relatively	utterly
completely	fairly	modestly	remarkably	vastly†
considerably†	far†	more	scarcely	very
deeply	greatly	most	significantly	

Figure 6.1: Adverbs, qualifiers, and idiomatic noun phrases that indicate gradability when used as modifiers of adjectives. The noun phrases in the list are identified during preprocessing and converted to special tokens marked as adverbials, so that they can be recognized as modifiers of adjectives by the grammar of Section A.4.2 of the Appendix. Adverbs marked with † appear primarily with the comparative form of the adjective they modify, or with adjectives that are inherently comparative (e.g., *different*, *superior*). Several adverbs that may function as either intensifiers (with gradable adjectives) or emphasizers (mostly with non-gradable adjectives) have been omitted. The last category includes, for example, *particularly* and *really*.

6.3 Determining whether a word or word group is scalar

In the previous section I identified two indicators of gradability that can be measured directly from a corpus: inflection for degree and modification by intensifying or diminishing adverbs. When we need to decide whether a particular adjective is gradable, we gather the indicative data from the corpus. The result is a collection of frequencies in a 2×2 table for each word, with one dimension representing inflection for degree and another modification by intensifiers or diminishers. Examples of these tables for one gradable (*cold*) and one

	cold		civil	
	Unmodified by grading words	Modified by grading words	Unmodified by grading words	Modified by grading words
Uninflected	392	20	1,296	1
Inflected for degree	18	0	0	0

Table 6.1: Extracted values of gradability indicators, i.e., frequencies of the word with or without the specified inflection or modification, for two adjectives, one gradable (*cold*) and one non-gradable (*civil*). The frequencies were computed from the 1987 Wall Street Journal corpus.

non-gradable (*civil*) adjective are shown in Table 6.1, using data from the 1987 Wall Street Journal corpus.

According to the arguments of Sections 6.1–6.2, a gradable word should exhibit significant numbers in the cells of the table that correspond to forms inflected for degree and occurrences modified by intensifiers or diminishers. For a non-gradable word these numbers should be ideally zero, or, allowing for errors in the detection algorithm, idiomatic uses, and typographical mistakes, at least low. Of course, "significant" and "low" are relative to the total frequency of the word. For example, we can see from Table 6.1 that $18 + 20 = 38$ out of a total 430 occurrences of *cold* in the corpus, or 8.84%, are in a context that indicates gradability; the corresponding number for *civil* is only 1 out of 1297 occurrences, or 0.08%.

Given these observations, I use the following three variables to determine whether a word is gradable:

- The ratio of the total frequency of forms of the word that are inflected for degree to the total frequency of all forms of the word, inflected or not.

- The ratio of the number of occurrences of the base form of the word modified by the intensifiers or diminishers of Figure 6.1 to the total number of occurrences of the base form of the word.

- The ratio of the number of occurrences of the word in both inflected form and with a modifying intensifier or diminisher over the total frequency of the base form of the word. The frequency of the base form is used in this ratio because, for a non-gradable word, the number of occurrences of inflected for degree forms will usually be zero.[6] This variable measures the proportion of occurrences of the word where both of the previously mentioned indicators of gradability are simultaneously present, and therefore is expected to be a stronger indicator of gradability.

To combine these three numbers into a decision for the gradability of a given word, I use a general log-linear learning model. Log-linear models [Santner and Duffy 1989] are an extension of the classical linear regression models that I described in Section 5.3.2. As I noted then, a linear regression model relates the response variable R to the predictor variables V_i through a weighted sum formula

$$R = \beta_0 + \sum_{i=1}^{n} \beta_i V_i$$

where the coefficients β_i are estimated from training samples for which the correct response R is known.

---

[6]Note that the algorithm for detecting inflected forms guarantees that the frequency of the base form of a word will always be non-zero; otherwise no inflected forms of this word can ever be identified.

Methods for including categorical variables (i.e., variables that take values from a predetermined, unordered set) as predictors have also been described in Section 5.3.2. These methods replace each categorical variable with a set of real-valued variables that encode the contrasts between the different values of the categorical variable. In the case of determining gradability, we need a different kind of extension to the linear model: now the response variable is categorical with the values "gradable" and "non-gradable". A similar situation arises in many other contexts, i.e., whenever a decision from a fixed set of alternatives is made on the basis of numerical and/or categorical predictors.

One solution to the problem, in the case of binary decisions, is to transform the response variable using an empirically determined threshold and convert values of R above the threshold to one of the two outcomes and values below the threshold to the other outcome. This approach, however, suffers from several disadvantages. In particular, using a linear model and then replacing predicted values above the threshold with one decision and the remaining values with the other decision results in loss of information, as the result doesn't contain any information about the closeness of the decision. This suggests that a more appropriate transformation would be the so-called *logistic transformation*

$$R' = \frac{e^R}{1 + e^R} \tag{6.1}$$

where R is the response predicted with the linear model of equation (6.3) and e is the base of the natural logarithms. This transformation guarantees that the final response R' lies in the interval (0,1) with the endpoints representing the two different outcomes, and with values closer to the endpoints indicating stronger evidence for the suggested decision.

Even with the transformation suggested by equation (6.1), one disadvantage of the direct application of the linear model remains, namely that constant variance is assumed for all samples. However, if the data is modeled as a binary response, then the transformed response R' should follow a binomial distribution and not a normal one. In that case, the variance depends on the mean according to the formula

$$\mathbf{var}[R'] = \mathbf{E}[R'] \cdot (1 - \mathbf{E}[R'])$$

taking larger values near the middle of the interval [0,1] for $\mathbf{E}[R']$.

To address this problem we can incorporate in the model both the dependence of the final response on the intermediate linear response and the dependence of the variance on the mean of the final response, resulting in a *generalized linear model* [McCullagh and Nelder 1989]. In the particular case of a binary response and with the assumption of binomial distribution given above, we get the well-known *logistic* or *log-linear regression* model [Santner and Duffy 1989]. This model uses the basic linear formula of equation (6.3), and transforms this intermediate result according to the logistic transformation of equation (6.1); however, the non-linearity of the overall model is taken into account when the coefficients are estimated. A variation of the least squares algorithm for this enhanced model, namely the *iterative reweighted non-linear least squares* algorithm [Baker and Nelder 1978; Bates and Watts 1988], enables the estimation of the coefficients β_i on maximum likelihood principles as before. The techniques of Section 5.3.2 can also be used to construct models with a binary response and a combination of numerical and categorical predictors.

Since I am interested in combining evidence into binary decisions, I will use the latter model for the problem of determining the gradability of words and in other similar applications in later chapters. In the particular case of the model for predicting gradability values, I use the three ratios described above as the variables entering the linear predictor; the appropriate weights are estimated from a list of labeled samples, using a small set of words for which gradability information has been assigned by hand, and a training corpus for estimating the frequency ratios. Once the weights have been determined from the training data, they are used for the labeling of the remaining words as gradable or not.

To decide if a given word group is scalar, I combine the gradability decisions made for the individual words in it. The log-linear regression model for gradability is constructed so that 1 corresponds to the "gradable" label and 0 to the "non-gradable" label. I sum the assigned gradability values (which lie between 0 and 1) across all the members of a given group. Then, if this sum exceeds $n/2$, where n is the cardinality of the group, the group is considered scalar; similarly, if the sum is below $n/2$ the group is considered non-scalar and is not analyzed by the subsequent components of the system. Groups for which the sum of gradability values is exactly $n/2$ (a very unlikely case) are randomly assigned to one of the two categories.

This general procedure is applied to groups with one or more than two members, as well as to groups with two words that are not morphologically related. Groups with only two words where one of them is morphologically derived from the other with the addition of a negative prefix (such as *in-* or *un-*) contain almost always two antonyms. The two words in the group may be both gradable (e.g., {*polite*, *impolite*}) or one of them may be gradable and the other an absolute, non-gradable term (e.g., {*constitutional*, *unconstitutional*}), as I noted in Section 6.1. In any case, the group should be considered scalar for our purposes. Therefore, I explicitly test whether the two words in each two-member group satisfy the above morphological relationship criterion, and include such groups in the collection of scalar groups irrespective of the gradability values that have been computed for their member words.

6.4 Experimental results and evaluation

6.4.1 Recovering gradability values for single words

I extracted from the 1987 Wall Street Journal corpus all adjectives with a frequency of 300 or more; this produced a collection of 496 words. Gradability labels specifying whether each word is gradable or not were manually assigned, using the designations of the Collins COBUILD (Collins Birmingham University International Language Database) dictionary [Sinclair 1987b]. COBUILD marks each sense of each adjective with one of the labels QUALIT, CLASSIF, or COLOR, corresponding to gradable, non-gradable, and color adjectives. In cases where COBUILD supplies conflicting labels for different senses of a word, I either omitted that word or, if a sense seemed predominant, I gave it the label of that sense. In some cases, the word did not appear in COBUILD; these typically were descriptive compounds specific to the domain (e.g., *anti-takeover*, *over-the-counter*) and were in most cases marked as non-gradable adjectives. Overall, 453 of the 496 adjectives (91.33%) were assigned gradability labels by hand, while the remaining 53 words were discarded because they were misclassified as adjectives by the part-of-speech tagger (e.g., *such*) or because they could not be assigned a unique gradability label in accordance with COBUILD. Out of these words, 235 (51.88%) were classified as gradable adjectives, and 218 (48.12%) were classified as non-gradable adjectives.

I extracted the relevant modification and inflection indicators for the 453 adjectives with manually assigned gradability labels, following the algorithms of Section 6.2; the specifics of the extraction process are discussed in detail in Sections A.4 and A.5 of the Appendix. I noticed from the collected data that one of the three indicators of Section 6.3, namely the modification of an adjective by a grading modifier while the adjective is already inflected for degree (e.g., *much better*), always predicts correctly that the adjective is gradable. There are 44 adjectives among the 453 in our collection that appear at least once with this "double" indication of gradability, and all these adjectives are marked as gradable in the COBUILD-based reference model. On the basis of this observation, I modify the algorithm of Section 6.3 to automatically consider an adjective with at least one occurrence of this indicator as gradable, and use the remaining two gradability indicators (inflection for degree and modification by a grading adverb) but not their combination as the predictor variables in the log-linear model.

I randomly choose 100 of the 453 adjectives as the test set on which the gradability prediction algorithm will be evaluated. Selecting a test set of a fixed size guarantees that the percentage of adjectives in the test set to which the surefire, "double" indicator of gradability will apply is representative of the overall percentage of adjectives in the entire collection to which this indicator applies. After fixing the test set, I randomly select 300 of the remaining adjectives to train the log-linear model of Section 6.3.[7] The resulting model is used to predict gradability values for the 100 adjectives in the test set, and these assignments are compared with the corresponding labels in the reference model. This entire operation of selecting random test and training sets, fitting the model's coefficients, making predictions, and evaluating the predicted gradability labels is repeated 100 times, to ensure that the evaluation is not affected by a lucky (or unlucky) partition of the data between training and test sets.

Applying the above procedure to the data collected for the 453 adjectives resulted in gradability classifications with an average precision of 93.55% and average recall of 82.24% (in terms of the gradable words reported or recovered, respectively). The overall accuracy of the predicted gradability labels is 87.97%. This represents a small gain in performance over letting the log-linear model automatically figure out that the third gradability indicator is always correct when it is applicable; in that case (i.e., as originally discussed in Section 6.3), the algorithm achieves on the same collection of words and with the same experimental design average precision of 93.86%, average recall of 81.70%, and average overall accuracy of 87.70%, again over 100 random repetitions of the modeling and evaluation cycle.

Figure 6.2 lists the gradability labels that were automatically assigned to one of the 100 random test sets using the modified prediction algorithm. This test set, and the corresponding log-linear model, were chosen at random among the 100 similar test sets and

[7] With the modification, as noted above, that only two predictors are now employed.

Classified as gradable:

acceptable accurate afraid aware busy careful cautious cheap creative critical dangerous different disappointing equal **fair familiar far favorable formal free frequent good grand inadequate intense interesting legitimate likely positive professional reasonable rich short-term significant slow solid sophisticated sound speculative thin tight tough uncertain widespread** worth

Classified as non-gradable:

additional alleged alternative annual antitrust automatic certain **criminal cumulative daily deputy domestic** elderly **false financial first-quarter** full hefty **illegal institutional internal legislative long-distance military minimum monthly** moral **national official one-time other** outstanding **present prior prospective punitive regional scientific secondary sexual subsidiary taxable three-month three-year total tremendous two-year** unfair **unsolicited upper voluntary** white **wholesale world-wide** wrong

Figure 6.2: Automatically obtained classification of a sample of 100 adjectives as gradable or not. Correct decisions (according to the COBUILD-based reference model) are indicated in bold.

models computed for this analysis, and achieves slightly better than average scores. The gradability assignments of Figure 6.2 receive a precision score of 95.56% when compared to the manually assigned labels, a recall score of 82.69%, and an overall accuracy score of 89%.

6.4.2 Classifying word groups as scalar

The algorithm of Section 6.3 classifies a group of semantically related words as scalar either on the basis of a specific criterion of morphological relatedness (when the group contains only two words and one of them is derived from the other with the addition of a negative

1. annual daily quarterly	9. international local national regional
2. difficult easy hard	10. commercial military
3. big modest small	11. private public
4. different similar	12. net pre-tax
5. final initial next	13. new old
6. current future past previous recent	14. negative positive
7. chief important senior significant	15. likely possible
8. high low	16. strong weak

Figure 6.3: One of the human-constructed models for the adjective test set 2 of Figure 4.2 on page 171.

prefix) or, in most cases, by aggregating the predictions for gradability made individually for each word in the group. In order to evaluate the performance of the algorithm, I selected two representative groupings of adjectives, one for each of the adjective test sets 2 and 3 that I used in Chapter 2. The two selected collections of word groups are shown in Figures 4.8 and 4.9, respectively, on page 178.

Because these two word groupings were produced automatically by the first stage of the system, the groups in them are not perfect; in particular, they may not always correspond to a meaningful scalar or non-scalar group because they may mix words from different groups. To measure the performance of the gradability prediction algorithm at the group level without being affected by the errors of the clustering stage, I also evaluated the algorithm of Section 6.3 on groupings produced by humans for the same two collections of adjectives. One such grouping for each of adjective test sets 2 and 3 was selected randomly. These additional collections of word groups are shown in Figures 6.3 and 6.4.

1. capitalist communist leftist socialist	12. hostile pleasant unfriendly
2. brilliant clever dumb smart stupid	13. abrupt gradual
3. clean dirty	14. endless protracted
4. catastrophic disastrous	15. delicate energetic vigorous
5. chaotic turbulent	16. lenient strict stringent
6. deadly fatal harmful hazardous toxic	17. dual multiple
7. affordable inexpensive	18. dry wet
8. fat slim	19. unreasonable unstable
9. flexible fragile rigid	20. dismal gloomy
10. affluent impoverished prosperous	21. generous insufficient meager scant
11. astonishing exotic outrageous plain stunning	22. confusing misleading

Figure 6.4: One of the human-constructed models for the adjective test set 3 of Figure 4.3 on page 171.

I assigned to each of the groups in these four adjective groupings a label, classifying the group as either a scale or a non-scalar group of complementary terms. Sometimes, an allowance had to be made when a group contained words from a particular scale or non-scalar group but also additional words from other groups, or when some of the words from the corresponding scale or non-scalar group were missing. These differences from the ideal groups were most frequent and pronounced in the automatically constructed groups, as expected. But in all cases, I was able to reach a classification that accurately reflected the status of the group, even if that was slightly distorted from word additions and omissions. Table 6.2 lists the distributional characteristics of the groups in the four test collections, according to whether they were classified as scalar or not. Naturally, this operation of

	Number of words	Number of groups	Scalar groups	Non-scalar groups
Automatically constructed grouping of words from set 2 (Figure 4.8)	43	15	6 (40.00%)	9 (60.00%)
Human-prepared grouping of words from set 2 (Figure 6.3)		16	9 (56.25%)	7 (43.75%)
Automatically constructed grouping of words from set 3 (Figure 4.9)	62	23	19 (82.61%)	4 (17.39%)
Human-prepared grouping of words from set 3 (Figure 6.4)		22	19 (86.36%)	3 (13.64%)

Table 6.2: Distribution of scalar and non-scalar groups in the four groupings which are used to evaluate the performance of the algorithm that predicts whether a group is scalar or not.

---

classifying the word groups according to scalarity was performed in advance of all training and evaluation experiments.

The adjective test sets 2 and 3 that provide the words in the group collections contain 43 and 62 words respectively. Since none of the evaluated groups consists of two morphologically related antonyms, decisions about the scalarity of each group must be based on the gradability values independently assigned to these $43 + 62 = 105$ words.[8] Unlike the experiments of the previous subsection, the test set for the automatic method that predicts gradability values is now fixed to these 105 words. Furthermore, 43 of these 105 words belong to the collection of 453 adjectives that was used to measure the performance of the gradability prediction algorithm earlier. Since I must exclude these words from the training data, I cannot re-use the fitted log-linear models of the previous subsection.

[8] There is no overlap between the members of adjective test sets 2 and 3; cf. Section 4.1.1.

I thus first extract the relevant data from the corpus (frequency in base form, in inflected forms, in the scope of grading modifiers, and with both of the last two conditions simultaneously satisfied) for the $105 - 43 = 62$ words in the test set that do not belong in the set of adjectives with preassigned gradability labels described in Section 6.4.1. I then select a random sample of 300 words among the $453 - 43 = 410$ adjectives of the previous subsection that do not belong in the test set, train the parameters of the log-linear model, and make predictions for the 105 words in the test set. The prediction algorithm incorporates the observation made in Section 6.4.1 for words that have an inflected form modified by grading adverbs, automatically labeling such an adjective as gradable with maximum confidence. The real-valued measure of gradability[9] that the algorithm produces is then aggregated across all members of a given group, as described in Section 6.3. The result is a single decision on whether each group is a scale or not, and is evaluated against the classification of the groups in the model.

I performed the above steps of selecting a training sample, estimating the coefficients of the log-linear model, making predictions for the 105 test words, aggregating these predictions to classify the groups, and evaluating the group labels, 100 times, to eliminate any effects from any particular selection of the training sample. Table 6.3 shows the average precision and recall (in terms of the scalar groups reported or retrieved) and the average overall accuracy achieved by this method across all 100 runs, while Table 6.4 shows a randomly selected set of gradability values computed during one of these runs. I observe from Table 6.3 that the algorithm performs with high accuracy, significantly higher than the

[9] Between 0 and 1, where 1 indicates gradability most strongly.

	Number of groups	Precision	Recall	Overall accuracy
Automatically constructed grouping of words from set 2 (Figure 4.8)	15	100.00%	100.00%	100.00%
Human-prepared grouping of words from set 2 (Figure 6.3)	16	100.00%	100.00%	100.00%
Automatically constructed grouping of words from set 3 (Figure 4.9)	23	94.50%	90.53%	87.83%
Human-prepared grouping of words from set 3 (Figure 6.4)	22	98.20%	100.00%	98.36%
Both groupings of words from set 2	31	100.00%	100.00%	100.00%
Both groupings of words from set 3	45	96.40%	95.26%	92.98%
Both automatically constructed groupings	38	95.87%	92.80%	92.63%
Both human-prepared groupings	38	98.76%	100.00%	99.05%
All four groupings	76	97.42%	96.60%	95.84%

Table 6.3: Performance scores for the algorithm that predicts whether a word group is scalar or not. The scores reported are averages over 100 sets of gradability predictions for the words in the groups, each of which was computed using a different random sample for training the gradability model.

accuracy of the underlying gradability predictions (cf. Section 6.4.1). In fact, for one of the test collections of adjectives, the system achieves perfect scores in all evaluation metrics for both the automatically constructed and the human-prepared word groupings, even though these two groupings differ significantly in the distribution of scalar and non-scalar groups (cf. Table 6.2). This performance increase is due to the aggregation of weaker (but, generally, mostly correct) decisions, which allows the system to correctly infer the overall status of a group even when a word is missing or an extraneous word is present and when a small number of the decisions for the words in the group are in error. As expected, the performance of the scale identification algorithm is higher on the groupings made by humans, as in these cases errors due to words being placed in the wrong group do not influence the evaluation process.

abrupt	0.3885	flexible	1.0000	plain	0.6486
affluent	1.0000	fragile	0.9984	pleasant	1.0000
affordable	1.0000	future	0.1437	positive	0.9998
annual	0.1085	generous	1.0000	possible	0.2835
astonishing	0.9690	gloomy	0.9999	pre-tax	0.1034
big	1.0000	gradual	0.9529	previous	0.1211
brilliant	0.9999	hard	1.0000	private	0.2406
capitalist	0.1034	harmful	1.0000	prosperous	1.0000
catastrophic	0.1034	hazardous	0.9163	protracted	0.7563
chaotic	0.9991	high	1.0000	public	0.2043
chief	0.1599	hostile	0.7402	quarterly	0.1034
clean	1.0000	important	1.0000	recent	0.9453
clever	1.0000	impoverished	0.3777	regional	0.1427
commercial	0.1696	inexpensive	0.9720	rigid	1.0000
communist	0.1034	initial	0.1171	scant	0.1992
confusing	1.0000	insufficient	0.6666	senior	0.2179
current	0.1343	international	0.1816	significant	0.9989
daily	0.1209	leftist	0.1944	similar	0.3531
deadly	0.9731	lenient	1.0000	slim	1.0000
delicate	1.0000	likely	1.0000	small	1.0000
different	0.9947	local	0.1617	smart	1.0000
difficult	1.0000	low	1.0000	socialist	0.2321
dirty	0.9905	meager	0.7332	strict	1.0000
disastrous	0.9575	military	0.1536	stringent	1.0000
dismal	0.9769	misleading	0.7088	strong	1.0000
dry	0.9805	modest	0.9994	stunning	0.9760
dual	0.1034	multiple	0.1034	stupid	1.0000
dumb	1.0000	national	0.1173	toxic	0.9832
easy	1.0000	negative	0.8358	turbulent	0.9998
endless	0.2130	net	0.1144	unfriendly	0.1992
energetic	0.9710	new	1.0000	unreasonable	0.3644
exotic	1.0000	next	0.1282	unstable	0.9979
fat	1.0000	old	1.0000	vigorous	1.0000
fatal	0.1605	outrageous	1.0000	weak	1.0000
final	0.1071	past	0.1066	wet	0.9846

Table 6.4: Sample gradability values for the 105 adjectives in adjective test sets 2 and 3, computed for the purpose of determining whether groups of these adjectives are scalar or not. The values listed are produced during one of the 100 runs of the algorithm on these word groups; this run was randomly selected.

Chapter 7

Separating word groups according to semantic orientation

Once we have identified the scalar word groups with the methods of the previous chapter, the next step is to retrieve the additional structure present in them. To this end, we want to subdivide each scalar group into subscales of different semantic *orientation* (or *polarity*), one containing positive and one negative terms. Several different types of semantic relationships may exist between two words belonging to the same semantic group. For the purpose of separating a scalar semantic group into two different subgroups according to orientation, it is useful to distinguish relationships of an antonymous character (e.g., *big–small*) from relationships that imply that the two words participating in them have the same polarity. The latter class of relationships includes synonymy (e.g., *tiny–minuscule*) and the relationship "is semantically less strong" ($\prec$) of Section 1.7 (e.g., *big–huge*).[1]

[1]Hyponymy is another relationship that implies identical polarity for the two related words, but in the interpretation I adopt in this dissertation, this relationship does not hold between scalar words. Rather it is replaced with the more precise "is semantically less strong" relationship. See also footnote 2 on page 30 and the related discussion in Section 1.3.

Although there are important differences between the relationships that indicate identical polarity, these differences do not bear on the task of separating the scale into two subscales. Therefore, as far as this particular task is concerned, I will group all evidence offered by these relationships together, creating a new superordinate relationship that I will call *same orientation*. The negation of this relationship, namely the relationship *different orientation*, is essentially the antonymy relationship, interpreted sufficiently broadly so that *hot* and *cool* are considered antonyms, in addition to the more traditional pairs *hot–cold* and *warm–cool*.

I have divided the task of identifying subscales into two phases: First, all words are placed in one of two large classes, so that words with the same orientation (either positive or negative) are put in the same class. By combining the information from this division of the words and their division according to their closeness in meaning (see Chapter 2), we obtain the classification of the words into subscales. These subscales are, however, unlabeled, i.e., we don't know which contain positive terms. The second phase, which is described in the next chapter, addresses this labeling problem, producing the final labeled subscales and, indirectly, information about antonymy.

I devote the first section of the present chapter to a description of the property of semantic orientation. I then describe how a powerful indirect indicator of same or different orientation can be extracted from the text, by proposing and verifying empirically[2] the

---

[2]Throughout this chapter, I use the 1987 Wall Street Journal corpus as the corpus from which data for a large collection of 1,336 adjectives is drawn to verify the proposed hypotheses and train and test the algorithms I present. I follow the presentation of each algorithm with the results produced when it is applied to this corpus.

conjunction hypothesis, namely that the choice of a connective is constrained by the semantic orientations of the words that are joined; this indicator is supplemented by information extracted from morphological relationships between adjectives. Since connectives cannot be used freely without regard to the orientation of the conjoined words, it follows that we can predict whether any two adjectives have the same or different orientation from the pattern of connectives used in the conjunctions the two words appear in. Section 7.5 presents a log-linear learning model that computes a measure of orientation similarity, analogous to the measure of semantic similarity that was computed for the purpose of semantic grouping in Section 2.3. Continuing the analogy with my approach to the semantic grouping problem, a clustering algorithm places the words into the two large classes corresponding to the two orientations (positive and negative; compare with Section 2.7.1). I conclude this chapter by presenting results from the application of the orientation classification method to several sets of adjectives, which I supplement with results from simulation experiments. These results clarify the relationship between the quality of the output and the size (or, more precisely, density in conjunctions) of the corpus and establish the scalability of the approach.

7.1 Semantic orientation

The semantic *orientation* or *polarity* of a word indicates the direction (positive or negative) the word deviates from the norm for its semantic group or *lexical field* [Lehrer 1974]. It also constrains the word's usage in the language [Lyons 1977], due to its *evaluative* characteristics [Battistella 1990]. For example, some nearly synonymous words differ in orientation because one implies desirability and the other does not (e.g., *simple* versus *simplistic*).[3] In linguistic constructs such as conjunctions, which impose constraints on the semantic orientation of their arguments [Anscombre and Ducrot 1983; Elhadad and McKeown 1990], the choices of arguments and connective are mutually constrained, as illustrated by:

$$\text{The tax proposal was} \left\{ \begin{array}{l} \text{simple and well-received} \\ \text{simplistic but well-received} \\ {}^{*}\text{simplistic and well-received} \end{array} \right\} \text{by the public.}$$

In addition, almost all antonyms have different semantic orientations.[4] This also holds in the reverse direction: if we know that two words relate to the same property (for example, by being members of the same scalar group such as *hot* and *cold*) but have different orientations, we can usually infer that they are antonyms.

Orientation can receive either a *positive* or a *negative* value; as noted above, in each pair of antonyms one of them will be characterized as positive and the other as negative.[5] The meaning of these descriptive terms can be defined according to a variety of criteria, which are all highly correlated:

- Evaluative criteria of a psycholinguistic character, where sets of antonyms are pre-

---

[3] While *simple* does not necessarily imply desirability in all contexts, it clearly possesses this characteristic in opposition to *simplistic*. Thus, within this pair of alternatives, *simple* has positive orientation.

[4] Exceptions include a small number of terms that are both negative from a pragmatic viewpoint and yet stand in an antonymic relationship; such terms frequently lexicalize two unwanted extremes, e.g., *verbose–terse*.

[5] Sometimes, one word may be characterized as *neutral*; however, as I explain later, this does not present a significant problem for the analysis that follows.

sented to a number of human subjects who are asked to select the "positive" term out of each pair. Such experiments show that the subjects consider *deep* more positive than *shallow*, *beautiful* more positive than *ugly*, etc.

- Lexico-semantic criteria, that relate to the property that all the members of a group describe and to the way this property is commonly lexicalized. This lexicalization process indicates which of the two words is more closely related with the property, and would therefore be characterized as +PROPERTY (hence the terms positive and negative); the positive term *possesses* the property, while the negative term *lacks* the property. For example, we speak of *height* or *difficulty*, not of *lowness* or *easiness*.

- Extra-linguistic denotational criteria, which relate to the property the words in a group describe and the way this property is measured in absolute terms among the objects that the words denote in the real world. The direction in which the property increases is the positive direction, and the words that lie in this direction belong to the positive subgroup. For example, *warm* and *hot* are identified as positive terms because the underlying physical property (temperature) is higher for objects described by these words than for similar objects described by *cool* or *cold*. Similarly, *intelligent* and *brilliant* are positive terms because of the higher values of the underlying abstract property (intelligence) in comparison with *dumb* and *stupid*. A related observation is the association of an absolute zero (or limit) with the negative side of a scale; as Lehrer [1974, p. 27] observes, "a thing can be so narrow or so short or so small that it approaches zero in extension, but there is no corresponding limit to how large, wide, or tall something can be."

Among the above criteria, I have adopted the evaluative criterion as the definitive one, because it is both easier to apply and more general (relevant to more words) than the other two criteria. When this criterion is applied to adjectives, it assigns orientation labels according to whether the use of a particular adjective ascribes in general a positive or negative quality to the modified item, making it better or worse than a similar unmodified item. In any case, disagreements between the three criteria are limited; in almost all cases that other criteria are applicable, they assign the same orientation to the word. Differences in the orientation assignment typically result because one or more of the three criteria may not seem applicable at all to a given word (e.g., is *large* generally evaluatively preferable to *small*?).

As with many other similar linguistic characterizations, some of these assignments of orientation may seem arbitrary, or at least dependent on the particular person who performs them. For some words, this is reflected in disagreements between humans who are called to perform the classification task (although, even in these cases, there is usually a clear majority for one of the labels). But for most words, as I empirically show later (Section 7.4), there is a clear distinction of orientation; for example, everyone I have asked to evaluate orientation assignments, either formally or not, has agreed that *beautiful* carries a positive orientation and *ugly* a negative one.

It should be noted here that a more precise division of words according to semantic orientation would involve three categories: positive, negative, and neutral. The latter characterization would apply to words that lexicalize the neutral element (zero) of a scale, e.g., *medium* in the example {*small*, *medium*, *large*} that was discussed in Section 6.1.

Nevertheless, even then the neutral term will be either positive or negative relative to another word in the scale when these two words are opposed. Furthermore, the number of words that should be assigned neutral polarity is very small [Hubler 1983]; as Bolinger [1977] notes, "...We explicitly lexicalize only concepts that deviate from the norm; the lack of deviation is linguistically uninteresting." Consequently, I have elected to proceed with a two-valued system for orientation, and thus a two-way partition of the scalar groups into positive and negative sides, rather than attempt a three-way partition with a separate category for the rare neutral words. If a reliable automatic method for the identification of neutral words is discovered in the future, a postprocessing step can remove such words from the positive or negative subgroups.

7.2 Orientation as an objective linguistic property

The definition of orientation given above allows, like many linguistic definitions, for different interpretations when it is applied to different words. This sometimes results in different assignments of orientation labels for the same word by different persons, and thus calls into question whether the whole concept of semantic orientation has a factual, objective basis. I describe here an empirical analysis of the agreement between independently produced orientation labels for a large set of adjectives, which establishes beyond doubt that semantic orientation has an objective status in the language.

I based this analysis on the collection of adjectives that I also used for the verification of the conjunction hypothesis and the training and testing of the learning models discussed later in this chapter. For all these tasks, we need a set of adjectives with predetermined (correct) orientation labels. I constructed this set starting from the 21 million word 1987 Wall Street Journal corpus, augmented with part-of-speech information as described in Sections A.2 and A.3 of the Appendix. I extracted all adjectives appearing in the corpus 20 times or more, and then removed the adjectives that have no orientation. These are typically members of groups of complementary, qualitative terms [Lyons 1977], e.g., *domestic* or *medical*.

I then assigned an orientation label (either + or −) to each adjective, using the evaluative approach. As noted earlier, the criterion was whether the use of this adjective produces a description that is in general better or worse than the corresponding description without this modifier. I was unable to reach a unique label out of context for several adjectives; for example, *cheap* is positive if it is used as a synonym of *inexpensive*, but negative if it implies inferior quality. I removed these adjectives from consideration, but in general I tried to maximize the number of adjectives included in the analysis, keeping an adjective if I felt that it had a clear orientation in most of its senses and expected uses. The operations of selecting adjectives and assigning labels were performed before testing the conjunction hypothesis or implementing any other algorithms, to avoid any influence on the assigned labels. The final set contained 1,336 adjectives (657 positive and 679 negative terms), covering 41.2% of the total number of 3,243 adjectives that were otherwise eligible according to the frequency criterion. Figure 7.1 shows ten positive and ten negative randomly chosen terms from the set of selected and labeled adjectives. Most of the adjectives that were

Positive:	adequate central clever famous intelligent remarkable reputed sensitive slender thriving
Negative:	contagious drunken ignorant lanky listless primitive strident troublesome unresolved unsuspecting

Figure 7.1: Randomly selected adjectives with positive and negative orientations.

judged not to have a semantic orientation were descriptive compound adjectives, frequently domain dependent, e.g., *arbitrage-related*, *investment-grade*, *self-described*, and *twin-engine*. 716 of the total 3,243 adjectives (22.08%) are such compounds.

This set of 1,336 adjectives and the corresponding positive or negative orientation labels are subsequently used to verify the hypothesis about constraints on the orientations of conjoined adjectives, and also to train and evaluate the algorithms that measure orientation similarity and actually partition the words into the two big classes that correspond to the two possible orientations. To further validate the labels I assigned to the adjectives, and thus establish that this task can be performed objectively, I asked four other persons to independently label a randomly drawn sample of 500 of these adjectives.[6] They agreed with me that the positive/negative concept applies to 89.15% of these adjectives on average. For the adjectives where a positive or negative label was assigned by both me and the independent evaluators, the average agreement on the label was 97.38%. The average inter-evaluator agreement on labeled adjectives was 96.97%. These results are extremely significant statistically and compare favorably with validation studies performed for other tasks (e.g., sense disambiguation) in the past. They show that positive and negative orientation are objective

[6]I thank Kathy McKeown, Gregory Sean McKinley, Dragomir Radev, and Eric Siegel, who provided these alternative models of orientation for the adjectives in this sample.

properties that can be reliably determined by humans.

7.3 Direct and indirect indicators of semantic orientation

Direct indicators of positive or negative semantic orientation are unfortunately sparse and not always reliable. Certain words inflected with negative affixes (such as *in-* or *un-*) tend to be mostly negative, but this rule applies only to a fraction of the negative words. Furthermore, there are words so inflected which have positive orientation, e.g., *independent* and *unbiased*. Therefore, we cannot expect to directly extract information about the orientation of words; rather, we have to rely on *indirect* indicators that provide information on whether two words have the same or different orientation.

Antonymy and the semantic relationships that indicate identical polarity have been extensively studied in the linguistics literature (e.g., [Ogden 1932; Bierwisch 1969; Katz 1972; Leech 1975; Kempson 1977; Lyons 1977]). However, all tests proposed for determining the existence of one of these relationships between two given words involve the insertion of the words in sentential frames and the subsequent judgement of the validity of the resulting sentence. While these tests can be applied by a human, they are not amenable to automatic implementation.

In the absence of established methods for the determination of the sameness or difference of orientation between two words, I rely on two indicators that can be automatically retrieved from the corpus. The first of these indicators is the existence of a formal (morphological) relationship between two words. More precisely, one of the two adjectives may be derived from the other with the addition of a negative prefix (e.g., *compatible* $\rightarrow$ *incompatible*, *popular* $\rightarrow$ *unpopular*). When such a relationship exists between the two words, their orientations are almost always different.

The above method gives indicators of antonymy with high precision, but is only applicable to morphologically related antonyms. While the majority of adjectival antonyms are related morphologically [Lyons 1977, p. 275] there also many formally unrelated antonyms (e.g., *good–bad*, *deep–shallow*). To complement the information provided by the morphology-based indicator, I use evidence from the occurrence of two adjectives in conjunctions. I have observed that scalar adjectives coordinated by *and* and *or* tend to satisfy one of the following constraints: either they belong in the same scale and are of opposite polarities (e.g., *good and bad*, *hot or cold*), or they belong in different scales and have the same orientation. Conjunctions such as *fair and legitimate* or *corrupt and brutal* (extracted from the 1987 Wall Street Journal corpus) are acceptable, but a conjunction involving two words of different orientation that are not antonyms is not acceptable; compare the above examples with **corrupt and legitimate* and **fair and brutal* (or the other cross-products of these conjunctions) which are semantically anomalous.

Conjunctions employing the remaining central coordinator, *but*, typically involve two adjectives from different semantic groups with different orientations, e.g., *small but significant* and *attractive but worthless*. Conjunctions involving *either–or* or *neither–nor* tend to behave in the same way as conjunctions involving *or*.[7] However, the occurrence of two antonyms (as opposed to two unrelated words of the same orientation) seems to be higher with these coordinators than with *or*.

The observations made above continue to hold if explicit negation is introduced in one of the conjoined adjectival phrases with the particle *not* or with some adverbial modifier with negative connotation (e.g., *hardly*, *scarcely*). A conjunction of the form *not X and Y*, which is interpreted by the system as [[*not X*] *and* [*not Y*]] (cf. Section A.4.3 in the Appendix), may contain two antonymous words from the same scale (e.g., *not good and bad*) or two unrelated words of the same polarity (e.g., *not naive and foolish*), although conjunctions of the second type are more frequent in the scope of negation. A conjunction of the form *X and not Y* usually contains two words of different polarity (e.g., *inappropriate and not successful*), usually from different scales although sometimes two antonyms appear in an emphatic construct (e.g., *white and not black*). Conjunctions involving *or* and *either–or* exhibit similar behavior with conjunctions involving *and*, while additional explicit negation does not occur with adjectival phrases conjoined by *neither–nor*. For conjunctions involving *but* and explicit negation, I observe that many times they contain two adjectives of the same polarity from different scales, e.g., *good but not strong*, *shabby but not squalid*. But they also frequently involve two adjectives from the same side of a scale, in which case the semantically stronger term is listed second and explicitly negated; e.g., *difficult but not impossible*, *rare but not unprecedented*. Less frequently, they may also involve two antonyms from the same

[7]The coordinator *or* usually implies mutual exclusion, unlike to what happens in formal logic. Consequently, the use of *either–or* and *neither–nor* is similar to the use of *or*, aside from the emphasis placed on the exclusive character of the disjunction (and the negation implied by *neither–nor*).

scale. In that case, the first adjectival phrase is negated while the second is not, and the first phrase serves to add to the force of the second; e.g., *not new but old*, *not chaotic but steady*.

These observations can be explained in part by noting that the meaning of a conjunction such as *X and Y* involving one of the coordinators *and* and *or* can be either combinatory (i.e., the modified term has at the same time both of the attributes described by X and by Y) or segregatory (i.e., one of the attributes associated with X and Y but not both apply to the modified item).[8] The combinatory interpretation is the most common one in a phrase such as *resourceful and energetic managers*; it attributes to *managers* both properties of resourcefulness and being energetic, while the segregatory interpretation would consider two distinct subsets of *managers*, with the *managers* belonging to one being *resourceful* and the *managers* belonging to the other being *energetic*. Only the segregatory interpretation is possible when the two adjectives are incompatible, as in the phrase *good and evil deeds*, or for an example involving non-gradable adjectives, *any company, public or private, must ...* When the combinatory interpretation is applicable, the words involved belong to different scales; this is mandatory since words from the same scale are mutually exclusive (cf. Section 1.7.5). On the other hand, when the segregatory interpretation is applicable, the words may belong in the same or different scales. However, the facts that words from the same scale that are conjoined usually are antonyms and that unrelated words that are conjoined

[8]A third interpretation is possible for a few adjectives, namely adjectives of color. In this interpretation, the two adjectives do not simultaneously apply to the entire object they modify, but each applies to parts of it. So in the phrase *blue and white sheets* the meaning of the conjunction can be segregatory (some blue sheets and some white sheets) or be interpreted to specify "sheets that are partly colored blue and partly colored white". The combinatory interpretation is blocked with color adjectives since they are mutually incompatible.

have the same orientation, are not explainable from semantic constraints. Rather, they seem to be pragmatic constraints on language use, which, to the best of my knowledge, have not been studied before.

The prevalence of pairs of antonyms among conjunctions involving *either–or* and *neither–nor* can be attributed to the necessarily segregatory interpretation of these conjunctions, which is made explicit with the use of the introductory *either* or *neither*. Note that while these conjunctions typically contain antonymous words (e.g., *either formal or informal*), they can also contain two unrelated words that express two alternatives; e.g., *neither feasible nor desirable*.

Conjunctions involving *but* usually contain two semantically unrelated words since *but* allows only a combinatory interpretation; X *but* Y is equivalent to X *but at the same time* Y. The two words are of different orientations because *but* introduces contrast between the two conjuncts. The special case X *but not* Y where X and Y are two words from the same scale that satisfy the constraint $X \prec Y$ follows these restrictions of polarity and compatibility, since *not* Y is true for something for which X is true and expressed[9]; note that at most one of X *but not* Y and Y *but not* X can be true, in accordance with the fact that $\prec$ is an order relation.

Finally, all statements made above for conjunctions that include explicit indicators of negation are in agreement with the justifications given earlier for the simpler cases of conjunctions involving non-negated adjectival phrases, after the semantic (logical) effect of the

[9]According to conversational implicature principles; cf. Section 1.7.4.

negation operator is taken into account.

The above observations about the information contained in conjunctions of adjectival phrases can be exploited to provide an additional source for the determination of the sameness or difference of the orientations of two adjectives. A straightforward method is to consider the scales formed after the application of the previous two stages of the system, and interpret conjunctions between two adjectives as follows:

- If the conjunction involves *and*, *or*, *either–or*, or *neither–nor* and the two conjoined adjectives belong to the same scale, then the two words are antonyms.
- If the conjunction involves *and*, *or*, *either–or*, or *neither–nor* and the two conjoined adjectives belong to different scales, then the two words have the same polarity.
- If the conjunction involves *and*, *or*, or *either–or* and also contains one or more explicit indicators of negation, then the two words are interpreted according to the two rules above appropriately modified to account for the negation, as discussed in the preceding analysis.
- If the conjunction involves *but* and no indicators of negation, the two conjoined words should be in different scales and with different polarities.
- If the conjunction involves *but* and an explicit indicator of negation for the first conjunct, the two conjoined adjectives may belong to the same or different scales but have different polarities.
- If the conjunction involves *but* and an explicit indicator of negation for the second

conjunct, the two conjoined adjectives have the same polarity. Furthermore, this pattern also offers evidence of relative semantic strength.

However, the subsystems that form and filter scalar groups are not infallible (see Section 4.1.2 for examples of semantic groups of adjectives formed by the system and Section 4.1.3 for a discussion of evaluation outcomes on these partitions). Consequently, a more robust approach is to perform the orientation-based classification of the words independently of their placement into scales, aggregate information across as many words as possible, and then combine the produced orientation classification with the semantic groups to identify antonyms and synonyms. On the basis of the observations made above, I formulate the *conjunction hypothesis*, which, in its more general form, states that the connective and the semantic orientations of the participating open-class words in a conjunction are mutually constrained. The expected direction of the constraints is as discussed above; for example, conjunctions involving *and* are expected to contain in most cases two adjectives of the same orientation from different semantic groups, otherwise two antonyms.

In addition to the connective employed in a particular conjunction, we can also consider other sources of evidence before deciding on the combinatory or segregatory interpretation of a conjunction and thus on the kind of orientation constraint it provides. Such evidence is offered by the number of the noun phrase that is modified by the conjoined adjectives. Plural noun phrases allow both interpretations, since there are several objects denotated by the noun phrase and some of them may be modified by one adjective while the remaining are modified by the second adjective. On the other hand, singular noun phrases denote a single physical object or entity, and therefore force the combinatory interpretation.[10] For example, the phrase *confused and angry voters* may refer to voters that are confused and angry at the same time, or to some confused voters and some angry voters.[11] By contrast, a *confused and angry voter* is always a single person who possesses both these attributes.

The surface syntactic manifestation of the modification relationship can also affect the nature or strength of the semantic constraint on orientation. Adjectives can modify nouns as pre-modifiers (*attributive modification*), through linking copular or resultative verbs (*predicative* and *resultative modification*), and through apposition (*appositive modification*). I have noted that some of these surface forms of the modification relationship constrain further the orientations of the participating words; for example, appositive conjunctions involving the coordinators *and* and *or* have a greater tendency to contain two words of opposite orientation (antonyms) than other conjunctions with the same coordinators.

In addition to number and modification type, explicit indicators can be identified and used to appropriately classify conjunctions as combinatory or segregatory. The use of the predeterminer *both* as a premodifier of conjunctions is an example of this type of information. Interestingly, *both* disambiguates the meaning of the conjunction when used together with the type of modification; depending on the latter, the use of *both* can uniquely select the segregatory or the combinatory meaning. If the modification is attributive, then *both* forces a segregatory interpretation, while if the modification is predicative, the presence of *both*

[10] Adjectives of color are again an exception, as we can have more than one such adjectives applying simultaneously to (parts of) a single object. For example, *red, white, and blue flag*.

[11] Although, in this example the first (combinatory) interpretation is preferable because of the semantic association between *confused* and *angry*.

forces a combinatory interpretation. Consider the following examples:

	Their employees are both indifferent and rude.	*predicative, combinatory*
	Both indifferent and rude employees are hired by this company.	*attributive, segregatory*[12]
*	Their decisions are both moral and immoral.	*predicative, combinatory*[13]
	Everyone makes both moral and immoral decisions.	*attributive, segregatory*

To summarize, I have presented several factors that affect the direction and strength of the semantic constraints that conjunctions impose on the orientation of the participating open-class words:

- The connective employed (*and*, *or*, *but*, *either–or*, or *neither-nor*).
- The presence and position of negation operators (i.e., whether they apply to the first, second, or both conjuncts). These operators can be explicit (such as *not*) or implicit (such as *hardly*).
- The number of the modified noun or head of noun phrase (plural or singular) for conjunctions of adjectival modifiers.
- The surface form of the modification relationship between conjoined adjectives and the noun they modify (attributive, predicative, resultative, or appositive).

[12]Note that the use of *both* overrides the normal preference for the combinatory meaning which is selected because of the correlation between *indifferent* and *rude* (at least as applied to a company setting). Compare with the sentence *Indifferent and rude employees are hired by this company.*

[13]In this case, the combinatory interpretation selected by *both* clashes with the semantic incompatibility of the antonyms *moral* and *immoral*, resulting in the unacceptability of the sentence.

- Additional explicit markers, such as the predeterminer *both* in conjunctions of adjectives.

In addition to these local indicators, we can also take advantage of the transitivity of the "same orientation" relationship, and of the fact that there are only two possible orientations. These two properties impose constraints on the validity of assignments of relative orientation. For example, we cannot at the same time claim that X and Y, and Y and Z, have pairwise the same orientation while Z has different orientation from X, because the transitivity constraint is violated. Similarly, we cannot simultaneously claim that X has different orientation from Y and from Z, and that Y and Z also have different orientations, because that would require the existence of at least three distinct orientation classes. Consequently, we can use a specialized version of a clustering algorithm similar to the one described in Section 2.7.1 to locate the assignment of words to classes that violates the least number of constraints derived from the conjunctions around adjectives. I return to this issue in Section 7.6, after discussing first how the strength of the constraints from conjunctions and the orientation similarity of any two adjectives can be quantified and estimated from the corpus.

7.4 Validation of the conjunction hypothesis

In the previous section I presented an analysis of the expected direction of the constraints that the choice of a connective imposes on the semantic orientation of the participating open-class words. This analysis was based on my linguistic intuition and on numerous examples drawn from the 1987 Wall Street Journal corpus[14], but not on a full quantitative analysis of all adjectival conjunctions it contains. In this section I report the results of such an exhaustive analysis, using the set of 1,336 labeled adjectives that I constructed from the corpus (see Section 7.2).

To extract conjunctions between adjectives, I use a two-level finite-state grammar, which covers complex modification patterns and noun-adjective apposition. This grammar is described in Section A.4.2 of the Appendix. Running the parser on the 21 million word corpus, I collected 13,601 conjunctions of adjectives. Very few of these contained explicit modification by *both* or negative markers; for example, only 175 conjunctions (1.29%) contained negation. Given the added complexity necessary to interpret conjunctions of negated adjectives and the potential scope interpretation errors, I decided to limit the further analysis to the non-negated adjectives only. Similarly, I eliminated from consideration the few occurrences of *both* as a premodifier of adjectival conjunctions to simplify the analysis.

The collection process produced 13,426 conjunctions of non-negated adjectives, expanding to a total of 15,431 conjoined adjective pairs. 383 of these pairs contained the same adjective in different degrees, or had two occurrences of the same adjective in the comparative degree (e.g., *better and better*) indicating a progressive feature; these pairs are useless for our purposes, and were removed from the analysis. After morphological transformations to the base form of each adjective and conflation of the resulting base-form pairs, the remaining 15,048 conjunction *tokens* involve 9,296 distinct pairs of conjoined adjectives

---

[14] All semantically valid examples given in the previous section were actually observed in the corpus.

(*types*). Each conjunction token is classified by the parser according to the three variables that are expected to influence the direction or strength of the constraint on semantic orientation: the conjunction used (*and*, *or*, *but*, *either-or*, or *neither-nor*), the type of modification (attributive, predicative, appositive, resultative), and the number of the modified noun (singular or plural). These three variables collectively represent the syntactic context of the conjunction.

Using the three attributes extracted by the parser, I constructed a cross-classification of the conjunctions in a three-way table. I counted types and tokens of each conjoined pair that had both members in the set of 1,336 pre-selected labeled adjectives discussed above; 2,748 (29.56%) of all conjoined pairs (types) and 4,024 (26.74%) of all conjunction occurrences (tokens) met this criterion. I augmented this table with marginal totals, arriving at 90 categories, each of which represents a triplet of attribute values, possibly with one or more "don't care" elements.

I then measured the percentage of conjunctions in each category with adjectives of same or different orientations. For each category, I compared the observed distribution against the null hypothesis that the proportions of adjective pairs (types) of same and different orientation in that category are equal. Under the null hypothesis, the number of same- or different-orientation pairs in each category follows a binomial distribution with $p = 0.5$ [Conover 1980].

I show in Table 7.1 the results for several representative categories rather than all 90, and summarize all results below:

Conjunction category	Conjunction types analyzed	% same-orientation (types)	% same-orientation (tokens)	P-Value (for types)
All conjunctions	2,748	77.84%	72.39%	$< 1 \cdot 10^{-16}$
All *and* conjunctions	2,294	81.73%	78.07%	$< 1 \cdot 10^{-16}$
All *or* conjunctions	305	77.05%	60.97%	$< 1 \cdot 10^{-16}$
All *but* conjunctions	214	30.84%	25.94%	$2.09 \cdot 10^{-8}$
All attributive *and* conjunctions	1,077	80.04%	76.82%	$< 1 \cdot 10^{-16}$
All predicative *and* conjunctions	860	84.77%	84.54%	$< 1 \cdot 10^{-16}$
All appositive *and* conjunctions	30	70.00%	63.64%	0.04277

Table 7.1: Validation of the conjunction hypothesis. The P-value column lists the probability that similar or more extreme results would have been obtained if same- and different-orientation conjunction types were actually equally distributed.

- The conjunction hypothesis is validated overall and for almost all individual cases. The results are extremely significant statistically, except for a few cases where the sample is small.

- Aside from the use of *but* with adjectives of different orientations, there are, rather surprisingly, small differences in the behavior of conjunctions between linguistic environments (as represented by the three attributes). There are a few exceptions, e.g., appositive *and* conjunctions modifying plural nouns are evenly split between same and different orientation. But in these exceptional cases the sample is very small, and the observed behavior may be due to chance.

- Further analysis of different-orientation pairs in conjunctions other than *but* shows that conjoined antonyms are far more frequent than expected by chance, in agreement with the linguistic analysis of Section 7.3 and with [Justeson and Katz 1991].

7.5 Prediction of same or different orientation links for two conjoined adjectives

The analysis of the observed conjunctions in the previous section suggests a baseline method for classifying links between conjoined adjectives as being of the same- or different-orientation type: since 77.84% of all links from conjunctions indicate same orientation, we can achieve this level of performance by always guessing that a link is of the same-orientation type. However, we can improve performance by noting that conjunctions using *but* exhibit the opposite behavior, usually involving adjectives of different orientations. Thus, a revised but still simple rule predicts a different-orientation link if the two adjectives have been seen in a *but* conjunction, and a same-orientation link otherwise, assuming the two adjectives were seen connected in at least one conjunction.

Morphological relationships between adjectives also play a role. Adjectives related in form (e.g., *adequate–inadequate* or *thoughtful–thoughtless*) almost always have different semantic orientations. I implemented a morphological analyzer which matches adjectives related in this manner. This process is highly accurate, but unfortunately does not apply to many of the possible pairs: in the reference set of 1,336 labeled adjectives (891,780 possible pairs), 102 pairs are morphologically related; among them, 99 are of different orientation, yielding 97.06% accuracy for the morphology method. This information is orthogonal to that extracted from conjunctions: only 12 of the 102 morphologically related pairs have been observed in conjunctions in the corpus. Thus, I add to the predictions made from conjunctions the different-orientation links suggested by morphological relationships.

We can improve the accuracy of classifying links derived from conjunctions as same or different orientation with a log-linear regression model [Santner and Duffy 1989], exploiting the differences between the various conjunction categories. This is a generalized linear model [McCullagh and Nelder 1989] with a linear predictor

$$\eta = \mathbf{w}^{\mathrm{T}}\mathbf{x}$$

where $\mathbf{x}$ is the vector of the observed counts in the various conjunction categories for the particular adjective pair we try to classify and $\mathbf{w}$ is a vector of weights to be learned during training. The response y is non-linearly related to η through the *inverse logit* function,

$$y = \frac{e^{\eta}}{1 + e^{\eta}}$$

Note that $y \in (0, 1)$, with each of these endpoints associated with one of the possible outcomes (i.e., same or different orientation). A similar model (with different predictors) was used for determining whether a particular adjective is gradable (Section 6.3); that section, as well as Section 5.3.2 on linear models in general, provide additional details about the construction, training, and variants of linear and log-linear learning models.

As noted earlier, categories of conjunctions are represented via three attributes: the conjunction employed, which can take five possible values (*and*, *or*, *but*, *either–or*, or *neither–nor*); the number of the modified noun, which can take two possible values (*singular* or *plural*); and the type of modification, which can take four possible values (*attributive*, *predicative*, *resultative*, or *appositive*). Consequently, we have $5 \times 2 \times 4 = 40$ linearly independent variables, each corresponding to the number of conjunctions falling in a specific category. When we aggregate these counts across the "conjunction employed" attribute (i.e., when we replace that attribute's specific values with a "don't care" condition), we get $2 \times 4 = 8$ additional variables; similarly, we get $5 \times 4 = 20$ additional variables by aggregating across the "number" attribute, and $5 \times 2 = 10$ additional variables when we aggregate across the "modification type" attribute. We also aggregate pairs of attributes, essentially differentiating the conjunctions on the basis of a single attribute. This creates 5 variables when only the conjunction used is considered, 2 variables when only the number of the modified noun is considered, and 4 variables when only the modification type is considered. A final variable is produced by aggregating across all three attributes, and gives the total count of conjunctions between a particular pair of adjectives, irrespective of their syntactic context.

This process results in $40 + 8 + 20 + 10 + 5 + 2 + 4 + 1 = 90$ possible predictor variables, 40 of which are linearly independent. Since using all the 40 independent predictors[15] invites overfitting [Duda and Hart 1973], I have investigated subsets of the full log-linear model for the corpus data using the method of *analysis of deviance* with *iterative stepwise refinement* [Draper and Smith 1981; Miller 1990]. An initial model is constructed and is subsequently revised in an iterative manner. Residual deviance, a measure of the quality of the fit, is used to score each model and variables are added or dropped if their contribution to the reduction or increase of the residual deviance compares favorably to the resulting loss or

[15] Note that we don't have to use the original 40 variables that have specific values for all three attributes. There are many subsets of the 90 predictors (each with 40 members or less) that include variables that are aggregates of the original categories and still contain linearly independent variables.

Prediction method	Morphology used?	Accuracy on reported same-orientation links	Accuracy on reported different-orientation links	Overall accuracy
Always predict same orientation	No	77.84%	–	77.84%
	Yes	78.18%	97.06%	78.86%
But rule	No	81.81%	69.16%	80.82%
	Yes	82.20%	78.16%	81.75%
Log-linear model	No	81.53%	73.70%	80.97%
	Yes	82.00%	82.44%	82.05%

Table 7.2: Accuracy of several prediction models for classifying links between adjectives as same- or different-orientation.

gain of residual degrees of freedom (a measure of the complexity of the model). This process led to the selection of nine predictor variables.

I evaluated the three prediction models discussed above with and without the secondary source of morphology relations. For the log-linear model, I repeatedly partitioned the data into equally sized training and testing sets, estimated the weights on the training set, and scored the model's performance on the testing set, averaging the resulting scores.[16] This was done 20 times and the results averaged, to eliminate any random effects from the training/testing set partition. Table 7.2 shows the results of these analyses and confirms that the log-linear model does indeed outperform, by a small margin, the simple *but* prediction rule. Although the log-linear model offers only a small improvement on pair classification than the simpler *but* rule, it confers the important advantage of *rating* each prediction between 0 and 1. I make extensive use of this property in the next phase of the algorithm.

[16]When morphology is to be used as a supplementary predictor, I remove the morphologically related pairs from the training set.

7.6 Finding groups of adjectives with the same orientation

The third phase of the subsystem that separates words according to orientation assigns the adjectives into groups, placing adjectives of the same (but unknown) orientation in the same group. Each pair of adjectives has an associated orientation dissimilarity value between 0 and 1; adjectives connected by same-orientation links have low dissimilarities, and conversely, different-orientation links result in high dissimilarities. Adjective pairs with no connecting links are assigned the neutral dissimilarity 0.5.

The baseline and *but* methods make qualitative distinctions only (i.e., same-orientation, different-orientation, or unknown); for them, I define dissimilarity for same-orientation links as one minus the probability that such a link classification is correct and dissimilarity for different-orientation links as the probability that such a classification is correct. These probabilities are estimated from separate training data. Note that for these prediction models, dissimilarities are always identical for similarly classified links.

The log-linear model, on the other hand, offers an estimate of how good each individual prediction is, since it produces a value y between 0 and 1. I construct the model so that 1 corresponds to same-orientation, and define dissimilarity as one minus the produced value.

In all cases, I make an exception for links connecting morphologically related adjectives. Since the system is much more accurate in identifying the different-orientation links between such adjectives than for other links, I set their orientation dissimilarity to a constant high value (10 in the experiments reported in this chapter).

Same and different-orientation links between adjectives form a graph, where nodes correspond to adjectives and edges to links inferred from conjunctions or morphology. To partition the graph nodes into subsets of the same orientation, I employ an iterative optimization procedure on each connected component, based on the *exchange method*, a non-hierarchical clustering algorithm [Späth 1985]. The same algorithm, which I briefly review here, was used for the problem of partitioning a collection of words into semantic groups (Section 2.7.1); in the present section, I utilize the additional knowledge that each connected component of the graph must be separated into exactly two groups to produce a faster and more sophisticated version of the algorithm, using an additional constraint that validates the clustering decisions on the fly.

I define an *objective function* Φ scoring each possible partition $\mathcal{P}$ of the adjectives into two subgroups C_1 and C_2 as

$$\Phi(\mathcal{P}) = \sum_{i=1}^{2} \left(\frac{1}{|C_i|} \sum_{\substack{x,y \in C_i \\ x \neq y}} d(x,y) \right)$$

where $|C_i|$ stands for the cardinality of cluster i, and $d(x,y)$ is the orientation dissimilarity between adjectives x and y. I want to select the partition $\mathcal{P}_{\min}$ that minimizes Φ, subject to the additional constraint that for each adjective x in a cluster C,

$$\frac{1}{|C|-1} \sum_{\substack{y \in C \\ x \neq y}} d(x,y) \quad \leq \quad \frac{1}{|\overline{C}|} \sum_{y \in \overline{C}} d(x,y) \tag{7.1}$$

where $\overline{C}$ is the complement of cluster C, i.e., the other member of the partition. This constraint, based on Rousseeuw's [1987] *silhouettes*, helps correct wrong cluster assignments.

To find $\mathcal{P}_{\min}$, I first construct a random partition of the adjectives, then locate the adjective that will most reduce the objective function if it is moved from its current cluster to the other one. I move this adjective and proceed with the next iteration until no movements can improve the objective function. At the final iteration, the cluster assignment of any adjective that violates constraint (7.1) is changed. As with the more general case with an unrestricted number of clusters, this is a steepest-descent hill-climbing method, and thus is guaranteed to converge. But it will also find, in general, a local minimum rather than the global one; the problem is NP-complete [Garey and Johnson 1979] (cf. Section 2.7.2). We can arbitrarily increase the probability of finding the globally optimal solution by repeatedly running the algorithm a fixed number of times with different starting partitions; in the current implementation, the optimization is repeated for 1,000 random starting partitions.

7.7 Results and evaluation

The clustering algorithm separates each component of the graph into two groups of adjectives, but does not actually label the adjectives as positive or negative. This problem is the topic of the next chapter; until methods for addressing it have been presented, I decouple the evaluation of the subsystem that separates adjectives according to orientation from the overall evaluation of how successfully we predict the orientation of each adjective. I accomplish this by assuming that the assignment of positive or negative labels to the groups produced by this stage is always optimal, i.e., that the group that has the most positive terms will be labeled positive. This assumption is not unwarranted; as we shall see, the subsystem that performs the labeling task always assigned the correct label to these groups in the experiments performed.

Since graph connectivity affects performance, I devised a method of selecting test sets that makes this dependence explicit. Note that the graph density is largely a function of corpus size, and thus can be increased by adding more data. Nevertheless, I report results on sparser test sets to show how the classification algorithm scales up.

The evaluation of the word separating stage is based on the sets of adjectives A of Section 7.2 (containing 1,336 adjectives) and conjunction- and morphology-based links L between these words (containing 2,838 links). I separated these sets into training and testing groups by selecting, for several values of the parameter α, the maximal subset of A, A_α, which includes an adjective x if and only if there exist at least α links from L between x and other elements of A_α. This operation in turn defines a subset of L, L_α, which includes all links between members of A_α. I train the log-linear model of Section 7.5 on $L - L_\alpha$ (excluding links between morphologically related adjectives), compute predictions and dissimilarities for the links in L_α, and use these to classify and label the adjectives in A_α. α must be at least 2, since we need to leave some links for training.

Table 7.3 shows the results of these experiments for $\alpha = 2$ to 5. The system produced the correct classification between 78% of the time on the sparsest test set up to more than 92% of the time when a higher number of links was present. As noted above, this measure-

α	Number of adjectives in test set ($\lvert A_\alpha \rvert$)	Number of links in test set ($\lvert L_\alpha \rvert$)	Average number of links for each adjective	Accuracy
2	730	2,568	7.04	78.08%
3	516	2,159	8.37	82.56%
4	369	1,742	9.44	87.26%
5	236	1,238	10.49	92.37%

Table 7.3: Evaluation of the subsystem that constructs groups of adjectives that share the same (but unknown) orientation. The adjective sets for which evaluation results are reported are representative of different densities of the graphs induced by conjunction-based links, as encoded by the parameter α. The evaluation results reported here measure only how many of the adjectives have been assigned in their optimal group, i.e., assume that the produced groups will be labeled correctly during the next stage; this assumption is correct for all these adjective sets (cf. Table 8.5 on page 330).

ment of performance ignores the label assignment subproblem; thus it evaluates how well each adjective fits into the group it is placed in. The results are extremely significant statistically (P-value less than 10^{-16}) when compared with the baseline method of randomly assigning orientations to adjectives, or the baseline method of always predicting the most frequent (for types) category (50.82% of the adjectives in the reference collection are classified as negative). Figure 7.2 shows some of the adjectives in set A_4 and their classifications (again with the assumption that the actual labeling of the groups as positive or negative is performed correctly; this assumption is borne out in practice for all sets A_2–A_5).

7.8 Graph connectivity and performance

One important characteristic of any learning approach is how well it scales up when more data is available. A strong point of the method I presented in this chapter is that decisions

Classified as positive:

bold decisive disturbing **generous good honest important large mature patient peaceful positive proud sound stimulating straightforward** strange **talented vigorous witty**

Classified as negative:

ambiguous cautious **cynical evasive harmful hypocritical inefficient insecure irrational irresponsible minor** outspoken pleasant **reckless risky selfish tedious unsupported vulnerable wasteful**

Figure 7.2: Sample retrieved semantic orientation information for some of the adjectives from set A_4. The determinations "classified as positive" and "classified as negative" are made under the assumption that the subsequent labeling stage will assign the correct labels to the identified groups (which is indeed the case). Correctly matched adjectives are shown in bold.

on individual words are aggregated to provide decisions on how to group words into a class that will be subsequently labeled positive or negative as a whole. Thus, the overall result can be much more accurate than the individual indicators. To verify this, I ran a series of simulation experiments. Each experiment measures how the classification algorithm performs for a given level of precision P for identifying links and a given average number of links k for each word. The goal is to show that even when P is low, given enough data (i.e., high k), we can achieve high performance for the grouping task.

As I noted earlier, the corpus data is eventually represented in the system as a graph, with the nodes corresponding to adjectives and the links to predictions about whether the two connected adjectives have the same or different orientation. Thus the parameter P in the simulation experiments measures how well we are able to predict each link independently of the others, and the parameter k measures the number of distinct adjectives each adjective appears with in conjunctions. P therefore directly represents the precision of the link classification algorithm (cf. Section 7.5), while k indirectly represents the corpus size.

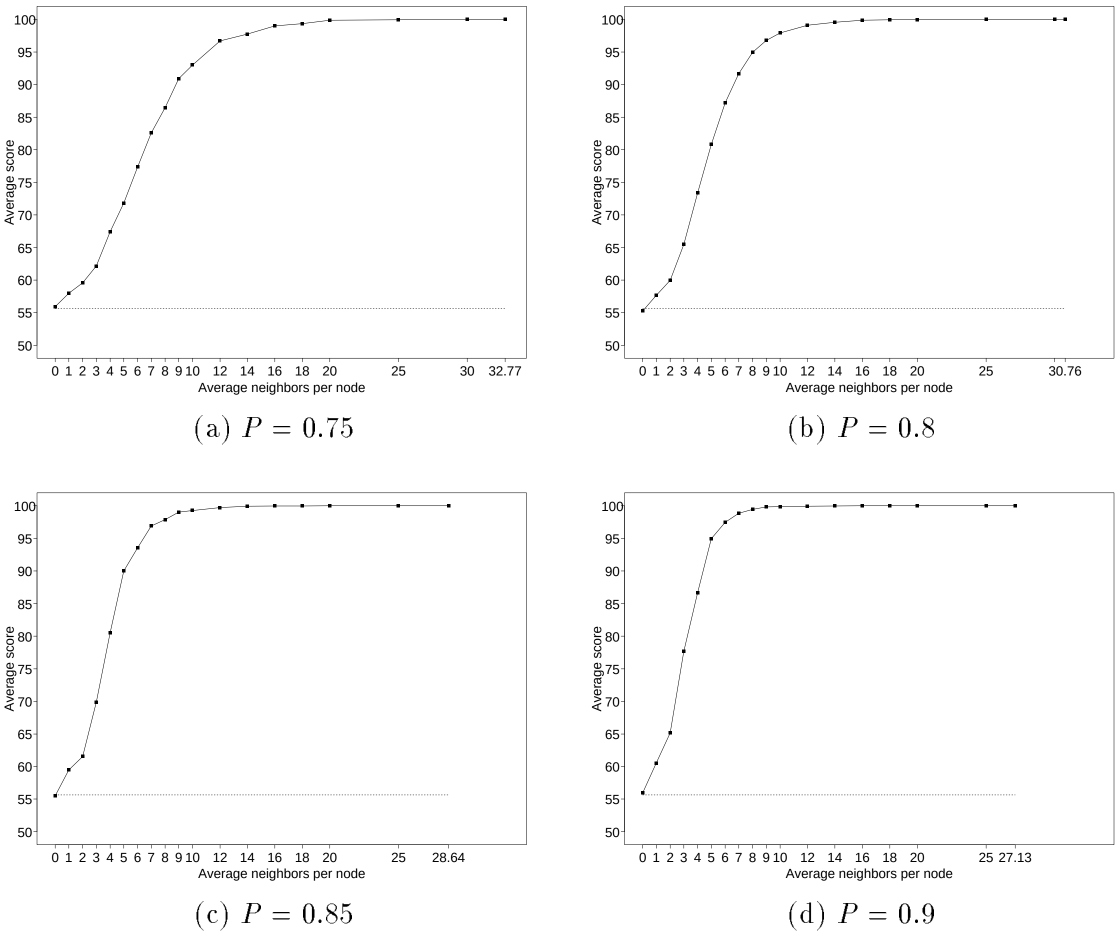
To measure the effect of P and k (which are reflected in the graph topology), I carry out a series of experiments where I systematically vary their values. For example, as k (or the amount of data) increases for a given level of precision P for individual links, I want to measure how this affects the overall accuracy of the resulting groups of nodes. Thus, I need to construct a series of data sets, or graphs, which represent different scenarios corresponding to a given combination of values of P and k. To do this, I construct a random graph by randomly assigning 50 nodes to the two possible orientations. Because we don't have observed links or morphology information on these abstract nodes, we cannot predict whether two nodes are of the same or different orientation. Rather, I randomly assign links between nodes so that, on average, each node participates in k links and $100 \times P\%$ of all links connect nodes of the same orientation.[17] Then I assume that these links have been identified by the link prediction algorithm as connecting two nodes with the same orientation (so that $100 \times P\%$ of these predictions will be correct). This is equivalent to the baseline link classification method, and provides a lower bound on the performance of the algorithm actually used in the system (Section 7.5), which actually computes a varying measure of orientation dissimilarity for each individual link. Note that by fixing P in advance, we evaluate the clustering component of the orientation subsystem, i.e., we investigate how well the words can be placed into categories given a certain quality of input information about the links ($P = 1$ corresponds to perfect knowledge of the links).

[17]In very dense graphs it may be impossible to have both P and k near the desired values; for example, we can't have $P = 1$ and k larger than approximately half the nodes, because all created links indicate same orientation and there are only so many nodes with the same orientation. In such cases of *saturated* graphs P takes precedence.

In a manner similar to the evaluation performed on real corpus data, I decouple the partitioning and labeling components of the system and score the partition found under the best matching conditions for the actual labels. Thus the simulation measures only how well the system separates positive from negative adjectives, not how well it determines which is which. However, in all the experiments performed with actual text (see Section 8.10 in the next chapter), the system correctly found the labels of the groups; any misclassifications in the overall orientation assignments came from placing an adjective in the wrong group. The whole procedure of constructing the random graph and finding and scoring the groups is repeated 200 times for any given combination of P and k, and the results are averaged, thus avoiding accidentally evaluating the system on a graph that is not truly representative of graphs with the given P and k.

I observe (Figure 7.3) that even for relatively low P, (i.e., when our ability to predict the category of a link is not that great), our ability to correctly classify the nodes approaches very high levels with a modest number of links. For $P = 0.8$, we need only about 7 links per adjective for classification performance over 90% and only 12 links per adjective for performance over 99%.[18] The difference between low and high values of P is in the rate at which increasing data increases overall precision. The simulation scores do not vary significantly for larger sets of words; I obtained 93% average classification accuracy for 1200 nodes and $k = 6$. These results are somewhat more optimistic than those obtained with real data (Section 7.7), a difference which is probably due to the uniform distributional assumptions in the simulation. Nevertheless, I expect the *trends* to be similar to the ones

[18] 12 links per adjective for a set of n adjectives requires $6n$ conjunctions between the n adjectives in the corpus.



(a) $P = 0.75$

(b) $P = 0.8$

(c) $P = 0.85$

(d) $P = 0.9$

Figure 7.3: Simulation results obtained on 50 abstract nodes, indicating how the performance of the semantic orientation classification subsystem is affected by the density of the graph in conjunctions (k) given a constant level of accuracy for identifying links between nodes (P). In each figure, the last x coordinate indicates the (average) maximum possible value of k for this P, and the dotted line shows the performance of a random classifier.

---

shown in Figure 7.3 and the results of Table 7.3 on real data support this expectation. This is important, as we can increase the actual average k, and consequently the performance of the system, by using a larger corpus.

Chapter 8

Labeling subscales as semantically positive or negative

In the previous chapter I presented an automatic method which, using information from conjunctions and morphology, classifies adjectives into semantic orientation classes, so that members of each class share the same (but unknown) orientation. That method was shown to perform satisfactorily with a modest number of observed conjunctions for each adjective, and to scale up well when more data becomes available. In this chapter I take up the remaining half of the approach I outlined at the start of Chapter 7, namely the task of determining which of the constructed classes contains the positive terms and which the negative ones. This is the final stage of the system I describe in this dissertation; when it is combined with the semantic grouping of the adjectives (Chapter 2) and the filtering of non-scalar groups (Chapter 6), it allows for the automatic identification of antonyms and near-synonyms.

Since there are no reliable and general direct indicators of semantic orientation, I base the classification of the orientation groups on another linguistic property, *semantic markedness*, which exhibits strong correlation with semantic orientation. I introduce this property in Section 8.1, and present several text-level indicators for it in the following section. These indicators have been proposed in the linguistics literature (although I had to make several of them more specific so that they can be automatically extracted from the corpus), but no empirical, quantitative study of their validity and effectiveness in predicting semantic markedness values had been made before.[1] Also, no formal investigation of the interactions between the various indicators had been carried out, thus no data existed to support conclusions about which indicators (and with what relative weights) should be used in a complex predictor for the determination of semantic markedness.

I thus devote most of this chapter (Sections 8.3–8.9) to presenting the organization and results of an exploratory analysis of the performance of the linguistic indicators for the automatic determination of semantic markedness in pairs of antonymous gradable adjectives. This analysis was designed to answer the above questions by measuring the performance of each proposed indicator independently of the others and investigating the combination of indicators that yields optimal performance.

Section 8.3 discusses the experimental setup, describing how collections of pairs of adjectives were selected as training and test sets and how data was collected for them from the Brown corpus [Kučera and Francis 1967; Francis and Kučera 1982] and the MRC Psycholinguistic Database [Coltheart 1981; Wilson 1987]. In Section 8.4 I present the accuracy scores

[1]This lack of quantitative data is not limited to semantic markedness, but also to other levels of markedness oppositions such as the syntactic level; cf. Section 8.2.

obtained for each indicator independently of other indicators, and I analyze their statistical significance. I proceed to discuss differences in the performance of these linguistic tests in the two strata of adjective pairs involving morphologically unrelated and morphologically related adjectives in Section 8.5, and the correlation between the various tests in Section 8.6. Subsequently, I discuss how the general learning approach of generalized linear regression (see Sections 6.3 and 7.5) can be applied to this problem to form complex predictors of semantic markedness, and introduce an alternative learning model for the same task, based on decision trees. In Section 8.8 I present the results of evaluating these complex predictors along with an analysis of their statistical significance. Finally, I conclude this exploratory analysis with some general comments about the results of the experiments, summarizing the findings about the linguistic tests and their value for linguistic analysis and practical prediction of semantic markedness.

The last section of the chapter returns to the initial problem of identifying the orientation group that contains the positive terms. I show how the results of the markedness analysis lead to the formulation of a simple criterion for this problem, and present the results produced when this criterion is applied to the adjective collections of the previous chapter (Section 7.7). For all these adjective sets, the markedness-based criterion correctly identified the group that contained the positive terms.

8.1 Semantic markedness

When I first sought methods for determining the semantic orientation of words automatically from corpus data, I mentioned that direct indicators for this property are limited in scope and not very reliable (Section 7.3); consequently, I adopted the two-stage approach that first separates the words into orientation groups using indirect indicators and then labels each group as positive or negative. For this second task, I also use an indirect approach, namely that *semantic markedness* can be predicted reliably once pairs of words (or pairs of groups of words) with different orientations have been identified. The orientation labeling decisions are thus based on criteria developed for determining semantic markedness.

Semantic markedness is an important linguistic property in its own right; its value for this application is that it is closely associated with semantic orientation, the unmarked member in an opposition almost always having positive orientation [Lehrer 1985; Battistella 1990]. The concept of *markedness* originated in the work of Prague School linguists [Jakobson 1984a] and refers to relationships between two complementary or antonymous terms which can be distinguished by the presence or absence of a feature (+A versus −A). Such an opposition can occur at various linguistic levels. For example, a markedness contrast can arise at the morphology level, when one of the two words is derived from the other and therefore contains an explicit formal marker such as a prefix; e.g., *profitable–unprofitable*. Markedness contrasts also appear at the semantic level in many pairs of gradable antonymous adjectives, such as *tall–short*. The marked and unmarked elements of such pairs function in different ways. The unmarked adjective (e.g., *tall*) can be used in how-questions to refer to the property described by both adjectives in the pair (e.g., *height*), but without any implication about the modified item relative to the norm for the property. For example, the question *How tall is Jack?* can be answered equally well by four or seven feet. In contrast, the marked element of the opposition cannot be used generically; when used in a how-question it implies a presupposition on the part of the speaker regarding the relative position of the modified item on the adjectival scale. Thus, the corresponding question using the marked term of the opposition (*How short is Jack?*) conveys an implication on the part of the speaker that Jack is indeed short; the distinguishing feature A expresses this presupposition.

While markedness has been described in terms of a distinguishing feature A, its definition does not specify the type of this feature. Consequently, several different types of features have been employed, resulting in oppositions at different linguistic levels. This, in turn, has led to some confusion about the meaning of the term *markedness*. Following Lyons [1977], I distinguish between *formal markedness* where the opposition occurs at the morphology level (i.e., one of the two terms is derived from the other through inflection or affixation) and *semantic markedness* where the opposition occurs at the semantic level as in the example above. When two antonymous terms are also morphologically related, the formally unmarked term is usually also the semantically unmarked one (for example, *clear–unclear*). However, this correlation is not universal; consider the examples *unbiased–biased* and *independent–dependent*. In any case, semantic markedness is the more interesting of the two and the harder to determine, both for humans and computers.

The semantically unmarked term of the pair can describe the entire scale in certain contexts, while this is not possible for the marked term. Therefore, the unmarked term is the more general and prominent of the two, and accordingly it should also be the positive one. This correlation has been noted frequently in the linguistics literature [Boucher and Osgood 1969; Lehrer 1985; Battistella 1990].

8.2 Text-level indicators of semantic markedness

As with antonymy and synonymy, markedness in general and semantic markedness in particular have received considerable attention in the linguistics literature. Consequently, several tests for determining markedness have been proposed by linguists. Most of these tests involve human judgments [Greenberg 1966; Lyons 1977; Waugh 1982; Lehrer 1985; Ross 1987; Lakoff 1987] and are not suitable for implementation on a computer. However, some proposed tests refer to comparisons between objectively measurable properties of the words in question and are thus amenable to automation. Although potentially automatable versions of some of these tests have been successfully applied to the problem at the phonology level [Trubetzkoy 1939; Greenberg 1966], little work has been done on the empirical validation or the automatic application of those tests at higher levels (but see [Kučera 1982] for an empirical analysis of a proposed markedness test at the syntactic level; some more narrowly focused empirical work has also been done on markedness in second language acquisition). In this section, I examine several of the tests of this type that have been discussed in the linguistic literature; in subsequent sections I empirically determine the validity of each of these tests and then employ the useful ones to distinguish between semantically marked and unmarked terms. The tests I consider are:

1. *Text frequency.* Since the unmarked term can appear in more contexts than the marked one, and it has both general and specific senses, it should appear more frequently in the text than the marked term [Greenberg 1966].

2. *Formal markedness.* A formal markedness relationship (i.e., a morphology relationship between the two words) should be an excellent predictor for semantic markedness whenever it exists [Greenberg 1966; Zwicky 1978].

3. *Formal complexity.* Since the unmarked word is the more general one, it should also be morphologically the simpler [Jakobson 1962; Battistella 1990]. The "economy of language" principle [Zipf 1949] supports this claim. Note that this test subsumes test 2.

4. *Morphological productivity.* Unmarked words, being more general and frequently used to describe the whole scale, should be freer to combine with other linguistic elements [Battistella 1990; Winters 1990].

5. *Differentiation.* Unmarked terms should exhibit higher differentiation with more subdistinctions [Jakobson 1984b] (for example, the present tense (unmarked) appears in a greater variety of forms than the past[2]), or, equivalently, the marked term should lack some subcategories [Greenberg 1966].

[2]While I analyze and exploit semantic markedness for adjectives, the property is not confined to this syntactic category. Much of the relevant literature examines semantic markedness for other categories, such as verbs.

The first of the above tests compares the text frequencies of the two words, which are clearly measurable and easily retrievable from the corpus. The mapping of the remaining tests to quantifiable variables is not as immediate. I use the length of a word in characters, which is a reasonable indirect index of morphological complexity, for tests 2 and 3. This indicator is exact for the case of test 2, since the formally marked word is derived from the unmarked one through the *addition* of an affix (which for adjectives is always a prefix). The number of syllables in a word is another reasonable indicator of morphological complexity that I consider, although it is much harder to compute automatically than word length.

For morphological productivity (test 4), I measure several variables related to the freedom of the word to receive affixes and to participate in compounds. Several distinctions exist for the definition of a variable that measures the number of words that are morphologically derived from a given word. These distinctions involve:

- Whether to consider the number of distinct words in this category (types) or the total frequency of these words (tokens).
- Whether to separate words derived through affixation from compounds or combine these types of morphological relationship. If a separation is made, a new category of words that involve both affixation and compounding should be considered.
- If words derived through affixation are considered separately, we may further subdivide them into words that are produced with the application to the base form of a single prefix, a single suffix, one prefix and one suffix, multiple prefixes, multiple suffixes, or a combination of multiple prefixes and suffixes. Subtotals corresponding to various

combinations of these categories can be also computed.

- If word types (rather than overall word frequencies) are measured, we can select to count homographs (words identical in form but with different parts of speech, e.g., *light* as an adjective and *light* as a verb) as distinct types or map all homographs of the same word form to the same word type.

These distinctions lead to the consideration of 28 variables expressing various combinations of the answers to the above choices. Since these variables are closely related and their number is so high that employing all of them in a prediction model for semantic markedness presents a clear danger of overfitting, I combined several of them. The result is 10 variables that represent different ways of measuring morphological productivity.

Finally, I mapped the differentiation test (5) to the number of grammatical categories (parts of speech) that each word can appear under, postulating that the unmarked term should have a higher such number.

8.3 Experimental setup for the empirical validation of semantic markedness indicators

In order to measure the performance of the markedness tests discussed in the previous section, I collected a fairly large sample of pairs of antonymous gradable adjectives that can appear in how-questions. The Deese antonyms [Deese 1964] is the prototypical collection of pairs of antonymous adjectives and they have been used for similar analyses in the past [Deese 1964; Justeson and Katz 1991; Grefenstette 1992a]. However, this collection contains only 75 adjectives in 40 pairs, some of which cannot be used in this study because they are either primarily adverbials (e.g., *inside–outside*) or not gradable (e.g., *alive–dead*). Unlike previous studies, the nature of the statistical analysis performed for the indicators of semantic markedness requires a higher number of pairs.

Consequently, I augmented the Deese set with the set of pairs used in the largest manual previous study of markedness in adjective pairs [Lehrer 1985]. In addition, I included all gradable adjectives which appear 50 times or more in the Brown corpus [Kučera and Francis 1967; Francis and Kučera 1982] and have at least one gradable antonym; the antonyms were not restricted to belong to this set of frequent adjectives. For each adjective collected according to this last criterion, I included all its antonyms (frequent or not) that were explicitly listed in the Collins COBUILD dictionary [Sinclair 1987b] for each of its senses. This process produced a sample of 449 adjectives (both frequent and infrequent ones) in 344 pairs.[3]

I separated the pairs on the basis of the how-test into those that contain one semantically unmarked and one marked term (e.g., *tall–short*) and those that contain two marked terms (e.g., *fat–thin*), removing the latter. For the remaining pairs, I identified the unmarked member, using existing designations [Lehrer 1985] whenever that was possible; when in doubt, the pair was dropped from further consideration. Eight of the 344 adjective pairs

[3]The collection method is similar to Deese's: He also started from frequent adjectives that appear 50 or more times per million words in the Thorndike-Lorge corpus [Thorndike and Lorge 1944] but used human subjects instead of a dictionary to elicit antonyms.

	One unmarked	Both marked	Total
Morphologically unrelated	211	54	265
Morphologically related	68	3	71
Total	279	57	336

Table 8.1: Cross-classification of adjective pairs according to morphological relationship and markedness status.

were removed at this stage because the markedness status of their members wasn't perfectly clear. I also separated the pairs into two groups according to whether the two adjectives in each pair were morphologically related or not. This allowed me to study the different behavior of the tests for the two strata separately. Table 8.1 shows the results of this cross-classification of the adjective pairs.

The next step was to measure the variables described in Section 8.2 which are used in the various tests for semantic markedness. For these measurements, I used the MRC Psycholinguistic Database [Coltheart 1981; Wilson 1987] which contains a variety of measures for 150,837 entries counting different parts of speech or inflected forms as different words (115,331 distinct words). I implemented an extractor program to collect the relevant measurements from the database for the adjectives in the selected sample, namely text frequency, number of syllables, word length, and number of different parts of speech. Frequencies of the adjectives in the one-million word Brown corpus of written American English [Kučera and Francis 1967; Francis and Kučera 1982] were used to supply the necessary text frequency estimates. All this information except the number of syllables can be automatically extracted directly from the corpus.[4] The extractor program also computes

[4]As values for the number of syllables variable are present in the MRC database, I had no need to explicitly compute them. If necessary, this variable can be approximated by the application of phonological

information that is not directly stored in the MRC database. Affixation rules are recursively employed to check whether each word in the database can be derived from each adjective, and counts and frequencies of such derived words and compounds are collected; Section A.6 in the Appendix discusses this morphological analysis component in detail. Overall, 32 measurements are computed for each adjective, and are subsequently combined into the 14 variables used in this study. Additional details on how the values of the variables for these adjectives are computed from the corpus can be found in the Appendix, and in particular in Section A.6 therein.

Finally, the variables for the pairs are computed as the differences between the corresponding variables for the adjectives in each pair. The output of this stage is a table, with two strata corresponding to the two groups of morphologically related and unrelated adjectives, and containing measurements on 14 variables for the 279 pairs with a semantically unmarked member.

8.4 Evaluation of independent linguistic tests

For each of the 14 variables identified in Section 8.3, I measured how many pairs in each group it classified correctly. A positive (negative) value for a variable indicates that the first (second) adjective is the unmarked one, except for two variables (word length and number of syllables) where the opposite is true. When the difference is zero, the variable

rules (for example, via a finite-state transducer) on the character string that represents each word.

selects neither the first nor the second adjective as unmarked. The percentage of non-zero differences, which correspond to cases where the test actually suggests a choice, is reported as the *applicability* of the variable. For the purpose of evaluating the *accuracy* of the variable, I assign the cases where the variable equally supports both choices of the first or second adjective as the unmarked one randomly to one of the two possible outcomes in accordance with the common practice in classification [Duda and Hart 1973].

For each variable and each of the two groups, I also performed a statistical test of the null hypothesis that its true accuracy is 50%, i.e., equal to the expected accuracy of a random binary classifier. Under the null hypothesis, the number of correct responses follows a binomial distribution with parameter $p = 0.5$. Since all obtained measurements of accuracy were higher than 50%, any rejection of the null hypothesis implies that the corresponding test is significantly better than chance. For the purposes of this test, I compute the exact probability of getting the observed or a higher number of correct responses under the null hypothesis, without using a normal approximation.

Table 8.2 summarizes the scores obtained for some of the 14 variables in the collected data and reveals some surprising facts about their performance. The frequency of the adjectives is the best predictor in both groups, achieving an overall accuracy of 80.64% with high applicability (98.5–99%). This is all the more remarkable in the case of the morphologically related adjectives, where frequency outperforms the length of the words; recall that the latter directly encodes the formal markedness relationship, so frequency is able to correctly classify some of the cases where formal and semantic markedness values disagree in addition to the cases where formal markedness suggests the correct answer. On

Morphologically Unrelated Adjectives			
Test	Applicability	Accuracy	P-Value
Frequency	99.05%	75.36%	$8.4 \cdot 10^{-14}$
Number of syllables	58.29%	55.92%	0.098
Word length	83.41%	52.13%	0.582
Number of homographs	71.09%	56.87%	0.054
Total number of compounds	64.45%	61.14%	$1.5 \cdot 10^{-3}$
Unique words derived by affixation	95.26%	66.35%	$2.3 \cdot 10^{-6}$
Total frequency of derived words	82.46%	66.35%	$2.3 \cdot 10^{-6}$

Morphologically Related Adjectives			
Test	Applicability	Accuracy	P-Value
Frequency	98.53%	97.06%	$1.6 \cdot 10^{-17}$
Number of syllables	95.59%	92.65%	$7.7 \cdot 10^{-14}$
Word length	100.00%	95.59%	$4.4 \cdot 10^{-16}$
Number of homographs	66.18%	79.41%	$1.1 \cdot 10^{-6}$
Total number of compounds	14.71%	60.29%	0.114
Unique words derived by affixation	98.53%	94.12%	$5.8 \cdot 10^{-15}$
Total frequency of derived words	83.82%	91.18%	$8.2 \cdot 10^{-13}$

All Adjectives			
Test	Applicability	Accuracy	P-Value
Frequency	98.92%	80.65%	$5.6 \cdot 10^{-26}$
Number of syllables	67.38%	64.87%	$7.6 \cdot 10^{-7}$
Word length	87.46%	62.72%	$2.5 \cdot 10^{-5}$
Number of homographs	69.89%	62.37%	$4.3 \cdot 10^{-5}$
Total number of compounds	52.33%	60.93%	$3 \cdot 10^{-4}$
Unique words derived by affixation	96.06%	73.12%	$5.8 \cdot 10^{-15}$
Total frequency of derived words	82.80%	72.40%	$4.2 \cdot 10^{-14}$

Table 8.2: Evaluation of simple markedness tests. The probability of obtaining by chance performance equal to or better than the observed one is listed in the *P-Value* column for each test.

the other hand, tests based on the "economy of language" principle, such as word length and number of syllables, perform badly when formal markedness relationships do not exist, with lower applicability and very low accuracy scores. The same can be said about the test based on the differentiation properties of the words (number of different parts of speech).

In fact, for these three variables, the hypothesis of random performance cannot be rejected even at the 5% level. Tests based on the morphological productivity of the words, as measured through affixation and compounding, tend to fall in between these extremes: their accuracy is generally significant, but their applicability is sometimes low, particularly for compounds.[5]

8.5 Differences in performance across the two strata of pairs

It is interesting to establish whether the performance of each linguistic test is the same in the two groups of pairs of morphologically unrelated and morphologically related adjectives. This knowledge is valuable in terms of its linguistic significance, but is also useful for the design of a complex predictor based on the various linguistic tests. If no performance difference exists across the two groups for all the individual predictors that will be combined in the complex predictor, then a single complex predictor can be constructed for both groups. If, on the other hand, significant differences exist between the two groups, then different classifiers should be constructed for each of them; these classifiers may assign different weights to the individual predictors, or even employ different sets of predictors.

To analyze the discrepancy in the accuracy of each variable between the two groups,

[5]This may be partly due to the decision to count only hyphenated compounds (cf. Section A.6 in the Appendix). This decision was made so as to avoid many spurious “compounds”, but it also results in a lower number of true compounds being collected.

I performed a statistical test of the null hypothesis that the proportions of the correct answers in these two independent samples are identical versus the alternative that they are different. Under the null hypothesis, the counts in the resulting fourfold table follow the hypergeometric distribution [Fleiss 1981; Dunning 1993], and the probability of observing a particular table under the null distribution can be computed with Fisher's exact test [Fisher 1934]. However, with 279 pairs in the two samples, the exact test becomes computationally infeasible, and I resort to an approximate test, namely Pearson's chi-square test [Fleiss 1981]. Under the null hypothesis, the χ^2 statistic (to which I apply Yates' correction for continuity [Yates 1934]) follows asymptotically a chi-square distribution with one degree of freedom. When the minimum expected frequency among the cells of the 2×2 table is reasonably large, the approximation is very good; this happens in this case, where the minimum expected frequency is always greater than 13.

Table 8.3 lists a subset of the results obtained when this statistical test is applied to the available variables. I observe that all variables except the ones based on compounds exhibit significantly different behavior between the two groups. Consequently, I construct separate complex classifiers for each stratum when I explore methods to combine the predictions of the individual linguistic tests in Section 8.7.

Test	Minimum expected count	χ^2 statistic	P-Value
Frequency	13.16	14.16	$1.7 \cdot 10^{-4}$
Number of syllables	23.89	28.84	$7.8 \cdot 10^{-8}$
Word length	25.35	39.70	$3 \cdot 10^{-10}$
Number of homographs	25.59	10.19	0.0014
Total number of compounds	26.57	0.015	0.9013
Unique words derived by affixation	18.28	18.78	$1.5 \cdot 10^{-5}$
Total frequency of derived words	18.77	14.64	$1.3 \cdot 10^{-4}$

Table 8.3: Statistical analysis of the discrepancy in the accuracy of simple tests for semantic markedness between the two groups of pairs of morphologically unrelated and morphologically related adjectives. High values in the "minimum expected count" column indicate more strongly that the chi-square test is applicable; as a rule of thumb, a minimum value of 5 is generally required. The "P-Value" column gives the probability that the same or a larger value of the χ^2 statistic, corresponding to more disparate differences in the observed classification accuracy between the two cases, would have occurred under the null distribution, i.e., if there were no difference in the performance of the analyzed markedness test across the two strata of pairs.

8.6 Correlation between the indicators of semantic markedness

Measurements of the correlation between the various indicators of semantic markedness can be used to estimate the benefit that is to be gained by combining these indicators in a complex, hopefully more accurate, predictor. If the answers based on the linguistic tests are statistically independent and informative, we can expect that combining them can improve the accuracy of the classifier; for truly independent indicators, the overall error rate decreases exponentially with the number of indicators employed in the complex predictor [Duda and Hart 1973]. If, on the other hand, we combine variables which do not satisfy the above two criteria, overtraining may cause a more complex classifier to perform worse than one based on a single variable.

I measured the correlation between the answers of any two tests using a linear transformation of the *simple matching coefficient* [Zubin 1938; Dumas 1955; Romesburg 1984]. Using these correlation estimates, I performed for each pair of variables a test of the null hypothesis that the predictions of the two variables are independent. By virtue of the randomized design of the markedness analysis experiment (Section 8.3), each variable has a prior probability of predicting each of the two outcomes equal to 0.5. Therefore, under the null hypothesis the probability of a match at any position is $0.5 \cdot 0.5 + 0.5 \cdot 0.5 = 0.5$. It follows that the number of matches between any two variables follows the binomial distribution with parameter $p = 0.5$.

The results of this analysis show that the hypothesis of independence must be rejected at any reasonable level for any pair of informative variables (such as word frequency and number or frequency of derived words). This reduces the expectation that a combination of multiple variables will improve the performance of a classifier. Nevertheless, although the single predictors are not statistically independent, they are not pairwise completely correlated either; thus, it remains worthwhile to examine methods that utilize more than one indicator simultaneously.

8.7 Predictions based on more than one test

While the frequency of the adjectives is the best single predictor, we would expect to gain accuracy by combining the answers of several simple tests.[6] I consider the problem of determining semantic markedness as a classification problem with two possible outcomes ("the first adjective is unmarked" and "the second adjective is unmarked"). To design an appropriate classifier, I employed two general statistical supervised learning methods: decision trees and generalized linear models.

Decision trees [Quinlan 1986] is the first statistical supervised learning paradigm that I explored. This approach is suitable when the response variable represents one of a set of prespecified alternatives, as in this case of binary classification. A decision tree contains at each internal node a binary choice of the form

if predictor variable $V_i < T$ then

  take left branch

else

  take right branch

[6]The correlation measurements between the various variables (Section 8.6) show little hope for improvement in the predictive power when more than one test are combined. However, I felt further work was necessary to either verify or disprove this pessimistic prediction.

where the variable V_i and the threshold T are generally different at different nodes. Each leaf of the tree is labeled with one of the alternatives for the response variable.[7] The tree is constructed iteratively, using a measure of the success of the current tree in correctly classifying the training samples to select which variable to partition and at which cutoff point. Several heuristics can be used to terminate the algorithm, including how many samples are currently classified in a given subtree and how homogeneous these samples are.

A popular method for the automatic construction of such trees from training data is *binary recursive partitioning*, which constructs a binary tree in a top-down fashion. Starting from the root, the variable X which better discriminates among the possible outcomes is selected and a test of the form $X < constant$ is associated with the root node of the tree. All training cases for which this test succeeds (fails) belong to the left (right) subtree of the decision tree. The method proceeds recursively, by selecting a new variable (possibly the same as in the parent node) and a new cutting point for each subtree, until all the cases in one subtree belong to the same category or the data becomes too sparse. When a node cannot be split further, it is labeled with the locally most probable category. During prediction, a path is traced from the root of the tree to a leaf, and the category of the leaf is the category reported.

If the tree is left to grow uncontrolled, it will exactly represent the training set (including its peculiarities and random variations), and will not be very useful for prediction on new cases. Consequently, the growing phase is terminated before the training samples assigned

[7]Multi-way classification is thus more natural with a tree-based model than with a (generalized) linear model. In the latter case, we need to break down the classification process into a series of related binary decisions and synthesize the results.

to the leaf nodes are entirely homogeneous. A technique that improves the quality of the induced tree is to grow a larger than optimal tree and then shrink it by pruning subtrees [Breiman *et al.* 1984]. In order to select the nodes to prune, we normally need to use new data that has not been used for the construction of the original tree.

In this classifier, I use about half of the available pairs in each of the two groups (morphorogically related or morphologically unrelated adjectives) as training data to construct a decision tree with the binary recursive partitioning method; each pair is randomly included in the training set with probability 50%. A maximum likelihood estimator based on the binomial distribution selects the optimal split at each node, and node splitting stops when the node is nearly homogeneous or has too few observations to ensure accurate estimation of the probabilities of each category. During the tree shrinking phase I optimally regress the probabilities of children nodes to their parent according to a *shrinking parameter* α [Hastie and Pregibon 1990], instead of pruning entire subtrees. A value of 1 for α results in the original tree, while a value of 0 represents maximum shrinking and replaces the tree with a single node. To select the optimal value for α, I initially held out a part of the training data. In a later version of the classifier, I employed *cross-validation*, separating the training data in ten equally sized subsets and repeatedly training on nine of them and validating on the other.[8] Finally I select the value of α and the corresponding tree that give the best fit during the validation stage.

[8] In this way, I avoid reducing the size of the training and testing sets by holding out some data for validation purposes.

Log-linear regression [Santner and Duffy 1989] is the second general supervised learning method that I explored. As I noted in Section 6.3, the log-linear regression model generalizes classical linear modeling to binomial sampling where the response variable follows a Bernoulli distribution (corresponding to a two-category outcome) and the variance is not independent of the mean of the response. The resulting *generalized linear model* [McCullagh and Nelder 1989] employs a linear predictor

$$\eta = \mathbf{b}^{\mathrm{T}}\mathbf{x}$$

where $\mathbf{b}$ is a vector of weights and $\mathbf{x}$ is the vector of the values of the predictor variables, but the response variable y is non-linearly related to η through the *inverse logit* function,

$$y = \frac{e^{\eta}}{1 + e^{\eta}}$$

Note that $y \in (0, 1)$; each of the two ends of that interval is associated with one of the possible choices (for example, in the particular problem of determining semantic markedness, whether the first or the second adjective in the pair is the unmarked one).

I employ the *iterative reweighted non-linear least squares* algorithm [Baker and Nelder 1978; Bates and Watts 1988] to approximate the maximum likelihood estimate of the vector $\mathbf{b}$, but first I explicitly drop the constant term (intercept) and most of the variables. The intercept is dropped because the prior probabilities of the two outcomes are known to be equal.[9] Several of the variables are dropped to avoid overfitting [Duda and Hart 1973];

[9] The order of the adjectives in the pairs is randomized before training the model, to ensure that both outcomes are equiprobable.

otherwise the regression model will use all available variables, unless some of them are linearly dependent. To identify which variables I should keep in the model, I use the *analysis of deviance* method with iterative stepwise refinement of the model by iteratively adding or dropping one term if the reduction (increase) in the deviance compares favorably with the resulting loss (gain) in residual degrees of freedom. Analysis of deviance methods were originally developed for basic linear models (where the computations are exact), and later extended to non-linear fitting functions, as in this case. Using a fixed training set, six of the fourteen variables were selected for modeling the morphologically unrelated adjectives. Frequency was selected as the only component of the model for the morphologically related ones.

Both classical and generalized linear models support transformations of the predictors, if we have additional knowledge about their effect on the response. In a transformation, a term $\beta_i V_i$ in the weighted sum of the linear predictor (equation (5.1) on page 221) is replaced by the term $\beta_i f_i(V_i)$, where f_i is any one-variable function. More generally, we may replace $\beta_i V_i$ with a the term $\beta_i f_i(V_{i1}, V_{i2}, \ldots, V_{ik})$ involving a function of more than one of the predictor variables, or with a series of terms $\beta_{i1} f_{i1}(V_{i1}, V_{i2}, \ldots, V_{ik}) + \beta_{i2} f_{i2}(V_{i1}, V_{i2}, \ldots, V_{ik}) + \cdots + \beta_{il} f_{il}(V_{i1}, V_{i2}, \ldots, V_{ik})$. Although the form of these functions cannot be predicted according to a general method, sometimes prior knowledge makes the choice of an appropriate function possible. For example, if we want to convert a predictor X to a binary variable via the application of a threshold T (see for example the discussion of the method that combines positive and negative evidence about the semantic similarity of two words in Section 2.6), we can use the function

$$f(X) = \begin{cases} 1 & \text{if } X \geq T \\ 0 & \text{if } X < T \end{cases}$$

A special case of these transformations on the predictors is to replace the observed values with estimates produced from nonlinear smoothing functions, such as cubic B-splines. This method has been shown to often improve the accuracy of the overall modeling process [Wahba 1990]. I investigated an alternative log-linear model where one or more of the variables that correspond to the simple linguistic tests are smoothed with cubic B-splines. The analysis of deviance for this revised model indicated that for the morphologically unrelated adjectives, one of the six selected variables should be removed altogether and another should be replaced by a smoothing spline.

8.8 Evaluation of the complex predictors

For both decision trees and log-linear regression, I repeatedly partitioned the data in each of the two strata into (almost) equally sized training and testing sets[10], constructed the predictors using the training sets, and evaluated them on the testing sets. This process was repeated 200 times, giving vectors of estimates for the performance of the various methods. The simple frequency test was also evaluated in each testing set for comparison purposes. From these vectors, I estimate the density of the distribution of the accuracy scores for each

[10] By including each pair of adjectives in the training set with probability 50%.

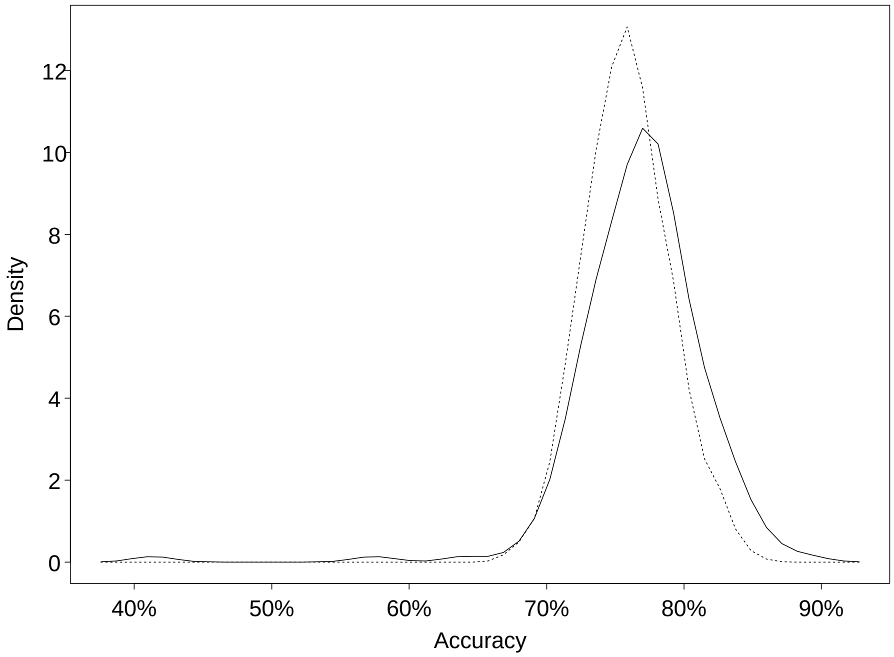


Figure 8.1: Probability densities of the accuracy of the frequency method (dotted line) and the smoothed log-linear model (solid line) for the prediction of semantic markedness on the collection of morphologically unrelated adjectives.

method; Figure 8.1 gives these densities for the frequency test and the log-linear model with smoothing splines on the most difficult case, the morphologically unrelated adjectives.

Table 8.4 summarizes the performance of the methods on the two groups of adjective pairs and should be contrasted with Table 8.2 on page 312.[11] In order to assess the significance of the differences between the scores, I performed a non-parametric sign test [Gibbons and Chakraborti 1992] for each complex predictor against the simple frequency variable. The test statistic is the number of runs where the score of one predictor is higher than the other's; as is common in statistical practice, ties are broken by assigning half of them to each category. Under the null hypothesis of equal performance of the two methods that

[11]The applicability of all complex methods was 100% in both groups.

Predictor tested	Morphologically unrelated		Morphologically related		Overall	
	Accuracy	P-Value	Accuracy	P-Value	Accuracy	P-Value
Frequency	75.87%	–	97.15%	–	81.07%	–
Decision tree (without cross-validation)	64.99%	$8.2 \cdot 10^{-53}$	94.40%	$1.5 \cdot 10^{-10}$	72.05%	$1.7 \cdot 10^{-54}$
Decision tree (cross validated)	69.13%	10^{-40}	94.40%	$1.5 \cdot 10^{-10}$	75.19%	$7.2 \cdot 10^{-47}$
Log-linear model (no smoothing)	76.52%	0.0281	97.17%	1.00	81.55%	0.0228
Log-linear model (with smoothing)	76.82%	0.0008	97.17%	1.00	81.77%	0.0008

Table 8.4: Evaluation of the complex predictors for semantic markedness. The probability of obtaining by chance a difference in performance relative to the simple frequency test equal to or larger than the observed one is listed in the *P-Value* column for each complex predictor and set of adjective pairs that are analyzed.

are contrasted, this test statistic follows the binomial distribution with $p = 0.5$. Table 8.4 includes the exact probabilities for obtaining the observed (or more extreme) values of the test statistic. Since the average scores observed for the decision tree methods were lower than the corresponding score for the simple frequency test, a rejection of the null hypothesis in these tests means that the tree-based methods perform indeed worse than frequency. As the opposite relationship holds between the average scores observed for the log-linear models and frequency, a rejection of the null hypothesis in these cases is interpreted as evidence that the log-linear models outperform the frequency test.

From the table, I observe that the tree-based methods perform considerably worse than frequency (significant at any conceivable level), even when cross-validation is employed.[12] Both the standard and smoothed log-linear models outperform the frequency test on the

[12] Although cross-validation improves, as expected, the performance of the decision tree.

morphologically unrelated adjectives (significant at the 5% and 0.1% levels respectively), while the log-linear model's performance is comparable to the frequency test's on the morphologically related adjectives. The best predictor overall is the smoothed log-linear model.[13]

The above results indicate that the frequency test essentially contains almost all the information that can be extracted collectively from all linguistic tests. Consequently, even very sophisticated methods for combining the tests can offer only small improvement. Furthermore, the prominence of one variable can easily lead to overfitting the training data in the remaining variables. This causes the decision tree models to perform badly.

8.9 Conclusions from the markedness analysis

In the previous six sections I presented a quantitative analysis of the performance of measurable linguistic tests for the selection of the semantically unmarked term out of a pair of gradable antonymous adjectives. The analysis shows that a simple test, word frequency, outperforms more complicated tests, and also dominates them in terms of information content. Some of the tests that have been proposed in the linguistics literature, notably tests that are based on the formal complexity and differentiation properties of the words, fail to give any useful information at all, at least with the approximations I used for them (Sec-

[13]It should be noted here that the independence assumption of the sign test is mildly violated in these repeated runs, since the scores depend on collections of independent samples from a *finite* population. This mild dependence will increase somewhat the probabilities under the true null distribution, but probabilities such as 0.08% will remain significant.

tion 8.2). On the other hand, tests based on morphological productivity are valid, although not as accurate as frequency.

The analysis presented here offers hard, quantitative evidence on the performance of markedness tests already proposed in the linguistics literature. Such tests are based on intuitive observations and/or particular theories of semantics, but their accuracy had not been measured before on actual data. The results of this analysis can be used to substantiate theories which are compatible with the empirical evidence, and thus offer insight into the complex linguistic phenomenon of antonymy.

As the validity of the results depends on the quality of the measurements, I should note here that many of the variables used in this study can be computed relatively easily with acceptable accuracy. Of course, for most of the variables the measurements will be approximate; the discrepancy between the measured and true values can be attributed to the following factors:

- The quality of measurements that involve frequencies ultimately depends on the quality of the sampling process. This, in turn, relates to how representative the corpus is, and to whether the extracted data is sparse enough to skew the estimates significantly.
- Indirect measurements, such as the ones based on morphological productivity which compare characteristics of secondary words deemed related to two primary words, are only as accurate as the process that identifies the words that are considered related to the primary words.

- Quantitative data based on a particular interpretation of a linguistic test can be representative only to the extent that this interpretation is correct.

I believe that the design of the experiments and the data extraction methods address the above concerns to a satisfactory degree, so that the obtained measurements can be considered accurate. The Brown corpus from which I drew frequencies for this study is a representative one, and I selected both a large sample of adjective pairs and a large number of frequent adjectives to avoid sparse data problems. An informal evaluation of the morphological component used to identify derived words and compounds (Section A.6 in the Appendix) shows that such words are retrieved with high precision and recall. Finally, the mapping of the linguistic tests to comparisons of quantitative variables was straightforward with the exception of the differentiation test—for that case, I believe that I considered a plausible interpretation.

The analysis of the linguistic tests and their combinations has also led to a computational method for the determination of semantic markedness. The method is completely automatic and produces accurate results at 82% of the cases. I consider this performance reasonably good, especially since no previous automatic method for the task has been proposed. While I used a fixed set of 449 adjectives for the analysis, the number of adjectives in unrestricted text is much higher. In the MRC Psycholinguistic Database, 25,547 of the 150,769 entries with part-of-speech information (16.94%) are classified as adjectives, not including past participles; if we only consider regularly used grammatical categories for each word, the percentage of adjectives rises to 22.97% (see Table 1.1 on page 79). For comparison, nouns (the largest class) account for 51.31% and 57.47% of the words under the two criteria.

In addition, while adjectives tend to have prevalent markedness and orientation values in the language at large, frequently these values are negated in specific domains or contexts. For example, *healthy* is in most contexts the unmarked member of the opposition *healthy–sick*; but in a hospital setting, *sickness* rather than *health* is expected, so *sick* becomes the unmarked term. The methods I have employed are based on the form of the words and their overall statistical properties, and thus cannot predict specific occurrences of markedness reversals. But they can predict the prevalent markedness value for each adjective in a given domain, something which is impractical to do by hand separately for each domain.

Finally, many of the indicators (and certainly the methodology for combining and evaluating them) are not restricted to adjectives. Although not within the scope of the present dissertation, the quantitative analysis reported here can be expanded to other classes of antonymous words, particularly verbs which are notoriously difficult to analyze semantically [Levin 1993]. A similar methodology can be applied to identify unmarked (positive) versus marked (negative) terms in pairs such as *agree–dissent*.

8.10 Labeling the orientation groups as positive or negative

I have presented an automatic method for the determination of semantic markedness values in pairs of antonymous adjectives, which can identify the semantically unmarked member of the pair with 82% precision. Since semantic markedness is closely correlated with semantic orientation [Boucher and Osgood 1969; Lehrer 1985; Battistella 1990], this method can be incorporated into the fourth stage of the system to aid in determining the polarity of adjectives and formulated sub-scales. Once again, the approach I follow has to be indirect: We cannot directly assign markedness values to adjectives, but we can do so once oppositions between the adjectives in terms of semantic orientation have been established. But the algorithms of Chapter 7 achieve exactly this goal, separating a list of adjectives into classes so that adjectives belonging in different classes have opposite semantic orientation.

The analysis of Section 8.8 has established that frequency is a simple criterion that performs almost equally well as the most complex predictor of semantic markedness (81.07% versus 81.77% precision). Frequency can be computed easily, and can also be averaged over many words when pairs of groups of words rather than pairs of single words are analyzed. Thus, I use word frequency as the sole criterion for determining semantic orientation. I compute the average frequency of the words in each orientation group, expecting the group with higher average frequency to contain the positive terms. This aggregation operation increases the precision of the labeling algorithm dramatically since indicators for many pairs of words are combined, even when some of the words have been incorrectly assigned to their present group.

Table 8.5 shows the results of applying this procedure to the adjective sets and corresponding retrieved orientation groups of Section 7.7; these adjective sets were constructed so as to be representative of different densities of conjunctions among their members. In all these cases, the ratio of the two group frequencies was higher for the group that contained the most positive terms. Thus, the method correctly identified the positive subgroup in all experiments. This strong correlation between average frequency and semantic orientation

Test set (A_α)	α	Number of adjectives in test set ($\lvert A_\alpha \rvert$)	Ratio of average group frequencies
A_2	2	730	1.8699
A_3	3	516	1.9235
A_4	4	369	1.3486
A_5	5	236	1.4040

Table 8.5: Evaluation of the subsystem that assigns semantic orientation labels to groups of adjectives with the same (but unknown) orientation. The adjective sets for which evaluation results are reported are the ones defined in Section 7.7; Table 7.3 on page 294 lists the accuracy of the sister subsystem that constructs these groups of adjectives with common orientation for the same adjective sets. The parameter α represents the density of the graph induced by conjunction-based links among these adjectives. Values in the "ratio of average group frequencies" column above 1 indicate a correct assignment of orientation labels by the system for the corresponding adjective set.

leads to an interesting linguistic observation: Although we have slightly more numerous ways to express negative properties and judgments (the reference set of 1,336 adjectives contained 679 negative terms, or 50.82% of the total), we refrain from using these terms, whose total frequency in the text is significantly smaller than the total frequency of the fewer positive terms.

Chapter 9

Putting it all together: From free text to antonyms

In earlier chapters I presented methods for grouping together semantically related words (Chapter 2), identifying gradable adjectives (Chapter 6), and assigning semantic orientation labels to words (Chapters 7 and 8). These three components can be naturally combined to produce a list of antonymous terms from scratch. The grouping stage identifies semantic links between the words, and since antonymy is a primary semantic relationship, antonyms should be placed in the same cluster. Thus, the semantic groups produced at the end of the first stage should contain antonymy information; what remains is to separate the antonymous semantic links from the relationships of other kinds of semantic relatedness. For gradable adjectives, this is achieved with the methods of the third component, since antonyms almost always have opposite semantic orientations. The intervening gradability component selects those words which the third component can be used on.

Consequently, an approach to the automatic identification of antonyms consists of three steps:

1. Partition the starting set of words into semantic groups, so that antonyms fall into the same group.

2. Identify the gradable words in each group.

3. Assign semantic orientation labels to the gradable words, and identify as antonyms the pairs of words that (a) belong to the same cluster and (b) have received different orientation labels.

I have applied the above algorithm, with modifications and extensions detailed below, to a set of well-known antonymous adjectives, the Deese antonyms [Deese 1964]. I describe this set and its psycholinguistic justification in Section 9.1. I then examine the corpus data available for its member adjectives and present the result of the grouping stage on this set. I also explore alternatives to looking at a single clustering for determining semantic relatedness; in particular, I discuss a new method that combines the results of multiple clusterings, each with a different number of clusters, to estimate the strength of the association between words. This is followed by the results of the gradability and semantic orientation stages. As I show in Section 9.5, where overall results on the Deese set are presented, each of the additional stages improves, for the purpose of identifying antonyms, the semantic classes formed during the word grouping stage. I conclude with a closer look at the retrieved word associations, and discuss several reasons that explain why some of them are not in the Deese associations set.

above–below	cold–hot	happy–sad	new–old
alone–together	dark–light	hard–soft	old*–young
active–passive	deep–shallow	heavy–light*	poor–rich
alive–dead	dry–wet	high–low	pretty–ugly
back–front	easy–hard*	inside–outside	right–wrong
bad–good	empty–full	large–small	rough–smooth
big–little	far–near	left–right*	short–tall
black–white	fast–slow	long–short*	sour–sweet
bottom–top	few–many	married–single	strong–weak
clean–dirty	first–last	narrow–wide	thick–thin

Figure 9.1: The Deese antonyms. In pairs where one word is marked with *, the marked word elicits the unmarked word in the pair as its second most frequent response, because the position of highest frequency has been pre-empted by another word in the collection; but the marked word is the primary response for the unmarked word. Thus each marked word appears in two pairs. In pairs with no marked words, each adjective is the primary response for the other member of the pair. Adapted from [Deese 1964, p. 350].

9.1 The Deese antonyms

The set of Deese antonyms (Figure 9.1) consists of 40 pairs of high-frequency adjectives. In all of these pairs, the two member adjectives exhibit a strong antonymic relationship on the basis of word association tests which provide an objective psycholinguistic foundation for judging the semantic relatedness of words. In such association tests, human subjects are presented with one word (the *stimulus*) and are asked to supply the single word that they consider most related to the stimulus word; this becomes the *response* word. The response is not explicitly constrained by the grammatical properties of the stimulus. In particular, the subjects are free to select a response with a different part of speech; adjectives, for example, typically elicit either other adjectives or nouns (e.g., *high* → *low*, *administrative* → *decision*).

Deese [1964] constructed the set of antonyms by starting with a list of 278 common (i.e., high-frequency) English adjectives. These were all the adjectives with a count of at least 50 per million words in the Thorndike-Lorge corpus [Thorndike and Lorge 1944]. Not all of these words were used primarily as adjectives; some, such as *inside–outside*, appeared more frequently as adverbs. Nevertheless, the criterion applied by Deese in determining what is an adjective was whether a word *could* be used in an adjectival position in an ordinary English sentence.

The list of 278 adjectives was given to 100 human subjects, each of whom produced a single response for each adjective X_i, the word Y_i (unrestricted by part-of-speech) that they thought as most related to X_i. On the basis of this information, Deese defined the *primary response*, $P(X_i)$, to each stimulus as the word that was most frequently supplied among the 100 responses collected for this stimulus. Thus, he obtained pairs of the form ⟨stimulus, primary response⟩. He observed that 70 of the 278 adjectives had the remarkable property that whatever word they elicited as primary response, that word had the original word as its primary response, i.e., $P(P(X)) = X$. Because of the reciprocal character of the association between the two words in each of these 35 pairs, these pairs can be thought as representing the strongest adjectival associations.

In addition to these 70 words that form a set D_1, five adjectives almost satisfied the reciprocal association property. For each of these adjectives (*easy*, *heavy*, *left*, *long*, and *young*), their primary response belonged in the set D_1 of the 70 words with reciprocal primary responses and elicited the adjective as its second most frequent response because the highest-frequency position was pre-empted by the corresponding member of D_1. For example, *young* elicited *old* as primary response, while for *old* the most frequent response was *new* and the second most frequent response was *young*.

Deese defined the set of adjectival association pairs (and adjectives) that later became known as the Deese set as the set D_1 augmented with the five associations that were described in the preceding paragraph. The set contains 40 association pairs involving 75 adjectives; the five adjectives *hard*, *light*, *old*, *right*, and *short* appear with two associated words each. It is no surprise that these five words are all highly polysemous.

The most remarkable property of the associations in the Deese set is that the words in each pair stand in a clear antonymic relationship. Recall that antonymy was not a consideration in the construction of the set; it just turns out that *all* reciprocal or almost reciprocal word relationships between adjectives that were extracted from this objective association data are antonymous in nature. In this way, experimental evidence supports the thesis that antonymy is the strongest lexical semantic relationship, at least for high-frequency adjectives.

The Deese set was used by Deese [1964] in his analysis of factor loadings for associative contrasts, and in subsequent psycholinguistic work. It has also been used in computational linguistics, e.g., for an experimental validation of the co-occurrence hypothesis for antonymy [Justeson and Katz 1991] and as a reference set of adjectival antonyms [Grefenstette 1992a]. It is for this last purpose that I decided to employ the Deese set; I supply its member words to the algorithm for automatically identifying antonyms, and then evaluate how many of the pairs the automated system can successfully retrieve.

9.2 Retrieving semantic groups for the words in the Deese set

Before applying the word grouping algorithm to the adjectives in the Deese set, we need to examine whether we have sufficient data for each of them in the corpus. Normally, we would expect them to appear with high frequency in the 21 million word 1987 Wall Street Journal corpus that I use for these experiments (see Section A.2 in the Appendix), since they are common adjectives not specific to any particular domain and one of the criteria for their inclusion was high frequency in a smaller corpus. However, Deese selected words that were frequent and could appear in adjectival positions in a sentence, but not necessarily primarily as adjectives. Deese himself noted that some of these words (e.g., *inside*) are primarily used as adverbs rather than adjectives [Deese 1964, p. 348].

Table 9.1 lists the frequency of each of the words in the Deese set, both in adjectival and non-adjectival use. These figures were obtained from the 1987 Wall Street Journal corpus automatically tagged with part-of-speech information using the PARTS tagger [Church 1988] (see Section A.3.5 in the Appendix for more details on the tagger). I observe that while 77% of the 75 words appear in the corpus more than 500 times (*shallow* being the only word appearing less than 100 times, with a total frequency of 59), the frequency with which several of the Deese words appear as adjectives is lower than expected. This is due to several reasons:

- As noted earlier, several of the Deese words are primarily adverbs, nouns, prepositions,

Word	Frequency as adjective	Frequency as non-adjective	Word	Frequency as adjective	Frequency as non-adjective
above	155	2,639	large	6,172	129
active	1,658	109	last	24,765	8,157
alive	286	16	left	328	3,484
alone	52	1,335	light	428	1,839
back	310	8,561	little	4,476	2,135
bad	1,948	212	long	3,636	3,366
below	0	2,547	low	3,207	406
big	8,287	2,294	many	19,528	217
black	2,202	969	married	17	469
bottom	21	709	narrow	632	239
clean	269	270	near	482	1,936
cold	392	175	new	32,457	27,360
dark	280	137	old	5,165	292
dead	575	99	outside	1,147	2,047
deep	666	134	passive	61	221
dirty	172	17	poor	1,510	1,466
dry	247	141	pretty	152	913
easy	1,106	96	rich	633	185
empty	240	24	right	2,280	3,484
far	603	6,100	rough	201	72
fast	373	635	sad	177	2
few	7,589	0	shallow	22	37
first	0	24,641	short	3,011	372
front	341	975	single	1,434	188
full	3,083	364	slow	853	509
good	6,368	974	small	5,796	238
happy	688	25	smooth	139	58
hard	2,001	789	soft	517	60
heavy	97	2,377	sour	89	59
high	6,085	1,362	strong	5,632	109
hot	808	42	sweet	106	76
inside	242	994	tall	211	5

Table 9.1: Frequency of the 75 words in the Deese set, in adjective and non-adjective positions, in the 21 million word 1987 Wall Street Journal corpus. Continued on the next page.

Word	Frequency as adjective	Frequency as non-adjective	Word	Frequency as adjective	Frequency as non-adjective
thick	151	18	wet	104	15
thin	517	11	white	1,273	4,953
together	0	2,194	wide	855	58
top	3,148	1,597	wrong	1,149	116
ugly	107	9	young	1,980	910
weak	1,455	5			

Table 9.1: Frequency of the 75 words in the Deese set, in adjective and non-adjective positions, in the 21 million word 1987 Wall Street Journal corpus. Continued from the previous page.

or verbal forms, although they can be used in an adjectival position. So *above* appears 155 times as an adjective (e.g., *If you had read the above headline* ...) but 2,632 times as a preposition. Similarly, *bottom* appears 632 times as a noun but only 21 times as an adjective, and *married* appears 384 times as a past participle but only 17 times as an adjective.

- Sometimes the tagger assigns to a word a specialized tag, precluding its being recognized as an adjective. This applies to only one of the Deese words, *first*, which is marked as an ordinal number rather than an adjective in 20,586 occurrences.

- The tagger may misclassify many occurrences of the Deese words in the corpus. Although the tagger has an overall accurracy of over 90%, it is exactly the frequent words with many possible parts of speech such as the Deese adjectives that account for the bulk of the tagger's errors (for example, *right* can be an adjective, noun, proper noun, qualifier (as in *right now*), adverb, or verb). The tagger is also frequently misled by capitalization to label a word occurrence as a proper noun. Tokens that are misclassified as adjectives will probably not match the finite-state grammar rules that are subsequently applied (see Sections 2.1–2.2 and Sections A.4.2–A.4.4 in the Appendix), therefore this type of error does not usually affect the system. However, adjectival occurrences of a word that are mistakenly assigned another part-of-speech tag result in loss of data, as these tokens will never be presented to the grammar. For example, *heavy* is systematically misclassified as a noun by the tagger, which reports 2,225 occurrences of *heavy* as a noun in the corpus and only 97 occurrences as an adjective.

- Genre differences in the corpus from which Deese's word counts were derived and the Wall Street Journal corpus I use are reflected in lower-than-expected frequencies for several of the words in the Deese set. Recall that the Deese words are supposed to occur at least 50 times per million words; for 29 of the 75 words (*alive*, *bottom*, *clean*, *cold*, *dark*, *dead*, *deep*, *dirty*, *dry*, *empty*, *happy*, *hot*, *married*, *narrow*, *passive*, *rich*, *rough*, *sad*, *shallow*, *smooth*, *soft*, *sour*, *sweet*, *tall*, *thick*, *thin*, *ugly*, *wet*, and *wide*) this hypothesis must be rejected at the 0.001% level[1], even when using their total frequency rather than their frequency as adjectives. This brings into question whether these words are indeed common adjectives for the domain of financial and general-news articles, and suggests the possibility that some of them may have specialized meanings in this domain, meanings that do not necessarily contrast with those of their counterparts in the Deese set. I return to these two issues in Section 9.6 where I discuss factors that affect the performance of the antonymy identification subsystem.

The lack of data is most severe for three of the Deese words (*below*, *first*, and *together*) which do not appear in the corpus as adjectives at all. Since the subsequent word grouping

[1] Under a marginal binomial distribution assumption.

stage relies on syntactic information for identifying syntagmatic and eventually semantic relationships between the words, these three adjectives (if indeed they can be called adjectives) must be dropped from further consideration. Their three antonyms in the Deese set (*above*, *last*, and *alone*, respectively) do not participate in any other pairs and their retainment in the reference set of antonyms serves no purpose. Consequently, I drop these six words and the three pairs they participate in from the Deese set, and use the remaining 69 words and 37 pairs in the rest of this chapter. Other researchers had to adopt a similar policy of exclusion of some words and pairs due to insufficient corpus data; for example, Justeson and Katz [1991] eliminated ten words in five pairs (*above–below*, *alone–together*, *few–many*, *first–last*, and *married–single*) because at least one of their member words did not appear as adjectives in the Brown corpus that Justeson and Katz used as their source of word frequencies. One could argue that more words should have been removed from the reference set for the purposes of the present study, especially words that because of their adverbial usage, tagger errors, or domain characteristics appear as adjectives with relatively low frequency. Nevertheless, I opted to apply the antonymy identification algorithm to as large a subset of the Deese collection as was possible, and note the potential effects of low frequencies on the results for these borderline adjectives rather than exclude them from the analysis altogether.

The surviving 69 adjectives are passed on to the word grouping stage (cf. Chapter 2). Since we are looking for antonyms, singleton word groups are not productive. Therefore, I use a modified version of the clustering algorithm of Section 2.7.1 that guarantees that the output clusters will each contain at least two adjectives. This variant of the clustering algorithm uses the same objective function Φ and starts, as before, with the creation of a random partition of the adjectives, but now this partition will always have at least two adjectives in each cluster. The algorithm proceeds as before, locating the locally best move of an adjective to a different cluster, and iterates until no more movements can improve (i.e., decrease) the objective function. Moves that leave a cluster with less than two elements are, however, not allowed. Because this restriction can frequently lead to a premature termination of the optimization process as the number of legal moves is greatly reduced, I have added another means for transforming a partition from iteration to iteration: In addition to simple moves of a word from its current cluster to another, the algorithm also considers all possible *exchanges* of two words (belonging, of course, to different clusters). Such compound moves are always legal, since they do not change the number of adjectives in any cluster, and correspond to partial lookahead, allowing a word to leave a cluster even if the latter has only two members. At each iteration, the operation (move or exchange) that leads to the largest decrease of the objective function is chosen. Observation of execution traces indicates that both types of partition transformation are chosen, and none appears to be dominant.

Since the current partition is replaced by another only if the objective function strictly decreases, this more sophisticated version of the clustering algorithm also terminates in finite time.[2] The version that incorporates lookahead runs somewhat slower than the original version, as expected, but not prohibitively so.

[2]All other observations made in Chapter 2, concerning for example time complexity and the need for randomization, continue to hold.

1. full ugly
2. big heavy light
3. fast slow
4. rich sour
5. dark near new
6. hard sad
7. inside outside top
8. dry good poor
9. right wrong
10. happy married young
11. far strong weak
12. back front high low
13. alive dead old
14. pretty single
15. bottom narrow wide
16. clean little
17. active passive
18. easy sweet
19. empty large left small
20. cold hot
21. few many thick thin
22. dirty white
23. deep shallow
24. bad rough
25. long short smooth
26. black tall
27. soft wet

Figure 9.2: Partition with 27 clusters produced by the system for the 69 adjectives in the modified set of Deese antonyms.

Running this version of the clustering algorithm on the 69 adjectives in the modified Deese set produces partitions where semantic relatedness is indicated by membership in the same group; one such partition is produced for each different number of clusters, a parameter that is externally specified to the clustering component. The heuristics of Section 2.7.3 suggest 25 to 29 clusters as the appropriate level of granularity for partitioning 69 words; Figure 9.2 shows the partition with 27 clusters that was produced by the system.

The Deese pairs imply a grouping of the words, where two adjectives are placed in the same group if they appear together in a pair. I applied to the automatically produced groupings the evaluation procedure of Chapter 3, using the grouping that corresponds to the Deese pairs as the single reference model. This process assigns to the partition of Figure 9.2 a recall score of 38.10% and a precision score of 26.67%, without the corrections of Section 3.6. Partitions with different numbers of clusters achieve recall scores between 28% and 50%, and precision scores between 20% and 30%.

These scores refer to retrieved associations, without distinguishing between associations of words of same or different semantic orientation. Evidence from Chapter 7 shows that we can separate adjectives on the basis of orientation with a high degree of accuracy (78–92% or more, depending on the frequency of conjunctions between the adjectives). Assuming that this holds for the Deese adjectives, recall should not suffer much when the second and third stages of the antonymy identification algorithm are applied, while precision should be almost doubled. Thus, on the basis of this partition of the words, we expect to recover about a third to a half of the Deese antonyms; at the same time, the system will propose roughly as many additional antonyms that are not in the Deese set as those that are retrieved.

To interpret these performance characteristics, one should keep in mind that the word grouping system tries to identify pairs and groups of closely semantically related words. These words may not be antonyms or synonyms but nevertheless be related in the language through their paradigmatic substitutability[3]; such non-standard semantic relationships will not be in the Deese set, but are not necessarily erroneous. In other cases, the retrieved associations may be antonymic in nature but not included in the Deese set. For example, group 8 in Figure 9.2 contains the association *good–poor*; these two adjectives are clearly

[3] *Doctor–nurse* is the classical example of such associations.

antonyms in many contexts, yet they are not part of the Deese set (which contains *good–bad* and *rich–poor*). The domain from which the corpus is drawn obviously affects the associations that are retrieved, and it should also affect the model of "correct" associations. It is possible that some of the Deese antonyms are not valid in this domain, because some words take specialized senses, while retrieved associations not in the Deese set are actually correct ones. Unfortunately, we cannot know this without the help of lexicographers specializing in this domain, so the system will generally be penalized for recovering domain-specific lexical associations.

9.3 Looking at multiple groupings to determine lexical association

One way to improve the system's performance at recovering lexical associations is to look at multiple clusterings simultaneously. The actual placement of words within groups can vary significantly as the number of groups changes. As I noted in Section 4.1.2, this effect is most pronounced for words that are relatively semantically "empty" and are frequently used as emphasizers, intensifiers, or diminishers (e.g., *little*, *real*). Significant movement of words from cluster to cluster can also take place when the competing association evidence between a word and the words in two or more clusters is nearly balanced. Therefore, it is advantageous to combine the information present in multiple clusterings, each for a different number of groups within a wide range, rather than rely exclusively on a single representative grouping.

In any particular grouping, placement of two words in the same group indicates that these words are semantically related. But the strength of this evidence varies from grouping to grouping. Clearly, a partition with more clusters has less pairs of associated words, and thus the presence of an association in such a partition is more significant than in a partition with a smaller number of clusters. At the extreme, a grouping with only one cluster contains an association for every pair of words, and the relative value of this evidence is zero. In general, when we randomly place $n \geq 2$ words in c groups ($c \leq n$) with the condition that no group has less than m words, but otherwise with all the groups having the same probability of receiving each word[4], the probability that any two given words will be in the same group is

$$\begin{aligned} P &= \frac{cm}{n} \cdot \frac{cm-1}{n-1} \cdot \frac{m-1}{cm-1} + \left(1 - \frac{cm}{n} \cdot \frac{cm-1}{n-1}\right) \cdot \frac{1}{c} \\ &= \frac{cm(m-1)}{n(n-1)} + \left(1 - \frac{cm(cm-1)}{n(n-1)}\right) \cdot \frac{1}{c} \end{aligned} \tag{9.1}$$

I have extracted the associations from all the groupings produced when the clustering algorithm of the previous section is applied to the adjectives in the modified Deese set. Each such association is assigned a value equal to $-\log P$, where P is the probability of randomly obtaining the association given the current grouping's characteristics, as specified

[4]This algorithm is employed by the clustering subsystem to determine random initial partitions. Note that not all partition shapes are equiprobable, with the randomization algorithm favoring more compact initial partitions, i.e., partitions with about the same number of elements per cluster, since there are more potential partitions with compact shapes than with non-compact shapes. But the words are indeed randomly assigned to clusters, and all possible partitions have the same probability of occurrence.

by equation (9.1) ($n = 69$, $m = 2$, $c = 1 \ldots 34$). This is the *information* we gain by actually observing the association [Shannon 1948; Fano 1961]. The sum of these weights for each association across all the groupings it has been observed in provides an overall measure of the strength of the association. Thus, we obtain a ranking of the retrieved associations according to overall strength.

Having a ranked list of associations allows the selection of those for which the evidence in the data and the system's confidence is highest. Thus, we can trade off some coverage for increased precision. For example, the top five retrieved associations all belong to the set of Deese pairs. Table 9.2 lists the 50 strongest retrieved associations in order of decreasing overall association strength, along with the cumulative number of recovered Deese antonyms and the cumulative precision of the association retrieval process within the sets of the top five, top ten, etc. associations. Precision gracefully degrades as we increase the number of associations we consider, from 100% for the top five associations to 32% for the entire set of 50 associations (the modified Deese set contains 37 associations, so any collection of more than 37 associations is bound to include some that do not belong in the Deese set). Even when all the top 50 associations are examined, overall precision is higher than what is achieved when we rely on a single partition of the words; and these precision figures do not include the improvement that is expected when associations between words of the same semantic orientation are weeded out by the subsequent stages of the antonymy algorithm.

The ranked list of associations can also be used to compute, for each word, an ordered list of the words most related to it. This presents lexical association information in the same format as Deese's original data, although very different means are used to collect it

n	Associations	Overall association strength	Cumulative total of retrieved Deese antonyms	Cumulative precision
5	**high–low**	134.098	5	100.00%
	inside–outside	130.002		
	right–wrong	128.987		
	heavy–light	126.459		
	strong–weak	125.231		
10	**narrow–wide**	125.065	8	80.00%
	large–small	118.413		
	back–low	117.828		
	hard–sad	117.509		
	fast–slow	117.277		
15	back–high	116.495	9	60.00%
	inside–top	114.835		
	happy–young	111.828		
	cold–hot	111.822		
	outside–top	110.996		
20	**few–many**	107.380	12	60.00%
	dead–married	103.125		
	easy–sweet	99.449		
	long–short	96.778		
	active–passive	94.175		
25	soft–wet	93.489	12	48.00%
	dry–poor	82.820		
	good–right	80.974		
	good–wrong	79.973		
	married–young	78.761		
30	happy–married	77.720	13	43.33%
	cold–dry	76.560		
	clean–empty	74.512		
	light–many	73.769		
	deep–shallow	73.046		

Table 9.2: The 50 strongest retrieved associations for the words in the modified Deese set, listed in order of decreasing overall association strength. The association strengths were calculated from all the groupings (with 1 to 34 clusters) produced when the clustering algorithm was applied to the Deese words. The associations are presented in groups of five, along with the cumulative total of Deese pairs observed among the top n associations and the precision of the retrieval process when these top n associations are considered. Associations that belong in the Deese set of antonyms are shown in bold. Continued on the next page.

n	Associations	Overall association strength	Cumulative total of retrieved Deese antonyms	Cumulative precision
35	soft–strong heavy–many big–fast soft–weak black–tall	72.399 71.990 70.681 70.391 68.828	13	37.14%
40	good–left dry–hot long–smooth alive–old dead–young	67.566 66.255 65.785 64.372 63.635	13	32.50%
45	**back–front** **thick–thin** **black–white** near–new slow–smooth	63.629 63.609 63.289 62.996 62.884	16	35.56%
50	big–slow fast–smooth tall–white short–thick full–new	62.649 62.344 62.168 61.911 61.544	16	32.00%

Table 9.2: The 50 strongest retrieved associations for the words in the modified Deese set, listed in order of decreasing overall association strength. Continued from the previous page.

in each case (a large corpus with no semantic knowledge versus human responses in association tests). Table 9.3 shows the ten most related words, ordered in decreasing association strength, for each of the 69 words in the modified Deese set. Note that this collection of most related words for each adjective is not in general symmetrical, i.e., the fact that word A is the word most strongly related to word B does not necessarily imply that so is word B to word A. This is also true in Deese's original pairs (e.g., *right* is the primary response for *left*, but *wrong* is the primary response for *right*). Actually, it is interesting to note that for the five adjectives with multiple antonyms within the Deese set, the system generally

Adjective	Deese antonym(s)	Most related words identified by the system	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
active	passive	**passive** small large rich thin left far front sour big	1	–
alive	dead	old **dead** young little married happy rich big single clean	2	–
back	front	low high **front** sour strong heavy full soft weak deep	3	–
bad	good	rough wrong right **good** pretty sad sour left sweet easy	4	–
big	little	fast slow smooth rich young heavy pretty alive light thin	36.5	–
black	white	tall **white** ugly dark married dead happy bottom wet dry	2	–
bottom	top	dry hot cold narrow wide wet dark poor deep rough	65.5	–
clean	dirty	empty **dirty** new full old small little large easy good	2	–
cold	hot	**hot** dry bottom wet poor soft dark new black old	1	–
dark	light	black near shallow white new bottom ugly tall full rough	30.5	–
dead	alive	married young happy **alive** old black tall white dirty big	4	–
deep	shallow	**shallow** little narrow near sour wide passive dark bottom smooth	1	–
dirty	clean	**clean** white empty full new easy single small hard dead	1	–
dry	wet	poor cold hot bottom **wet** good dark soft black happy	5	–

Table 9.3: The ten most strongly associated words for each adjective in the modified Deese set. For each word, the Deese antonym and the ten most associated words that are retrieved by the system are listed, in order of decreasing overall association strength. Five of the Deese adjectives have two antonyms within the Deese set; these words are polysemous not just in the language at large but also within the Deese collection. For these five words, two antonyms are listed in the second column, with the primary antonym (the most frequent response to the word) listed first. If the Deese antonym(s) appear in the list of ten best recovered words, they are shown in bold. The last two columns give the ranks of the primary and secondary (if any) antonyms within the entire ordered list of recovered associated words. Ranks vary from 1 (best) to 68, and may take non-integer values in the presence of ties. The table is continued on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
easy	hard	sweet new sad **hard** dirty empty clean wrong full rough	4	–
empty	full	clean old large small new dirty **full** left easy near	7	–
far	near	weak strong few many rich shallow **near** small active large	7	–
fast	slow	**slow** big smooth sour bottom long strong small clean large	1	–
few	many	**many** light heavy far pretty thick thin sweet single slow	1	–
front	back	**back** low high single rich active long thick far small	1	–
full	empty	new rough clean **empty** ugly dirty near hard easy shallow	4	–
good	bad	right wrong left **bad** poor sour sweet clean dry sad	4	–
happy	sad	young married dead old alive outside inside top black tall	64	–
hard	soft easy	sad new full **easy** rough shallow dirty dark empty old	44.5	4
heavy	light	**light** many few big slow low far high back sweet	1	–
high	low	**low** back front sour soft weak strong thin light heavy	1	–
hot	cold	**cold** dry bottom wet poor large dirty weak dark good	1	–
inside	outside	**outside** top single happy old active tall alive left sour	1	–
large	small	**small** left empty rich active thin clean far near passive	1	–
left	right	good wrong **right** small large active empty sour rich bad	3	–
light	dark heavy	**heavy** many few slow big high low thick sweet bad	24.5	1
little	big	alive deep wide narrow passive near thick thin shallow clean	34.5	–

Table 9.3: The ten most strongly associated words for each adjective in the modified Deese set. Continued from the previous page, and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
long	short	**short** smooth thick ugly rough near front fast deep left	1	–
low	high	**high** back front strong soft sour thin heavy weak far	1	–
many	few	**few** light heavy far pretty thick thin slow dark sweet	1	–
married	single	dead young happy alive tall old white black rich **single**	10	–
narrow	wide	**wide** passive bottom little deep thin large small shallow easy	1	–
near	far	new shallow dark little deep full **far** short large rich	7	–
new	old	near full empty clean hard dark dirty easy **old** rough	9	–
old	new young	alive empty dead happy **young** clean **new** married single easy	7	5
outside	inside	**inside** top single happy passive bad high light old soft	1	–
passive	active	**active** narrow wide little thin small deep large low heavy	1	–
poor	rich	dry good hot wet cold rough tall bottom sour large	11	–
pretty	ugly	single bad few sweet many big tall rich new easy	47	–
rich	poor	sour small large active far big left young shallow alive	14	–
right	wrong left	**wrong** good **left** bad sad sweet sour ugly easy pretty	1	3
rough	smooth	bad full short long hard new poor **smooth** shallow near	8	–
sad	happy	hard wrong easy bad right sweet good new pretty narrow	67	–
shallow	deep	**deep** near dark far little rich rough small full large	1	–
short	tall long	**long** thick smooth rough ugly near front single deep little	64	1

Table 9.3: The ten most strongly associated words for each adjective in the modified Deese set. Continued from the previous page, and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
single	married	pretty front inside outside top dirty tall few empty clean	16	–
slow	fast	**fast** smooth big heavy sour light few many high bottom	1	–
small	large	**large** left rich empty active clean far thin shallow passive	1	–
smooth	rough	long slow fast big short **rough** bottom deep narrow near	6	–
soft	hard	wet strong weak high sour low far bottom cold dry	64	–
sour	sweet	rich deep left good strong bad wrong high weak active	63.5	–
strong	weak	**weak** soft far wet sour thin low high back fast	1	–
sweet	sour	easy good right pretty bad sad wrong few single heavy	53	–
tall	short	black white married top poor dark single pretty dead happy	27	–
thick	thin	**thin** short little long few many front full new active	1	–
thin	thick	**thick** large active little weak small passive strong wide narrow	1	–
top	bottom	inside outside tall single happy active left front large poor	59.5	–
ugly	pretty	black full long dark short white good right deep bad	38.5	–
weak	strong	**strong** soft far wet thin sour high low poor back	1	–
wet	dry	soft weak strong hot **dry** cold bottom poor dark black	5	–
white	black	**black** tall dirty dark married dead ugly bottom wet shallow	1	–
wide	narrow	**narrow** passive bottom little deep thin small large ugly shallow	1	–
wrong	right	**right** good left bad sad sour easy sweet rough ugly	1	–

Table 9.3: The ten most strongly associated words for each adjective in the modified Deese set. Continued from the previous page, and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
young	old	happy married dead alive **old** rich big small dirty left	5	–

Table 9.3: The ten most strongly associated words for each adjective in the modified Deese set. Continued from the previous page.

prefers the second most frequent response rather than the primary antonym.[5] In three of these cases (*hard*, *light*, and *short*) the system locates this secondary antonym among the top related words, but fails to identify the primary antonym as a strongly associated word (e.g., *light* is strongly associated with *heavy* but not so with *dark*). In the remaining two cases, the system is able to overcome polysemy and identifies strong correlations between *old* and both *new* and *young*, and between *right* and both *wrong* and *left*.

The Deese set contains 37 pairs of antonyms, with each word mapped to one or two antonyms. If we select just the top word from Table 9.3 as a candidate antonym for each adjective (or, for adjectives with two antonyms in the Deese set, the top two words), the recovery process also identifies 74 potential antonyms (identification of A as an antonym of B does not necessarily mean that B will be identified as an antonym of A, although this is usually the case). Out of these, 31 belong to the Deese set, for a recall of 41.89%. Since we produce the same number of candidates as the reference set (i.e., in this specific case, the modified Deese set) has members, it follows from the definitions of precision and recall that precision is bound to be equal to recall, i.e., 41.89%. But the ordered lists of related

[5]This might reflect a domain characteristic, or peculiarities in Deese's sampling process. In any case, both of the responses in such cases are antonyms, clearly identified as such by humans.

words in Table 9.3 allow the *overgeneration* of responses [Chinchor 1992], trading some precision for increased coverage of the Deese set. Frequently, the correct antonym will be ranked high in the list of related words, although not at the top position, and by taking the top n candidates (plus one for adjectives with two antonyms) we can retrieve a larger part of the Deese antonyms, at the cost of having (at best) only about 1 in n of the produced candidate antonyms being correct. This is especially useful when the list of antonyms that is automatically computed by the system is to be presented to a human domain expert, who can examine the candidates for each word, validate the true antonyms, and reject the others. As long as recall is sufficiently increased and overgeneration remains tolerable, we have a viable process for the selection of antonyms in a new, possibly specialized, domain. When the system described in this dissertation is used to extract lexical semantic information from a new domain, it is generally much cheaper to have the expert validate the system's proposed solutions than generate the solutions from scratch. And the automatic approach increases the likelihood that solutions that the expert may not initially think of will be presented to him or her, thus increasing the coverage of the resulting lexicon.

Table 9.4 shows the results of this overgenerating approach for several values of n. In this evaluation, I have assumed that the system knows in advance the five words with two antonyms, and considers $n+1$ candidates for them. If this condition does not hold, then the system would be penalized for these polysemous adjectives; e.g., when $n = 1$, the system could at best find only one of the two valid antonyms. Making the evaluation truly blind would thus complicate the interpretation of evaluation metrics such as recall and precision. In any case, the relative scarcity of words with two antonyms means that the difference in

Candidates examined per adjective (n)	Total potential antonyms produced	Correct antonyms found	Recall	Precision	Expected recall of random system	P-value
1	74	31	41.89%	41.89%	1.67%	$6.94 \cdot 10^{-36}$
2	143	35	47.30%	24.48%	3.14%	$5.84 \cdot 10^{-33}$
3	212	38	51.35%	17.92%	4.61%	$3.37 \cdot 10^{-31}$
5	350	47	63.51%	13.43%	7.55%	$1.86 \cdot 10^{-34}$
10	695	55	74.32%	7.91%	14.90%	$2.90 \cdot 10^{-30}$
20	1,385	58	78.38%	4.19%	29.61%	$5.57 \cdot 10^{-18}$

Table 9.4: Performance of the related words recovery subsystem on the Deese antonyms at various levels of response overgeneration. Each line indicates how many responses per adjective in the Deese set (n) are produced; this number is increased by one for the five Deese adjectives with two antonyms. For $n = 1$ we have no overgeneration, i.e., exactly as many antonyms as in the Deese set are generated. The total number of candidate antonyms that are produced, and the number of them that match the corresponding Deese antonym, are also shown. Recall rates are computed on the basis of the Deese set; since the latter contains 74 antonymous relationships (37 pairs, each producing a relationship in each direction), precision will suffer when more than 74 responses are generated. The last two columns indicate the average performance of a baseline system that randomly orders the related word list for each adjective before extracting the same number n of responses per adjective (plus one for adjectives with two antonyms in the Deese set), and the probability that such a system would retrieve the same or more correct Deese antonyms as the real system does.

the resulting scores between the current approach and a truly blind approach would never be more than 5/74, or about 6.75%.[6]

From Table 9.4 I observe that with just five candidates extracted for each adjective, we cover over 63% of the antonyms in the Deese set; if we are willing to put up with 20 candidates per word (out of a maximum of 68 choices), this percentage rises to more than 78%. As noted earlier, the use of semantic orientation information to remove associations

[6] And this bound would be realized only if the system always placed one of the correct antonyms at the $(n+1)$th position for all five words with two antonyms, something which does not happen in the actual evaluation.

between words with the same orientation is expected to reduce the value of n necessary to achieve these recall figures (or, equivalently, increase the recall at a given number of candidates that pass the orientation test and are presented to the domain expert).

I have also listed in Table 9.4 the recall rate that would have been expected if the system ordered the related words randomly, and the probability that the actual recall rates would have been achieved under these circumstances. In all cases, the difference in performance between the actual system and the random baseline is highly significant (P-value less than 10^{-18}), thus empirically validating the adopted approach.

Overgenerating responses increases recall at the expense of (usually significant) drops in precision. This need not be the case: We can keep most of the benefits with only a fraction of the precision penalty if we dynamically modify the number n of candidate antonyms that are retrieved per adjective, on an adjective to adjective basis. The information present in the association strength values between an adjective and its candidate antonyms (which was omitted from Table 9.3 in the interest of brevity) can guide heuristics that select a variable number of retrieved candidate antonyms. I have observed that if the correct antonym is located in one of the top places, there is usually a sharp drop in the associative strength values after the first few related words. For example, the most strongly related word for *slow* is *fast* with an association strength of 117.28, followed by *smooth* at 62.88 and *big* at 62.65; the fourth most strongly related word, *heavy*, has an association strength of only 18.57. The same pattern recurs when the correct antonym is a little bit further down the list of related words. Conversely, if the system does not place the correct antonym in one of the top positions, the association strength values are usually lower, and there is no abrupt drop in them. For example, the system fails to identify *light* as an antonym to *dark* (*light* and *outside* are tied for 30th and 31st place in the list of related words for *dark*), in part because of the strong relationship between *light* and *heavy* that the system is able to retrieve. The top few association strength values for *dark* are 52.74, 43.60, 40.33, 38.90, 30.82, 29.19, ..., with no remarkable differential rates.

Note that the system typically either locates the correct antonym for an adjective near the top of the list of related words for that adjective or misses it completely (i.e., places it near the end of the list). The last two rows of Table 9.4 reflect this property: extracting twenty instead of ten candidate antonyms per adjective almost doubles the total number of words reported as related ones; yet this leads to a meager recall increase of 4.06% (and a sharp drop in precision). This evidence further supports the hypothesis that a variable number of retrieved antonyms per adjective, heuristically guided by the rate of change in the association strength values, would reduce overgeneration and keep precision high without significantly harming recall. I have not implemented this extension yet, so I can only offer plausible arguments for it; but I consider exploring it outside the scope of the dissertation.

9.4 Extracting gradability and semantic orientation labels for the adjectives in the Deese set

Using the clustering techniques of Chapter 2 and the methods of Section 9.3 that process the resulting groupings, I retrieved a list of semantically related words for each of the 69 adjectives in the modified Deese set. This list includes the Deese antonym(s) for an adjective in 42–78% of the cases, depending on how long we are willing to make it. But up to this point, I have not utilized the contrast information that different semantic orientation labels imply. In this section, I apply the algorithms from Chapters 7 and 8 that retrieve such orientation labels to the adjectives in the Deese set.

These algorithms are based on the assumption that the words they classify have some semantic orientation, and are not neutral, qualitative adjectives such as *medical* or *public*. To select the subset of the Deese words for which an orientation label can be produced, I rely on the fact that gradable adjectives usually have an orientation label, and, conversely, non-gradable adjectives usually do not. Consequently, I first determine the gradable words in the Deese set, using the algorithm of Section 6.3.

9.4.1 Selecting the gradable Deese words

In Section 6.4.1 I described how I extracted from the 1987 Wall Street Journal corpus the 496 adjectives which appeared 300 or more times, computed their frequencies in base form, inflected forms, and within the scope of a grading modifier, and assigned to each adjective a label specifying whether it is gradable according to the Collins COBUILD dictionary [Sinclair 1987b]. The 453 adjectives that received a gradability label[7] were then used to train and evaluate the gradability prediction model of Section 6.3; the results suggested a

[7] The remaining 43 words were either erroneously classified as adjectives by the part-of-speech tagger or were sometimes gradable and sometimes non-gradable depending on sense.

modification to the prediction algorithm (automatically marking an adjective as gradable if any of its inflected forms appears within the scope of a grading modifier) which further increased classification performance.

I subsequently used this collection to classify according to gradability individual words and groups of words not in the set of most frequent adjectives (Sections 6.4.1 and 6.4.2). Given the original collection of the 453 adjectives A, and another set B of adjectives to which we want to assign gradability values, I computed the predictor variables (frequencies of inflected forms and grading modifications) and assigned gradability labels from COBUILD to the words in $B-A$. I then trained the prediction model on a sample of $A-B$, and evaluated the computed gradability labels against the ones assigned by hand for the members of B (and for word groups containing elements of B).[8]

I have adopted the same approach for predicting gradability for the 69 words in the modified Deese set (which takes the role of set B in the above discussion). 47 of the Deese words already appear in the original collection A of frequent adjectives. I thus extract the relevant frequency information from the corpus and assign gradability values by hand according to the COBUILD designations for the remaining 22 Deese words. These 22 gradability labels, as well as the 47 labels assigned earlier to the other Deese words, are used for evaluation purposes only; in the corresponding analysis of Section 6.4.2 I did not need to explicitly assign labels to the members of $B-A$ as the evaluation was performed at the word

[8]I used separate sets B for the evaluating the gradability algorithm on individual words and word groups. In the former case, B was a randomly selected subset of A; in the latter case, B was fixed and I did not manually assign gradability labels to its members since the evaluation was performed at the level of word groups.

Classified as gradable:

active alive **bad big clean cold dark deep dirty dry easy empty far fast few full good happy hard heavy high hot large light little long low many narrow near new old passive poor rich rough sad shallow short slow small smooth soft sour strong sweet tall thick thin ugly weak wet wide young**

Classified as non-gradable:

back black **bottom dead front inside left married** pretty **outside right single top** white wrong

Figure 9.3: Automatically obtained classification of the 69 adjectives in the modified Deese set as gradable or not. Correct decisions (according to the COBUILD-based reference model) are indicated in bold.

group level. It is interesting to note here that the Deese set contains overwhelmingly more gradable than non-gradable words (something not true for English adjectives in general, even frequent ones; see Section 6.4.1); 57 of the 69 Deese words are gradable.

Since some of the Deese adjectives were included in the training data for the experiments of Chapter 6, I have to retrain the parameters of the gradability prediction model from a modified training set. I train the model of Section 6.3 (with the adjustment for inflected words that appear modified by a grading adverb) on 200 randomly selected words from the 406 adjectives that belong in the original collection and not in the Deese set. A typical classification of the Deese adjectives according to gradability obtained in this manner is shown in Figure 9.3. The performance of the prediction algorithm on the Deese set is remarkably good: the classification of Figure 9.3 recovers 92.98% of the gradable words, with a precision of 98.15%, and overall accuracy across all classification decisions is 92.75%. Even the gradability assignments that are considered wrong according to the reference model are mostly borderline mistakes (e.g., *white* and *black* are classified as non-gradable, and *alive* as gradable). The results shown in Figure 9.3 are not exceptional. I have repeated the training and evaluation cycle 100 times, with different random samples used for training each time, to ensure that the evaluation would not be affected by a particular sample. The average recall of gradable words in this experiment was 92.00%, the average precision 98.01%, and the average overall accuracy 91.84%.

Once we have selected a subset of the Deese adjectives to which orientation labels can be assigned, we can apply the methods of Chapters 7 and 8 to its member words to automatically retrieve their semantic orientation. It is worthwhile to examine the output of this stage both for the Deese adjectives that have been automatically identified as gradable (see Figure 9.3) and for the Deese adjectives that are actually gradable according to our reference model. The results in the first case correspond to what the unsupervised algorithm for antonym identification actually uses. In the second case, the produced output can be evaluated to judge the performance of the semantic orientation identification component on the Deese set, unencumbered by mistakes made at the earlier stage that filters out the words that have no semantic orientation.[9] Consequently, I will extract orientation labels for the set D_2 containing the 58 words from the Deese collection that are characterized as gradable either in the model or in Figure 9.3.

[9]Given that the automatic method for identifying gradable words is more than 90% accurate, there will be little difference between these two analyses.

9.4.2 Computing semantic orientation labels for the gradable Deese words

A large collection A of 1,336 adjectives was used in Chapters 7 and 8 to train the statistical prediction models and evaluate their results on the tasks of separating positive and negative terms and correctly labeling each of them. This collection consisted of adjectives which appeared at least 20 times in the 1987 Wall Street Journal corpus and which had a clear semantic orientation. Orientation labels were assigned by hand in advance, and were subsequently verified against the labels produced by four independent human judges (see Section 7.2).

Out of the 58 members of set D_2 that I am interested in classifying in this section, 41 belong to the collection A. Although all of the words in set D_2 occur as adjectives in the corpus with frequency higher than the threshold of 20 (see Table 9.1 on pages 337–338), 17 of these words do not have a clear semantic orientation when not subject to contextual restrictions (e.g., *far* or *soft*). As the system selects the words to which orientation labels will be assigned on the basis of (estimates of) gradability, it cannot identify these problematic words. Consequently, orientation information will be computed for all 58 words, but evaluation metrics that score the system's performance will compare the labels for the 41 words for which orientation distinctions are clear and a human-constructed reference model exists. These words form a core group that I will refer to as set D_3.

I have extracted from the corpus the relevant indicators from conjunctions and morphology (see Sections 7.4–7.5) for the 17 new words from the Deese set, and augmented the original collection of words A and links L between conjoined or morphologically related words. In this manner, I construct a new set of words A', containing $1{,}336 + 17 = 1{,}353$ adjectives, and a new set of links L', containing data for 3,139 pairs of words from A'. Note that there are no morphological opposition relationships directly involving two Deese antonyms, and only a few morphological relationships involving a Deese adjective and another adjective (e.g., *happy–unhappy*). This is consistent with prior observations on the relative scarcity of direct morphological relationships between high-frequency antonymous adjectives [Lyons 1977; Quirk *et al.* 1985] as compared to less frequent antonyms.

Following the methodology of Section 7.7, I define subsets of A' and L' according to the density of the graph induced by the links between adjectives. For different values of a parameter α, I define A'_α as the maximal subset of A' with members that have at least α links with other members of A'_α; L'_α is then the set of links that have both endpoints in A'_α. These sets are directly analogous to the sets A_α and L_α of Section 7.7. Note that each A'_α is contained in all sets A'_β with $\beta \leq \alpha$; thus the size of A'_α (and L'_α) decreases as α increases. Because a set A'_α that is too small would lead to misclassification problems, I consider values of α that leave A'_α with at least 200 members. For the actual data extracted from the corpus, this means that the maximum value of α that I examine is five.

In Section 7.7 I used for each value of α the set of links $L - L_\alpha$ as training material and predicted link orientations (whether the two words have the same or different orientation) for the links in L_α; on the basis of those predictions, I then assigned orientation labels to the individual words in A_α. To leave some links for training, α had to be at least two, and its highest value was five. For the present task, however, the set of words for which we must make semantic orientation predictions is fixed: it is the set D_2 of gradable (or thought as gradable by the system) Deese adjectives. Consequently, for each α, I set aside for training the links in $L' - (L'_\alpha \cup L'_D)$, where L'_D is the set of all links involving at least one word from the Deese set[10], so that information about the Deese words is not used during any training phase. Then, I predict same or different orientation for the links in $L'_\alpha \cup L'_D$ and finally orientation labels for the words in A'_α, with α ranging from two to five as before. In general, A'_α will contain many words not in the Deese set. While we ultimately are not interested in the orientation labels that these words will receive, the entire classification operation is much more robust if it is performed on a significantly larger set of words (and associated larger set of links). Once the results are available for A'_α, we can extract the labels assigned to the Deese words that also belong to that set.

The characteristics of the corpus data for the Deese adjectives present a further complication. Two of the Deese words, *near* and *sour*, fail to satisfy the criteria for inclusion in the broadest of the test sets considered, i.e., set A'_2, as each of them appears in a single conjunction in the corpus (*near* with *wide* and *sour* with *sweet*). Since I need to make predictions for the orientations of these two words, I relax the criteria of the preceding paragraph, adding these two words to A'_2 and the two links they participate in to L'_2.[11]

In another departure from Section 7.7, I do not use the entire collection of links available for training ($L' - (L'_\alpha \cup L'_D)$, as explained above), but rather a randomly selected sample of

[10] L'_D contains 976 links, or almost a third of the total number of links.

[11] Had any Deese word appeared in no links, its orientation would have to be guessed on the basis of the distribution of positive and negative labels. The *a priori* probability of a word having negative semantic orientation was estimated to be 50.82% in Section 7.7.

α	Number of adjectives ($\lvert A'_\alpha \rvert$)	Number of links ($\lvert L'_\alpha \rvert$)	Average number of links each word participates in	Links available for training ($\lvert L' - (L'_\alpha \cup L'_D) \rvert$)	Size of sample used for training	Covered Deese adjectives ($\lvert A'_\alpha \cap D_2 \rvert$)
1	1,043	3,139	6.02	0	–	58
2	763	2,869	7.52	199	150	58
3	559	2,481	8.88	495	300	54
4	405	2,043	10.09	818	500	48
5	272	1,529	11.24	1,200	500	43
6	167	1,042	12.48	1,556	–	37
7	99	660	13.33	1,787	–	27
8	55	371	13.49	1,945	–	21

Table 9.5: Characteristics of the corpus data extracted for the classification of the Deese adjectives according to semantic orientation. The adjective sets listed are representative of different densities of the graphs induced by morphology- and conjunction-based links, as encoded by the parameter α. The entry for $\alpha = 1$ represents all the relevant data in the corpus; there are $1,353 - 1,043 = 310$ additional words in the collection A' that do not appear in any conjunction or morphological opposition relationship in the corpus. The entry for $\alpha = 2$ includes two extra words from the Deese set, and the two links they participate in, as described in the text. Set A'_9 is empty, i.e., there is no subset of the words with nine or more links between each of its members and other words in it. Actual training and evaluation was performed for $\alpha = 2$ to 5. Training is impossible with $\alpha = 1$, and the sets A'_α contain less than 200 words for $\alpha \geq 6$.

these links. This is done to avoid overtraining. The size of these samples is fixed in advance to about two thirds of the available links. In Table 9.5 I list the selected sample sizes, along with information on the cardinality of each word and link set defined by the parameter α and the number of Deese words each set A'_α contains.

Experimental results on real corpus data (Section 7.7) and more abstract simulation studies (Section 7.8) have shown that the performance of the algorithm that classifies adjectives according to differences in semantic orientation increases with the density of the conjunction- and morphology-based links between the adjectives. To take advantage of this, I do not perform the training of the model and the calculation of orientation label

α	Size of test set ($\lvert A'_\alpha \rvert$)	Number of links for which predictions were computed	Covered Deese adjectives ($\lvert A'_\alpha \cap D_2 \rvert$)	Covered Deese adjectives with orientation labels in the model ($\lvert A'_\alpha \cap D_3 \rvert$)	Classification accuracy on all adjectives from A'_α with orientation labels in the model	Classification accuracy on covered Deese adjectives with orientation labels in the model
2	763	2,792	58	41	72.21%	70.73%
3	559	2,424	54	38	81.40%	94.74%
4	405	2,003	48	34	87.41%	82.35%
5	272	1,508	43	30	90.81%	86.67%

Table 9.6: Evaluation of the classification algorithm for semantic orientation on the Deese adjectives. Scores are given for various adjective sets with different densities in links between their members, as represented by the parameter α. For each test set A'_α, the table lists the accuracy obtained on the all the words in the test set which have orientation labels in the reference model (i.e., $A'_\alpha - (D_2 - D_3)$) and on the Deese words from that test set with orientation labels in the model (i.e., $A'_\alpha \cap D_3$).

predictions on just the first set that covers all the Deese adjectives (i.e., set A'_2). Instead, I analyze all the sets A'_2 to A'_5, computing generally different labels from each set for their overlapping members.[12] Then I combine the suggested labels by assigning to a word the orientation label proposed for it in the smallest (and densest in links) set A'_α (the one with the largest α) the word is a member of.

Table 9.6 shows the scores obtained by the classification algorithm for the various values of α. Evaluation was performed against the hand-assigned orientation labels for both the entire set A'_α (except for those Deese words which were not assigned orientation labels in the reference model) and its subset that consists of Deese adjectives with a clear semantic orientation (i.e., the words in $A'_\alpha \cap D_3$). As in the experiments previously reported in Section 8.10, the average frequency ratio method correctly identified the group of positive

[12]Recall that $A'_\alpha \subseteq A'_\beta$ if $\alpha \geq \beta$.

Classified as positive:

active big clean deep easy full good happy large little **rich** shallow **smooth strong sweet tall white**

Classified as negative:

bad black cold dark dirty empty fast **hard** high hot **low narrow passive poor** pretty **rough sad slow small sour ugly weak** wide **wrong**

Figure 9.4: Retrieved semantic orientation labels for the Deese adjectives in set D_3, i.e., the Deese adjectives that have an orientation label in the reference model constructed in advance by humans. Correctly matched adjectives are shown in bold.

---

terms in all cases.

The scores in Table 9.6 monotonically increase with α for the entire test set, as expected. They also increase in a similar manner for the Deese words, with the exception of $\alpha = 3$. Generally, the scores are somewhat lower for the Deese words than for all the words classified in each case, which is unfortunate for our purposes but may be due to the increased polysemy of the Deese words. Note that the aberration of significantly higher classification scores for the Deese adjectives in set A'_3 means that it would have been optimal to ignore the results for $\alpha = 4$ and 5. However, the system cannot know this property of the data in advance, so the general method that combines orientation labels by selecting for each word the label suggested by the run with the highest α was applied on the produced label sets. The labels assigned to the 41 words in set D_3 by this method are listed in Figure 9.4. The accuracy obtained by the label combining method using the labels from the sets A'_2 to A'_5 is 81.39% on all 763 words in the test sets, and 82.93% on the 41 Deese words with orientation labels in the reference model.

9.5 Combining evidence from semantic groups, gradability, and semantic orientation

In the previous section I discussed automatic approaches that retrieve gradability labels with 92.75% accuracy and semantic orientation labels with 82.93% accuracy when applied to the Deese adjectives. The purpose of computing this semantic information is to combine it with the ranked lists of associated words extracted for the Deese adjectives according to the methods of Section 9.3 (see Table 9.3 on pages 349–353); it was shown earlier that these lists are already useful for locating antonyms, since antonyms to a given adjective are frequently placed near the top of list corresponding to that adjective. There are two, largely orthogonal, ways to use the learned gradability and orientation values to further restrict membership in the list of candidate antonyms for a given word (and, indirectly, differentiate between antonyms and synonyms):

- Two antonyms tend to be both gradable or both non-gradable. Therefore, if a word B appears in the list of candidate antonyms for a word A but A and B have different gradability labels, B should be eliminated from that list.
- Two antonyms tend to have different semantic orientations (assuming that both have some kind of evaluative semantic orientation). Therefore, if a word B appears in the list of candidate antonyms for a word A, both A and B are gradable (and thus orientation inferences are probably valid for them), and the same semantic orientation label has been assigned to both words, then B should be removed from the list of candidate antonyms for A.

These observations offer a straightforward way to remove from the list of candidate antonyms for a word A all words for which gradability and/or semantic orientation information indicates that they cannot be antonyms of A. By eliminating words that may be otherwise semantically related to A, the true antonyms advance nearer the top of the list. Furthermore, this operation can be performed in a single pass in linear time. Before, however, I apply this algorithm to the data extracted for the Deese adjectives, it is prudent to examine the potential harm erroneous decisions in this step can cause.

If for some reason the system misidentifies the gradability and/or orientation of the original word and its true antonym(s) so that they violate the conditions set above, the result could be catastrophic. The correct antonym would be removed from the list, whereas other evidence from semantic groups might strongly point at it. On the other hand, if almost all of the decisions of this last combining step are correct, or if errors tend to be made in cases where the true antonym was not placed very high in the original list, both the recall and the precision of the system can be raised significantly.

There are three factors that influence the effect of the combining step on the lists of words originally produced on the basis of grouping information. First, if the original list is already of high quality, its potential for improvement is limited; if the correct antonym is already placed at the top of the list, there is nothing to be gained from gradability and orientation information when a constant number of proposed antonyms is extracted from the list (except, perhaps, for increased confidence in the system's decision).[13] Second, there

[13] If the extension discussed in Section 9.3 is implemented and a variable number of proposed antonyms is extracted from each list according to association strength differentials, then gradability and orientation information can help make that number smaller when the correct antonym is already at the top of the list.

are words that don't have a clear evaluative semantic orientation, and this makes it harder for the system to assign meaningful orientation labels for them. In such cases, the likelihood of an error during the filtering of the lists of candidate antonyms is higher. For example, one of the pairs in the Deese set is *thick–thin*; but it is not clear what the orientation of these words is, if indeed they do have a semantic orientation. It would look most plausible to classify both of them as negative, which the system actually does. Then, the expectation that two gradable antonyms should have different semantic orientations is violated and the other evidence in favor of their closeness in meaning is ignored. This is not a case where the system's learning component is at fault. Rather, it illustrates that the definitions of gradability and semantic orientation, and the rules and constraints derived from them, are inexact and cannot be always applied successfully even by humans.

Finally, what we really want to determine for the filtering of candidate antonyms is whether two words have the same or different gradability or orientation, not the actual values of these two properties. Assuming that the linguistic hypotheses about the validity of gradability and orientation hold for these two words (as they do for most pairs of words), there are two ways to achieve the desired result: either predict the correct label in both cases, or predict the wrong label for both of them. If the probability of identifying the correct label for each word is p, the probability of identifying the relationship between the two labels correctly is

$$q = p^2 + (1-p)^2 = 2p^2 - 2p + 1$$

Comparing the above probability with p, I note that

$$q - p = 2p^2 - 3p + 1 = (2p - 1)(p - 1)$$

and this quantity is negative for all $p \in (\frac{1}{2}, 1)$! Thus, assuming our system has some (but not perfect) success in predicting the labels, it will always do worse when predicting the relationship between two labels. According to the data of the previous section, the probability of correctly determining whether two words are both gradable or both non-gradable is therefore 86.55%, and the probability of correctly determining that two gradable words have the same or different semantic orientation is 71.69%.

Given the above potential pitfalls, and the fact that prediction of gradability is much easier than prediction of semantic orientation (see Section 9.4.1), I have modified the second of the elimination criteria to keep a word in the list of potential antonyms if it is rated as the strongest candidate on the basis of semantic grouping information, even if it appears to have the same orientation as the word for which the list was constructed. I then applied the gradability restriction and the modified orientation restriction criteria to the information collected in the previous sections. Table 9.7 indicates the change in performance when the gradability, orientation, or both of these stages are added to the association information provided by the semantic groups of the Deese words, while Table 9.8 shows the ten most strongly associated words for each Deese adjective that survive after both of the filtering steps have been applied to the original association data; the latter table should be compared with Table 9.3 on pages 349–353. Note that for many of the Deese adjectives, the filtered list of candidate antonyms looks less cohesive than before, because words that were strongly

	$n = 1$	$n = 2$	$n = 3$	$n = 5$	$n = 10$	$n = 20$
Recall						
Semantic groups	41.89%	47.30%	51.35%	63.51%	74.32%	78.38%
Semantic groups plus gradability	47.30%	51.35%	55.41%	63.51%	74.32%	77.03%
Semantic groups plus orientation	41.89%	47.30%	58.11%	71.62%	77.03%	78.38%
Semantic groups plus gradability and orientation	47.30%	54.05%	60.81%	72.97%	75.68%	75.68%
Precision						
Semantic groups	41.89%	24.48%	17.92%	13.43%	7.91%	4.19%
Semantic groups plus gradability	47.30%	26.57%	19.34%	13.43%	7.91%	4.12%
Semantic groups plus orientation	41.89%	24.48%	20.28%	15.14%	8.20%	4.19%
Semantic groups plus gradability and orientation	47.30%	27.97%	21.23%	15.43%	8.06%	4.04%

Table 9.7: Performance of various combinations of modules of the antonym recovery subsystem on the Deese adjectives. Each row corresponds to a combination of the indicated semantic information sources; the "semantic groups" entry repeats the evaluation scores of Table 9.4 which establish a baseline for comparison purposes. Recall and precision are listed at various levels of overgenerating responses. Each column indicates how many responses (n) per adjective in the Deese set are produced; this number is increased by one for the five Deese adjectives with two antonyms.

related with, but not opposed to, the main word for that entry (and with each other) have been eliminated.[14]

I observe from Table 9.7 that the introduction of the additional gradability or semantic orientation information offers in almost all cases significant improvements in the performance of the antonym identification algorithm, despite the potential negative effects of errors in these knowledge sources that were discussed above. The increase occurs in both recall and precision, and is of the order of 10–20% relative to the baseline value of the metric

[14]This reduction in semantic cohesion is not universal; see, for example, the revised list of candidate antonyms for *good* in Table 9.8, which includes words such as *poor* that are not in Deese's set.

Adjective	Deese antonym(s)	Most related words identified by the system after filtering according to gradability and orientation	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
active	passive	**passive** small thin far sour thick narrow wide fast long	1	–
alive	dead	old small empty dirty far heavy rough wet sad low	47	–
back	front	**front** single wrong dead inside left top outside black bottom	1	–
bad	good	rough **good** sweet easy shallow strong dry happy young active	2	–
big	little	fast slow heavy light thin sour small far dirty long	46	–
black	white	**white** married dead bottom wrong pretty outside inside back right	1	–
bottom	top	black white pretty inside dead wrong married single front back	41	–
clean	dirty	empty **dirty** old small hard fast rough few narrow sad	2	–
cold	hot	**hot** dry new short near easy strong large deep full	1	–
dark	light	near shallow new tall full deep dry happy easy smooth	44	–
dead	alive	married black white single wrong back bottom outside pretty inside	41	–
deep	shallow	**shallow** narrow sour wide passive dark thick long ugly empty	1	–

Table 9.8: The ten most strongly associated words for each adjective in the modified Deese set, after filtering out candidate antonyms that fail the gradability or semantic orientation tests. For each word, the Deese antonym and the ten most associated words that are retrieved by the enhanced system version that includes the gradability and orientation components are listed, in order of decreasing overall association strength. Five of the Deese adjectives have two antonyms within the Deese set; for these five words, two antonyms are listed in the second column, with the primary antonym (the most frequent response to the word) listed first. If the Deese antonym(s) appear in the list of ten best recovered words, they are shown in bold. The last two columns give the ranks of the primary and secondary (if any) antonyms within the entire ordered list of recovered associated words, after the filtering operations have been performed. Ranks vary from 1 (best) to 68, and may take non-integer values in the presence of ties. Because of the filtering stages, only a total of eight candidate antonyms survive (and are listed) for *top*; similarly, only nine candidate antonyms survive for each of *married* and *white*. The table is continued on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system after filtering according to gradability and orientation	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
dirty	clean	**clean** full new easy young large big short rich alive	1	–
dry	wet	poor cold hot **wet** dark soft high sour bad light	4	–
easy	hard	sweet sad **hard** dirty empty rough bad old narrow dark	3	–
empty	full	clean large new **full** easy near shallow deep rich short	4	–
far	near	weak strong rich shallow **near** active large little big young	5	–
fast	slow	**slow** big smooth strong clean large near deep active short	1	–
few	many	**many** sweet clean smooth tall little alive new shallow near	1	–
front	back	**back** single left top inside outside bottom wrong dead pretty	1	–
full	empty	new rough **empty** ugly dirty hard dark thick thin narrow	3	–
good	bad	**bad** poor sour sad rough ugly thin hot fast weak	1	–
happy	sad	young old sour high dark passive bad light slow low	50	–
hard	soft easy	sad new full **easy** shallow clean near smooth deep tall	44.5	4
heavy	light	**light** big sweet smooth shallow little strong active alive clean	1	–
high	low	**low** strong dry happy smooth easy young active deep large	1	–
hot	cold	**cold** dry large good strong active big full sweet happy	1	–
inside	outside	**outside** top single left bottom front back black dead wrong	1	–
large	small	**small** empty thin far passive wide poor narrow hot sour	1	–

Table 9.8: The ten most strongly associated words for each adjective in the modified Deese set, after filtering out candidate antonyms that fail the gradability or semantic orientation tests. Continued from the previous page and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system after filtering according to gradability and orientation	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
left	right	wrong **right** front top inside back outside single dead bottom	2	–
light	dark heavy	**heavy** big sweet dry happy smooth strong easy young active	42.5	1
little	big	alive wide narrow passive thick thin far long heavy fast	46	–
long	short	**short** smooth near deep little active shallow big strong large	1	–
low	high	**high** strong smooth dry happy shallow young active rich easy	1	–
many	few	**few** sweet smooth tall little alive new shallow near big	1	–
married	single	dead white black **single** pretty bottom inside wrong right	4	–
narrow	wide	**wide** little deep large shallow easy smooth active full new	1	–
near	far	new dark **far** long rough small empty old hard fast	3	–
new	old	near empty hard dark dirty **old** rough thin thick sad	6	–
old	new young	alive happy **young** clean **new** easy near tall full shallow	5	3
outside	inside	**inside** top single wrong black front left back dead bottom	1	–
passive	active	**active** little deep large strong happy rich shallow short dry	1	–
poor	rich	dry good tall large **rich** strong active happy big full	5	–
pretty	ugly	single right married white bottom black dead inside back front	40	–
rich	poor	sour small far **poor** empty passive dirty soft bad thick	4	–
right	wrong left	**wrong left** pretty black back front outside bottom inside married	1	2

Table 9.8: The ten most strongly associated words for each adjective in the modified Deese set, after filtering out candidate antonyms that fail the gradability or semantic orientation tests. Continued from the previous page and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system after filtering according to gradability and orientation	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
rough	smooth	bad full short new **smooth** shallow near easy good deep	5	–
sad	happy	hard easy sweet good new smooth shallow full active deep	46	–
shallow	deep	**deep** dark far rough small hard empty narrow wide bad	1	–
short	tall long	**long** thick rough ugly fast soft empty dirty cold old	48	1
single	married	pretty front inside outside top **married** back dead white left	6	–
slow	fast	**fast** smooth big strong sweet dry happy active easy shallow	1	–
small	large	**large** rich active clean shallow near young big alive strong	1	–
smooth	rough	long slow fast **rough** narrow hard sour heavy far sad	4	–
soft	hard	wet strong dry short rich happy young smooth sweet alive	43	–
sour	sweet	rich deep good strong active big large happy smooth dry	43	–
strong	weak	**weak** soft far wet sour thin low high fast slow	1	–
sweet	sour	easy bad sad few heavy many light ugly slow hard	27	–
tall	short	poor dark old hard rough few many narrow sad wet	46	–
thick	thin	**thin** short little full new active deep large near shallow	1	–
thin	thick	**thick** large active little strong new big full good deep	1	–
top	bottom	inside outside single left front back dead wrong	38.5	–
ugly	pretty	full short good deep sweet large big strong near active	43	–

Table 9.8: The ten most strongly associated words for each adjective in the modified Deese set, after filtering out candidate antonyms that fail the gradability or semantic orientation tests. Continued from the previous page and on the next page.

Adjective	Deese antonym(s)	Most related words identified by the system after filtering according to gradability and orientation	Rank of primary Deese antonym	Rank of secondary Deese antonym (if any)
weak	strong	**strong** dry large good near smooth little happy sweet alive	1	–
wet	dry	soft strong **dry** shallow smooth near large big happy deep	3	–
white	black	**black** married dead bottom pretty single inside wrong right	1	–
wide	narrow	**narrow** little deep large shallow active strong near tall clean	1	–
wrong	right	**right** left black back dead outside bottom front single inside	1	–
young	old	happy **old** small dirty far bad fast sour low sad	2	–

Table 9.8: The ten most strongly associated words for each adjective in the modified Deese set, after filtering out candidate antonyms that fail the gradability or semantic orientation tests. Continued from the previous page.

(when just the information from semantic groups is used). The benefit is most pronounced when a small (but not minimum) number of candidate antonyms is selected, as expected. In fact, there is a slight degradation of performance when the largest number of antonyms per Deese adjective, 20, is produced. But this takes place only for one knowledge source, gradability, and in only one of the cases analyzed.

These results clearly justify the introduction of the two additional knowledge sources, which, in almost all cases, co-operate to further increase the performance when they are used together. It is informative to also study evaluation metrics that do not depend on a particular number of selected antonyms per word. In most cases, the additional constraints from gradability and orientation improve the position of the correct antonym in the list of words associated with a given word. But this improvement may not be enough to cross one

	Average rank of true antonym(s)	Better	Worse	Ties	P-value
Semantic groups	13.1824	–	–	–	–
Semantic groups plus gradability	12.2027	29	3	42	0.0034
Semantic groups plus orientation	13.0338	21	6	47	0.1030
Semantic groups plus gradability and orientation	12.0473	32	8	34	0.0071

Table 9.9: Intrinsic properties of the rankings produced by various combinations of modules of the antonym recovery subsystem on the Deese adjectives. Each row corresponds to a combination of the indicated semantic information sources; the "semantic groups" entry represents the baseline version that does not use additional information besides associations extracted from word clusterings. The average rank in the produced list of associated words for the true antonym(s) specified in the Deese set can vary from 1.0676 (best) to 67.9324 (worst). The columns labeled "better" and "worse" display the number of antonyms out of the 74 in the Deese set for which the introduction of the additional semantic knowledge improved or worsened, respectively, their rank. The final column lists the probability that such an improvement and deterioration record would be matched or exceeded by chance, and was computed with an exact non-parametric sign test [Gibbons and Chakraborti 1992].

of the cutoff points n and have a measurable effect on the scores of Table 9.7. For example, improving the rank of the correct antonym for a given word from 10 to 7 does not change any of the measures in that table, while an improvement from the 6th to the 5th, or from the 11th to the 10th, position in the list of candidate antonyms would improve the scores for $n = 5$ or $n = 10$ respectively. To account for this implicit organization of ranks into ranges, I list in Table 9.9 intrinsic measures of the quality of each ranking of the words: the average rank assigned to the correct antonym(s)[15], and the number of antonyms across all Deese words whose rank is better or worse in an alternative version of the antonymy algorithm that utilizes semantic orientation, gradability, or both relative to the baseline system that

[15]There are 69 words in the modified Deese set, and 74 data points across which the average is computed, since five of the Deese adjectives have two antonyms. The best possible average rank is $(69 \cdot 1 + 5 \cdot 2)/74$, or 1.0676; the worst possible average rank is $(69 \cdot 68 + 5 \cdot 67)/74$, or 67.9324.

is based on grouping only. The latter information allows the calculation of the probability that a system with the same level of performance as the baseline system would achieve the observed, or better, improvement and deterioration characteristics on individual antonyms purely by chance. As Table 9.9 shows, the difference in the performance of the system is weakly significant statistically (at the 10% level) when orientation only is considered, and strongly significant (at the 1% level or less) when the other two cases of gradability alone or both orientation and gradability are analyzed.[16]

9.6 Discussion of the results

I have presented a three-pronged approach for automatically learning the antonyms of adjectives from free text, and applied it to the standard collection of the Deese antonyms [Deese 1964]. The system correctly identified the antonyms for the Deese words in 47.30% of the cases when it was allowed to produce only a single best choice for each potential antonym. By producing more than one candidate antonym for each word, the system achieves 61% recall at 21% precision when three candidates are proposed for each word, and 73% recall at 15% precision when five candidates per word are proposed. The precision figures given above should be interpreted by taking the factor of overgeneration into account. Clearly, when the evaluation model has x correct answers and the system generates $k \cdot x$ responses, the maximum precision is $\frac{100}{k}$%, and this maximum will be attained if and only if the x

[16]The small number of data points (74), and the large number of ties (which is expected given the nature of the filtering processes), contribute towards the reduction of the statistical significance of the performance differences to the levels mentioned above.

correct responses are included in the kx responses that the system reports. Taking into account the extra responses generated for the five adjectives with two antonyms in the Deese set, this means that the system achieves 21.23% precision out of a possible maximum of $\frac{69+5}{3 \cdot 69+5} = 34.91\%$ when three candidate antonyms are proposed for each word, and 15.43% precision out of a possible maximum of $\frac{69+5}{5 \cdot 69+5} = 21.14\%$ when five candidate antonyms are proposed for each word. In other words, the system reaches, in these two cases, levels of precision at 60.81% and 72.99% of their corresponding maxima.

Furthermore, the linguistically motivated information sources of gradability and semantic orientation contribute to the ability of the system to automatically identify antonyms, as attested by Tables 9.7 and 9.9. Not only is this useful for the practical application of the algorithm, but also provides an independent validation of the relevance of these two linguistic properties to antonymy.

The obtained scores are more remarkable when we consider that the Deese set is perhaps not the most suitable for evaluating the performance of the system. The Deese words include several that are primarily adverbs, prepositions, or nouns, and they are all highly frequent and polysemous. The antonyms among them are rigid and fixed in advance (for example, the system is penalized for retrieving *good* as an antonym for *poor*, since this relationship is not in Deese's set), and were selected according to the responses of humans out of context; thus, they cannot capture any domain-specific oppositions that the system might automatically identify. In this way, differences in antonymy because of different senses or domain usage are not accounted for; for example, *single* is expected to be the antonym of *married* according to the Deese set, but in the 1987 Wall Street Journal corpus it overwhelmingly has the sense of “only” or “unique”, e.g., *the biggest single investor*, *the single largest expenditure*. Furthermore, several of the words in the Deese set are exceptional in their behavior as adjectives (e.g., *many*), violating some of the theoretical expectations about semantic orientation on which the system’s algorithms are based.

I believe that the performance of the present system can be further improved by taking advantage of intermediate measures that are already being computed and could serve as indicators of the system’s confidence in its decisions. For example, a measure of how strongly a particular word is believed to be gradable or have a given semantic orientation would help in deciding whether this information should be used to filter the list of candidate antonyms identified from automatically constructed semantic groups. The latter list, which contains the words strongly associated with another word, could also separately be pruned in a selective manner according to how fast the association strength measure declines. These extensions are likely to improve the system’s recall and precision by limiting the number of risky filtering operations in the first case, and considerably shortening the list of potential responses in the second.

Part III

Related work, applications, future extensions, and contributions

This last part of the dissertation contains the end material that supports, reviews, and extends the core of the thesis presented in Chapters 2–9. Following the time-honored paradigm of numerous other dissertation authors, I have organized this end material into four chapters:

- Chapter 10 contains a brief overview of related previous (and contemporary) research. A vast amount of literature from traditional (non-computational) and, more recently, computational linguistics is broadly related to the several topics this dissertation addresses. Among those I focus mostly on linguistics work that set the foundations for the approaches I have taken, and on computational linguistics work that is most directly related either to the particular problems I address or to the methods I have adopted.

- Chapter 11 discusses several potential applications for the knowledge extracted by the present system. Extensive testing of the system I describe in this thesis has been performed, on different categories of words (adjectives and nouns) and different domains (general news and financial reports). Since this thesis is accompanied by a fully implemented and evaluated system, I have not included a specific application with it, an approach which commonly provides a way to empirically test the conclusions of a more theoretical dissertation. However, I have identified several applications that could be implemented as extensions of the thesis, and describe in some detail three such applications from the medical domain.

- Chapter 12 lists several future extensions to the system presented in this thesis, in the form of either additional processing modules, additional knowledge sources, or refinements in the methods that are currently employed. The limitations of the approaches I have adopted for the various components of the thesis, limitations that may be either inherent in the design or due to the lack of unlimited time for the completion of the dissertation, are also assessed, and ways to rectify some of them in the future are presented.

- Finally, Chapter 13 summarizes the main points of the dissertation and lists the contributions it offers, both from a technical point of view to the computational and non-computational linguistics fields and from a methodology point of view to a broader list of fields (linguistics, psycholinguistics, and information retrieval).

Chapter 10

Related work

In this chapter, I briefly examine related previous (and contemporary) work from both computational and non-computational linguistics. The latter usually provide the theoretical foundation upon which computational approaches build. Since the thesis covers a number of different interrelated topics, many pieces of previous research could have been included in this chapter. Rather than attempting to provide an exhaustive description of all the related work, I focus instead on only the most relevant research. I also omit from this chapter many computational approaches that are only marginally related to this work although they use similar methods to achieve their goals. This includes, for example, work on corpus-based sense disambiguation.

Following the division that I have adopted in the rest of this dissertation, the related work is separated into two main sections, plus a section on evaluation. Section 10.1 describes research related to the first stage of the system (word grouping), the results of which are used by all subsequent stages. Section 10.2 presents work related to the other stages of the system, which further analyze and process the semantic groups formed by the first stage.

The final section discusses related work for the problems of evaluating the system's results and interpreting the meaning of the evaluation scores.

10.1 Work related to semantic word grouping

The semantic groups that the first stage of the system aims to find, and the associated concept of semantic relatedness, have been studied extensively in linguistics. Trier [1934] was the first researcher that focused on these groups and on the relations between their members. He proposed that the extra-linguistic concepts which the words represent are organized in non-overlapping sets, which he called *conceptual fields* ("Sinnfeld"). As a result of this underlying conceptual organization, the lexical items of the language are organized in corresponding groups of semantically related words, which Trier called *lexical fields* ("Wortfeld"). Trier was most interested in the diachronic and cross-linguistic aspects of lexical fields, especially between modern German and its predecessor (Old High German). But his students and other researchers expanded his theory, and emphasized its importance for the general organization of the lexicon. Ullmann [1957] and especially Lehrer [1974] provide comprehensive treatments of the theory of lexical and semantic/conceptual fields, while summaries of the theory can be found in most general discussions of semantics (e.g., [Kempson 1977; Lyons 1977]). In addition to the work in theoretical linguistics, psycholinguists were also drawn to the properties of lexical fields, and carried out several experiments measuring the relatedness of words on the basis of the strength of their association by human subjects in a stimulus-response framework [Osgood *et al.* 1957; Deese 1964].

Semantic groups have also been found useful for computational purposes, although so far the only widely available classifications of words into groups have been done by hand. WordNet [Miller *et al.* 1990] is perhaps the most well known example of a large scale classification of words according to semantic criteria. WordNet organizes words into sets of synonyms called *synsets*, and provides links between synsets which represent several types of semantic relationships, including antonymy, hyponymy, and meronymy. The semantic word classification provided by WordNet has been used in several natural language systems, e.g., [Resnik 1993; Resnik and Hearst 1993; Knight and Luk 1994; Basili *et al.* 1994; Jing *et al.* 1997]. Another application of semantic groups has been discussed by Sadler [1989] and Hindle [1990], who proposed pooling the frequencies and probabilities of words within each group in order to deal with the data sparseness problem during corpus-based language processing in general and statistical language modeling in particular.

While the work on lexical fields by linguists has highlighted both the importance of the problem and some of the properties of these fields, it suffers from two drawbacks when the design of a system that automatically finds these groups is considered: First, no rigorous formal definition of what exactly a lexical field is or what criteria two words should meet to belong in the same field has been provided. Second, the informally defined criteria for determining semantic relatedness are not amenable to direct implementation on a computer (see Section 1.8 for evidence supporting this claim).

In Section 1.6 I discussed the first problem, and concluded that, until the development of a sufficiently general and acceptable formal theory of lexical fields in linguistics, alternative informally defined criteria should be used for semantic relatedness and semantic groups.

For the second problem, I have replaced the direct application of the high-level semantic relatedness criteria with properties of the words that can be directly observed in a corpus. These properties relate to co-occurrence patterns of the words that are grouped with other appropriately selected *marker* words. Another extensive body of work in linguistics provides justification to the claim that some of the surface co-occurrence relationships could map to, or be indicative of, deeper semantic relationships. Firth [1957] strongly advocated this position ("words shall be known by the company they keep"), and other linguists have provided support for this thesis (e.g., [Harris 1951; Greenberg 1966; Harris 1968]). Arguments for this functional view of lexical semantics can also be found in the philosophy of language literature, particularly in the work of Wittgenstein [1972]. Perhaps because it is easy to retrieve indicators of co-occurrence with a computer, the above thesis has been adopted either implicitly or explicitly by a number of researchers working on a variety of computational linguistics problems, including sense disambiguation [Brown *et al.* 1991; Dagan *et al.* 1991; Gale *et al.* 1992b; Liddy and Paik 1992; Yarowsky 1993; Dagan and Itai 1994] and collocation extraction [Smadja 1993] and translation [Smadja *et al.* 1996].

Several researchers have proposed systems that retrieve semantic groups using different forms of the co-occurrence (or distributional) hypothesis described above. In the fifties, at a time when statistical approaches to language in general and lexical semantics in particular enjoyed considerable support, Karen Sparck-Jones designed a semi-automatic system for this task. Sparck-Jones' [1986] system used a dictionary to manually identify sets of related words, called *rows*, for each word of interest. Similarity between rows is measured on the basis of their common elements, and a simple hierarchical clustering algorithm assigns the words into groups.

Most likely because of the lack of suitable machine-readable corpora at the time of the system's development, Sparck-Jones' system offers only partial automation for the task. Furthermore, the value of the system's results is limited since a human needs to provide *semantic* information (through the sets of related words), essentially determining the final results to a significant degree. For example, there is no way that domain-dependent information could be entered in the system, save for extracting rows from a specific domain-dependent dictionary that is already available; but even then, the system would merely reflect what was implicitly present in that dictionary. Other criticisms can be made for the parts of the system that work without human supervision, in particular about her choices of the similarity metric and clustering method. Nevertheless, Sparck-Jones' system was one of the first that exploited semantic relatedness relationships for the organization of the lexicon.

Later, psycholinguists used measurements of similarity between words to classify them into semantic groups. Unlike Sparck-Jones, who based similarity on dictionary definitions, their source was the responses of human subjects who were asked either to rate the similarity between any two words or to place the words into groups. Miller [1969] used this approach to classify 48 common nouns into semantic groups, and further studied these groups to retrieve the criteria that the human subjects implicitly used in their classification. Fillenbaum and Rapoport [1971] performed a similar but more extensive study with various groups of words, including adjectives of color and terms of kinship.

The interest in the problem was renewed in the computational linguistics community with the advent of reasonably sized machine readable dictionaries in the mid-80's. Several researchers worked on the problem of identifying *semantic neighbors* of any given word [Chodorow *et al.* 1985; Lesk 1986; White 1988; McDonald *et al.* 1990; Veronis and Ide 1990; Guthrie *et al.* 1991], usually producing an ordered list of words according to some measure of similarity based on the dictionary definitions of the words or on information such as topic (subject) codes assigned to each of the word senses in the dictionary. Since dictionaries provide separate definitions (and frequently additional labels) for each word sense, sense disambiguation was an obvious application of the semantic neighbors information. Thus, many of the papers cited above addressed this problem by recomputing neighbor information for an occurrence of a word from the surrounding context and labeling that word instance with the dictionary sense which most closely matched the retrieved neighbor profile. More recently, Resnik [1993; 1995] used the taxonomic information present in WordNet [Miller *et al.* 1990] to measure the semantic similarity between two words according to their common superordinate concepts.

With the wide availability of large corpora at the end of the last decade, and the realization of the limitations of the dictionary-based approach, other researchers have worked on the problems of automatically measuring the semantic similarity between words directly from the text. Schütze [1993] used co-occurrence information to derive a *vector space* representation for each word. He then employed this representation to measure the similarity between words using information retrieval methods (the cosine measure) and applied the results to the problem of word sense disambiguation [Schütze 1992]. Grefenstette [1992b] used context information extracted with a syntactic analyzer to compute similarity measurements between words. The similarity was based on a simple measure of matching between the context vectors (the Jaccard coefficient). In the future work section of his Ph.D. thesis, Smadja [1991] proposes to use the strength of collocational association that his XTRACT system [Smadja 1991; Smadja 1993] computes as a measure of semantic association, compute a vector of association measures for each word in a given set with other words in the language, and then use the differences in these vectors to classify the words in the given set into semantic groups. Smadja did not implement his proposal, and it is not clear that collocational association would be a good measure for indirectly measuring semantic similarity. Nevertheless, his proposal, as well as the work of Schütze and Grefenstette, share elements with the similarity computing part of the first stage of the system that I discuss in this dissertation.

More recently, two word classification systems that explicitly target the problem of word classification have been developed, by Brown *et al.* [1992b] at IBM and by Pereira *et al.* [1993]. These systems are the most similar to the one I have implemented among the work discussed above. Brown *et al.*'s and Pereira *et al.*'s systems both use an approach to semantic similarity that is based on information theory, employing measures such as mutual information and relative entropy (Kullback-Leibler distance) to estimate the similarity between words (or between words and cluster centroids). They also share the same motivation for the development of semantic groups, namely their belief that such groups can help in alleviating problems of sparse data for statistical language modeling.

My own work on producing semantic groups has evolved independently of the above efforts[1], and approximately at the same time.[2] My approach differs from the other contemporary solutions to the problem in many of the details, but also in several important aspects:

- I have examined different alternatives for the similarity measure and the clustering algorithm in considerable detail. As a result, I use a robust measure of similarity that satisfies the desirable properties I have identified (see Section 2.3), that is, independence from the scale of the compared frequencies, invariability to linear and generally order-preserving transformations of these frequencies, a bounded range for the produced measure, asymmetry in treating the observed word associations as more important than the many potential ones that have not been observed, robustness in the presence of outliers and low frequencies, and amenability to statistical interpretations of the computed similarity values. I also employ a non-hierarchical clustering algorithm that outperforms hierarchical alternatives (see Sections 2.7.1 and 2.7.4). The methods I have adopted are not used by any of the other researchers, who generally follow the established practice in the particular approach to statistical natural language processing they adhere to, using methods either from information retrieval [Grefenstette 1992b; Schütze 1993] or from information theory [Brown *et al.* 1992b; Pereira *et al.* 1993]. In general, the current approach is distinguished by its more exhaustive examination of alternatives for the technical issues of measuring similarity and

[1] I had though the benefit of extensive discussions with Frank Smadja, who suggested to me the problem of grouping adjectives. While I ended up not using his collocation-based approach for measuring similarity, I was certainly influenced by the model he proposed for the task.

[2] A first version of the word grouping subsystem was operational in late 1991; this version was extended and completed during 1992.

clustering, including experimental studies to verify the superiority of the proposed methods.

- I explore syntactic information as much as possible and in several different ways. Words are analyzed separately for each syntactic (part-of-speech) class, so that specialized methods that apply to only one class can be taken advantage of. This is not done by the other researchers; for example, Brown *et al.* classify all syntactic classes simultaneously, while Pereira *et al.* address the problem with a syntactic-class-specific solution but only for one class (nouns). Grefenstette and Schütze do not explicitly compute word classes, but rather extract ordered lists of words related to a given word; in both cases, words from all syntactic classes are handled together when the candidate members of these lists are considered. When the syntactic class has been fixed, the system I have developed takes maximum advantage of syntactic information, so that co-occurrence relationships are retrieved with high recall and precision. Different types of syntactic information can be used. For example, adjective-noun and noun-noun relationships are currently employed to classify nouns, while the framework provides for the future incorporation of subject noun-verb, and verb-object noun relationships for the same task. Again, several of the alternative approaches do not use any syntactic information at all [Brown *et al.* 1992b; Schütze 1993], while others use less information of this type than I do, limited to one syntactic relationship only [Pereira *et al.* 1993]. In Chapter 5 I showed that syntactic knowledge can offer significant performance improvements to the classification system; and trends in the system's performance as the training corpus grows larger indicate that these improvements cannot be offset by simply increasing the amount of training data (corpus size). Evidence of the relevance of syntactic information for the determination of the semantic relatedness of words, and in particular of the semantic constraints that adjective-noun modification implies, is also offered by Justeson and Katz [1995a]. They pursue the *principled* disambiguation of adjectives according to the nouns they modify, and empirically establish that different senses of an adjective are associated syntactically with different nouns.

- The word grouping subsystem that I have developed also uses other types of linguistic information, such as morphology and spelling checking, which are not used in the other systems. Linguistic knowledge of this type improves the system's performance as the experiments of Chapter 5 attest.

- Finally, the architecture of the system for the grouping task allows the incorporation of multiple knowledge sources and provides for a principled combination of evidence from potentially conflicting sources. Negative examples are used during word grouping as a second knowledge source, in addition to the indirect statistical evidence of similarity. These examples of semantic unrelatedness offer an improvement in performance which can be sometimes dramatic (in the order of 40–50%) [Hatzivassiloglou and McKeown 1993]. None of the other approaches to the problem uses negative evidence or allows for the employment of multiple sources beyond the basic statistical evidence of co-occurrence.

Interest in measuring semantic similarity between words and using it to cluster them into semantic groups continues unabated, and several newer approaches [McMahon and Smith 1996; Li and Abe 1996; Lee 1997] explore methods such as minimum description length, simulated annealing, and soft clustering. A common characteristic of these efforts, as well of the earlier ones mentioned above, is their reliance on hierarchical clustering algorithms, which generally sacrifice performance for implementation simplicity and operational speed.

In closing this section, I note that the general approach adopted in the first stage of the system, i.e., the measurement of similarity between linguistic objects and their classification into groups according to this similarity metric is a quite general technique which is not limited to the task of grouping words on semantic grounds. The technique is not even limited to the corpus-based approach to language learning. For example, Batagelj *et al.* [1992] measure similarity between words from different languages in terms of their formal relatedness as strings of characters. Using the translations of 16 manually selected common words as their data, they are able to produce a hierarchical clustering of 65 languages and dialects that reflects the independent linguistic and ethnologic evidence about the similarity and classification of the languages quite well. Another example of the general applicability of clustering methods in language problems is given by the work of Iyer *et al.* [1994], who cluster text paragraphs on the basis of simple lexical similarity measures between the forms of the words as character strings, then separately estimate the parameters of an n-gram model for speech recognition for each paragraph group. The resulting mixture model offers a significantly lower error rate than the straightforward global n-gram model. Such techniques of clustering together similar text units and documents are also useful in information retrieval, both for the identification of relevant information and for its presentation to the user, and in automatic text summarization.

10.2 Work related to the further processing of semantic groups

I now turn to previous work related to the further analysis of the semantic groups formed by the first stage of the system. The goals of this analysis are to determine which of the groups are gradable or scalar, partition the scalar groups into two subscales of opposite semantic orientation, label the subscales as positive or negative, and use this information to automatically extract antonymous lexical relationships. While considerable work has been done on these problems in linguistics, the topics are largely untouched from a computational perspective. Thus, most of the work presented in this section comes from the linguistics field.

10.2.1 Identifying scalar groups on the basis of gradability

Work in gradability as a semantic phenomenon in linguistics owes much to Sapir [1944], who made many of the important observations discussed in Section 6.1. The equivalence of explicit, semi-explicit, and implicit grading (through grading adverbs, through comparative and superlative forms and constructions, and through an implied but unmentioned norm respectively) is discussed by Lyons [1977, pp. 273–275] and (for the first two types of grading) by Quirk *et al.* [1985, p. 435]. Lyons also discusses the dependence of the meaning of gradable modifiers on the norm for the property, and contrasts with the fixed interpretation of non-gradable modifiers such as color adjectives [Lyons 1977, pp. 274–275]. The strong relationship between gradability and participation in a scale has been noted by a number of researchers. Lehrer [1974] discusses the discrepancies between gradability and scalarity (i.e., the property of a word being scalar), and introduces the term *ranks* to refer to scales containing non-gradable lexemes. Bolinger [1977] discusses properties of the neutral element in scalar groups and argues for its rarity as an explicitly lexicalized item.[3] Quirk *et al.* [1985, p. 445] note that adverbs modifying adjectives usually introduce a grading effect and not an elaboration of the semantic content of the head adjective. They also note that adverbs modifying adverbs always act as intensifiers or diminishers [Quirk *et al.* 1985, p. 449].

The discussion in Sections 6.1 and 6.2 draws on the work done in linguistics, although I have added some observations and hypotheses of my own, in particular in the area of the relationship between scales and gradability and on the existence of antonymy relationships between one gradable and one non-gradable lexeme. Figure 6.1 on page 248, which lists the most important modifiers of adjectives that indicate gradability, was constructed using experimental data from the 1987 Wall Street Journal corpus and incorporates similar observations and lists of grading modifiers made by Quirk *et al.* [1985, pp. 445–448 and 589–591].

[3]This work is relevant to the issue of gradability, as well as those of separating, labeling, and ordering subscales, because it provides support to the claim that neutral elements of scales cannot be graded.

10.2.2 Dividing a scalar group according to semantic orientation

While synonymy, antonymy, and hyponymy have been extensively studied in linguistics (e.g., [Ogden 1932; Bierwisch 1969; Katz 1972; Leech 1975; Kempson 1977; Lyons 1977]), the tests proposed for determining whether one of these relations exists between two words all involve the insertion of the words in a sentential frame and the subsequent judgement of the semantic validity of the resulting sentence. Such tests can be applied by human linguists, but are not amenable to computer implementation. As a result, I use instead observable instances of lexical relations to indirectly determine whether two words have identical or different semantic orientation. Conjunctions between adjectives are the main type of evidence that I employ, in addition to the rather obvious property that two words are of opposite semantic orientation when one of them is derived from the other with the addition of an overt negative marker (e.g., *a-*, *in-*, *un-*).[4] The detailed analysis of the constraints on adjectives participating in conjunctions that was given in Section 7.3 is an original contribution of this dissertation, although it incorporates observations about the segregatory and combinatory meaning of conjunctions and about explicit markers that affect this distinction (such as *both*) from [Quirk *et al.* 1985].

Little has been done in computational linguistics on the problem of automatically classifying semantically related words as synonyms or antonyms. WordNet [Miller *et al.* 1990] includes pointers to antonyms[5] and implicitly represents synonyms in each synset, but it

---

[4] Because inflection is not always a productive compositional process, this latter property is almost always, but not universally, true; see Section 7.5 for an empirical validation.

[5] As well as to hyponyms, hyperonyms, and to words related through a part-whole relationship with a given word.

was constructed manually, with the disadvantages that such an approach inevitably has (see Sections 1.1 and 11.1).[6] Researchers have also studied associations between antonymous adjectives both through psycholinguistically motivated stimulus-response experiments [Deese 1964] and through the statistical analysis of texts [Justeson and Katz 1991]. Justeson and Katz's work is most related to the work reported here for the identification of antonyms from text, although it addresses the problem from the opposite direction: Instead of looking at text co-occurrence patterns to determine whether two adjectives are antonymous, Justeson and Katz studied the text co-occurrence patterns of known pairs of antonyms and compared them with randomly chosen pairs of adjectives. Justeson and Katz determined that antonymous adjectives occur together in the text much more frequently than random pairings of adjectives do, and this observation could be used as an indicator of antonymy if it is further refined. More specifically, Justeson and Katz did not compare the co-occurrence rate of antonymous adjectives to the co-occurrence rate of synonymous or otherwise semantically related adjectives. If a statistically significant difference exists between the rate of co-occurrence of antonymous adjectives and other cases of related adjective pairs, then a sufficiently high co-occurrence rate can be used as an alternative means of indicating antonymy; such a supplemental indicator can be incorporated into the dissimilarity model of Sections 7.5–7.6 for predicting different semantic orientation for two words.

[6] And, in fact, interest in automated approaches to identifying antonyms has been expressed to me by both Christianne Fellbaum of the original WordNet team in March 1993 and by members of the more recent EuroWordNet effort in July 1997.

10.2.3 Labeling subscales as positive or negative

I determine the semantic orientation (positive or negative) of each subscale by aggregating semantic markedness decisions which are made for each individual word in the subgroup. Markedness has received extensive treatment in the linguistics community, both at the semantic level in which I am interested and as a more general theory of semiotic opposition at various linguistics levels. Several researchers have written on the topic and proposed particular tests for determining semantic markedness. Most of these tests involve human judgements [Greenberg 1966; Lyons 1977; Waugh 1982; Lehrer 1985; Ross 1987; Lakoff 1987], and thus are not suitable for inclusion in an automatic system. Nevertheless, some of the tests proposed in the linguistics literature are automatable, either directly or through reasonable approximations; these tests are based on measurable properties of the words in isolation or in the text, such as frequency, morphological complexity, morphological productivity, and variability of different related forms. Tests of this type were described in [Jakobson 1984b; Jakobson 1962; Greenberg 1966; Zwicky 1978; Battistella 1990; Winters 1990] and were discussed in more detail in Section 8.2. I employ these tests for the purpose of determining semantic markedness and thus indirectly semantic orientation, using appropriate approximations whenever necessary to ensure that the quantities compared by these tests are measurable from a corpus and an on-line word list.

Again, little computational work on markedness, and on the automatic identification of orientation values in general, has been done. In addition to my own study of markedness in gradable adjectives [Hatzivassiloglou and McKeown 1995], Kučera [1982] has performed an analysis of the predictive power of one markedness test, text frequency, at the syntactic level, comparing the marked past tense to the unmarked present. But Kučera's analysis was exploratory in nature, aiming to verify or disprove the hypothesis that the unmarked present tense should be more common. In contrast, my work emphasizes the use of markedness indicators as predictors, to be used for the extraction of semantic orientation information from the text.

The order of words in conjunctions is another potential indicator of semantic orientation. The fact that the order of words in central coordinating conjunctions (*and* and *or*) is constrained despite the equivalence of the two places in the corresponding logical relationship has been noted in the linguistics literature, most notably in the case of personal pronouns; *you and I* is preferable to *?I and you*, although the semantic content of the two expressions is the same. Malkiel [1959] noted that this asymettry extends to some pairs of coordinated antonyms which he called *irreversible binomials*. In such pairs, there is a strong preference for the positive term to precede the negative one. Malkiel based his analysis on a small set of intuitive examples; I have generalized this approach to general pairs of antonymous adjectives, but I have not yet performed an empirical analysis verifying how frequent and accurate this predictor is. In future work, I plan to investigate this hypothesis on a large corpus and incorporate the results as an additional element to the component that labels words and subscales as positive or negative.

10.2.4 Other work related to scalar groups

A lot of the work in linguistics on the properties of scalar words is directly related to the problem of ordering semantically related terms of the same polarity according to semantic strength, as what essentially differentiates scalar words from non-scalar ones is their ability to be linearly ordered. A considerable body of work on scales has been carried out in linguistics. Initially, scales were studied in isolation and not as part of a general theory [Ayer 1952; Horn 1969; Horn 1972]. Ayer [1952], in particular, examined the effects of negation on scalar terms, which provide one of the reasons that representation of scales in the lexicon is beneficial for natural language systems. With the advent of Grice's general theory of conversational implicature [Grice 1975; Grice 1978], the properties of scales were reinterpreted in that framework and many were explained as manifestations of *scalar implicature*. The work on scalar words after Grice includes [Bolinger 1977], [Gazdar 1979], [Hubler 1983], and [Hirshberg 1985]; Levinson [1983, pp. 133–136] presents a concise and well-written summary of the theory of scalar implicature. Of the works cited above, Bolinger [1977] focused on the analysis of the neutral and extreme elements of a scale, while Hirshberg [1985] is the first (to the best of my knowledge) who considered scales from a computational perspective. However, Hirshberg's emphasis was on using and analyzing existing scalar information, not on automatically retrieving it from the text. Hirshberg, as well as Smadja [1991], note the dependence of scalar groups on the domain, a fact that further supports the need for their automatic retrieval. More recently, Elhadad [1991; 1993] has used the order relationship between scalar adjectives to appropriately select an adjective expressing the speaker's *argumentative intent* during lexical choice in text generation.

While not on the topic of identifying relative semantic strength relations, research carried out by Hearst [1992] on automatically retrieving hyponymy relationships between noun phrases from corpora is similar in its goals to the ordering problem. Hearst uses instances of a prespecified pattern of the form "NP_1, NP_2, …, *or other* NP_0" to deduce that the noun phrases NP_1, NP_2, … are hyponyms of the noun phrase NP_0. Patterns of this kind, with placeholders for particular words, can be used to retrieve semantic strength relationships, and can be complemented with criteria based on relative frequency and modification by adverbs indicating extremity (e.g., *extremely*).

10.3 Work related to evaluation

For the evaluation of the results produced by the various stages of the system I use the measures of precision, recall, and fallout that have been developed and are extensively used in information retrieval [Salton and McGill 1983; Frakes and Baeza-Yates 1992]. Since in many cases it is beneficial to supply a single measure of performance (for example, for comparing across system variants (Chapter 5) or with the performance of a random classifier and of humans who provide solutions to the same problem instance (Section 3.5.2)), and since precision and recall are often inversely related [Frakes and Baeza-Yates 1992; Chinchor 1992; Harman 1993], I have also used the F-measure which combines precision and recall in a single number. The F-measure, which was proposed by van Rijsbergen [1979] in the information retrieval context, has several interesting properties. In particular, it is larger when precision and recall are closer together, thus giving better scores to solutions that are balanced on the two types of errors. In recent major co-ordinated development and evaluation efforts, ARPA has used information retrieval metrics such as precision, recall, fallout, and overgeneration for evaluating natural language information extraction systems [Sundheim 1992; Chinchor 1992; Lewis and Tong 1992; Sundheim and Chinchor 1993; Harman 1993], and in particular has adopted the F-measure as the main measure on which to score and rank competing systems during the Message Understanding Conferences [Sundheim 1992; Chinchor 1992; Lewis and Tong 1992].

The evaluation methodology differs from that adopted in other natural language systems in several respects. First, it provides for truly "blind" evaluation of the system's results, by requiring the evaluators to provide their own solutions *independently* from the system (i.e., without access to the system's output). In this way, the burden of comparing the system's output to the correct solution(s) is shifted from the evaluators to the evaluation module of the system, and the introduction of bias is avoided. The more common approach of having the human experts grade the output of the system, while easier on the human judges, has been criticized for being prone to bias and inflation of the system's scores.

Another innovation of the adopted evaluation method is the employment of multiple models, in a way that incorporates the discrepancies between the models in the final score. Most evaluations of natural language systems employ only a single model (correct answer) for each problem instance. This simplifies the evaluation process and may be justified if this correct answer can be clearly defined[7], but this is not the case for many natural

[7]For example, if several evaluators produce identical or very similar answers for a representative sample of the problem instances, then there is no need to use multiple models, or even to prevent the evaluators from seeing the system's output.

language problems. Gale *et al.* [1992a] have used multiple models indirectly: at each decision point, the answers in the multiple models are compared and the answer that appears in the majority of the models is selected as the correct one. In this way, a single "best" or most representative model is constructed from the multiple disagreeing models, and this model is subsequently used in the evaluation, using the same techniques for comparing against a single model as before. This evaluation method was later explicitly applied in another problem by Passonneau and Litman [1993; 1997]; Hearst [1994] also drew on Gale *et al.*'s methodology in extrapolating a single answer from seven models and in computing upper and lower performance bounds for her text segmentation system. While Gale *et al.*'s method reduces the problems caused by relying on a single model, it does not allow the differences between the models to be reflected in the evaluation scores. In particular, all evaluation points are given the same importance, even though for some of them the disagreement between the models may be much higher (and thus our confidence in the prevailing answer should be lower). The evaluation method developed in this thesis addresses the above concerns by automatically weighing the importance of each decision point according to the homogeneity of the answers for it in the multiple models and by directly reflecting the degree of disagreement between the models in the produced scores.

Finally, the evaluation methodology includes the estimation of upper and lower bounds for the obtained scores. Several other researchers have previously advocated the establishment of such bounds, using an approach similar to the one described in the thesis: A measure of the performance of humans is used for the upper bound, while a simple model which is assumed to operate at random or use very limited distributional knowledge provides the lower performance bound. Work in sense disambiguation has used both of these types of bounds [Dagan *et al.* 1991; Gale *et al.* 1992a; Dagan and Itai 1994]; the paper by Dagan and Itai [1994] is notable for its detailed computation of statistical significance levels for the difference between the system's performance and the baseline performance offered by always selecting the a priori most frequent choice. Upper and lower bounds have also been estimated for the task of selecting the correct attachment of prepositional phrases [Whittemore *et al.* 1990; Hindle and Rooth 1991] and for text segmentation according to discourse subtopics [Hearst 1994], while work on the evaluation of statistical part-of-speech taggers and of parsers has provided estimates of a lower bound for these tasks [Brill 1993]. In a completely different framework which does not employ human judges at all, work on statistical language modeling frequently discusses upper bounds on performance based on established theorems in information theory, although computing the values of these bounds is itself a hard problem; see for example [Brown *et al.* 1992a] and [Brown *et al.* 1992b].

Chapter 11

Applications

Each stage of the system described in this dissertation produces a different type of lexical semantic entry for the input words. Such entries include lists of words semantically related to any given word, links between synonyms and antonyms, and semantic orientation information. This knowledge, which is domain-dependent and generally not available in precompiled form[1], can be used in a variety of applications. In the following, I first present a list of potential applications where the knowledge extracted by the system can be used, and then I discuss in more detail three potential applications from the medical domain. In all applications described in this chapter, the output of the system can be used in the form it is produced, or it can be modified by human domain experts in a post-editing step to increase its accuracy.

[1]Even with the disadvantages that such a fixed representation has; see Section 1.4.

11.1 Potential applications

The ultimate application of the system's output is the automatic compilation of domain-dependent, dynamically changing lexicons. The lexicon is the major bottleneck in current natural language systems that move beyond a very small controlled domain. Currently available lexical information is largely incomplete, and is almost never adaptable to the sublanguage of a particular domain without the application of significant manual effort. To plug this gap, many natural language systems use machine-readable dictionaries and precompiled lexical databases. WordNet [Miller *et al.* 1990] is perhaps the most well known example of a large (121,962 words and compounds, represented as 99,642 *synsets*, i.e., intersections of word denotations used to define word senses)[2] lexical database, and it has been used in several natural language systems (e.g., [Resnik 1993; Resnik and Hearst 1993; Knight and Luk 1994; Basili *et al.* 1994; Jing *et al.* 1997]). Yet, WordNet still suffers from the following disadvantages:

- All entries must be encoded by hand, which represents significant manual effort.
- Changes to lexical entries may necessitate the careful examination and potential revision of other related entries in order to maintain the consistency of the database.
- The lexical information is not specific to any domain. Rather, the entries attempt to capture what applies to the language at large, or represent specialized senses in a disjunctive manner. There is no way to encode domain-specific information; even if

[2] As of December 1997, when version 1.6 was released.

there were such a way, it would mean that a lot of manual work would have to be repeated for each domain.

- The database is necessarily incomplete.

- Time-varying information, i.e., the currency of words, compounds, and collocations, is not adjusted automatically. Thus, words that are no longer in widespread use may be in the database[3], and there is always a delay until fairly recent additions to the language get incorporated in the database.

- The validity of any particular entry depends on the assumptions made by the particular lexicographer(s) who compiled that entry. Since entries for the same word differ significantly across dictionaries (see Figure 1.1 on page 30), it is not certain that the entry will contain all the relevant information, or that the information listed will be broadly acceptable as correct.

An automated system such as the one described in this thesis for the extraction of lexical information from text corpora, in particular lexical semantic information which frequently generates the most disagreement and is more domain-dependent, does not suffer from the above disadvantages. Given the limitations of precompiled lexical databases such as WordNet, automated methods for retrieving lexical information (e.g., [Church and Hanks 1990; Klavans and Tzoukermann 1990; Smadja 1993; Smadja *et al.* 1996]) can offer considerable benefits to language systems even when these methods require limited human intervention or produce imperfect output. The output of the present system provides such

[3]This is not yet a problem for WordNet, as it was developed fairly recently, but it will be a problem for any similar project that continues for several decades.

lexical information and can populate several different fields of the semantic component of the lexicon. If such a lexical knowledge acquisition system can operate with acceptable accuracy or with minimal human supervision, the automatically produced lexicon can potentially replace human-constructed lexical databases. At the very least, such a system can supplement the information encoded by hand, either for missing words or for types of knowledge that are not part of the main database (e.g., WordNet does not currently encode semantic orientation or gradability information, or any other specialized information about scalar words).

Whether the outputs of the various stages of the system are grouped together in a comprehensive, automatically produced lexicon or represented as separate lists of relationships between words and of values for word features, the lexical semantic knowledge extracted by the system can aid in a variety of natural language problems. Such applications include:

- **Language modeling** Several researchers have argued that one promising way to address the sparse data problem during statistical language modeling is to pool together the frequency and probability estimates for semantically related words [Sadler 1989; Hindle 1990; Brown *et al.* 1992b; Pereira *et al.* 1993]. The first stage of the present system produces such semantic groups of words within a given syntactic class, and can thus be used as the basis for this *class-based smoothing* approach to model parameter estimation.

- **Sense disambiguation** It is well accepted in the natural language processing community that specific properties of a given domain predicate specialized methods and knowledge sources [Kittredge and Lehrberger 1982]. Many polysemous words tend to have a predominant sense within a specific domain. This is supported by previous empirical work that shows a significant reduction in ambiguity when a polysemous word is examined within a specific discourse [Gale *et al.* 1992c] or with respect to a particular collocation [Yarowsky 1993]. The word grouping stage of the present system implicitly and indirectly identifies this predominant sense for each word, by placing in the same group these words which are used in a semantically similar manner throughout the training corpus.[4] The members of a group, taken together, imply constraints on which senses these words have in the training corpus so that they would be compatible with each other. For example, if *note* is grouped together with *sound* or *tone*, it probably has the sense of musical sound in the training corpus; if the same word is grouped together with *coin*, it has predominantly the sense of paper money; similarly, different senses for *note* are implied if it is placed in the same semantic class as *reputation*, or in the same class as *memorandum*, etc. In this way, the produced semantic classes implicitly restrict the viable senses of the words in them. Note that these mutual constraints between word senses are bidirectional; in the example given above, not only is *note* partially disambiguated if it is placed in the same cluster as *tone* but so is the latter word. Using a secondary source of *sense* affinity information (such as WordNet), which would indicate which specific senses of each word are closely related semantically with *some* sense of the other words in the same semantic group, we can prune several of the original senses of the words. A preliminary investigation of this approach [Jing *et al.* 1997] has shown a reduction in ambiguity of up to 85%,

[4] For this application, placing each word in a unique group rather than creating soft clusters is an advantage.

including cases of full disambiguation, with an average of 56% of the original senses being pruned.

- **Verifying the semantic compatibility of modifiers** A study of medical case histories and reports has shown that doctors frequently assign multiple modifiers that are incompatible to the same technical term (e.g., they are synonyms, contradict each other, or one represents a specialized case of the other; see Section 1.7.5) [Moore 1993]. Given the technical character of these words, it is quite hard for non-specialists to edit these incorrect expressions, or even to identify such problematic cases. But the output of the grouping stage of the system, which identifies semantically related words, can be used to flag occurrences of incompatible modifiers.

- **Document labeling according to evaluative criteria** Authors of documents frequently take a positive or negative stance towards a particular topic or person that they refer to. This is expressed through the use of loaded (either positively or negatively) terms and modifiers.[5] Such modifiers, which are primarily adjectives and secondarily nouns, can be automatically identified with the techniques of Chapters 7 and 8. Then, such indicators can be aggregated over the entire document, thus classifying each document according to its evaluative orientation towards its subject. This can be performed on the fly over dynamically changing sources of documents (e.g., newsgroups and Web pages), with applications such as automated opinion polling and market research.

---

[5] For example, 1997 Nobel Peace Prize laureate Jody Williams' characterization of President Bill Clinton as a "weenie".

- **Using orientation information to interpret conjunctions** In Section 7.3, I presented constraints between the semantic orientations of two conjoined adjectives in a central conjunction and the meaning of the conjunction (combinatory or segregatory). Such constraints were used to justify the indicators employed for retrieving orientation information from the corpus. The learned semantic orientation categorization of the adjectives can also be used in the reverse direction, to help in interpreting each occurrence of the conjunctions they participate in as combinatory or segregatory (e.g., a conjunction is more likely to be segregatory if the two adjectives are of different orientations). The orientation information can also be useful in determining the scope of conjunctions, a particularly thorny problem.

- **Lexical choice in Meaning-Text Theory** Knowledge of semantic orientation, which is extracted by the third and fourth stages of the system, can be combined with corpus-based collocation extraction tools [Smadja 1993] to automatically produce entries for the *lexical functions* used in Meaning-Text Theory for text generation [Mel'čuk and Pertsov 1987]. For example, if the collocation extraction tool identifies the phrase *hearty eater* as a recurrent one, then knowing that *hearty* is a positive term enables the assignment of *hearty* to the lexical function MAGN[6], i.e., MAGN(*eater*)=*hearty*.

- **Handling argumentative force** The relative semantic strength of scalar adjectives directly correlates with the *argumentative force* of the adjectives in the text [Elhadad 1991; Elhadad 1993]. While the current version of the system does not directly com-

[6]MAGN stands for *magnify*.

pute such relative semantic strength information, it provides a first step in this direction by automatically identifying scalar groups and subscales. Semantic strength information can be used in language understanding to properly interpret the meaning of scalar words and in generation to select the appropriate word to lexicalize a semantic concept with the desired argumentative force [Elhadad 1991]. Even just the knowledge that a particular word is scalar, as determined by the second stage of the system, may help a system designate that word for additional processing.

- **Interpreting scalar implicature and negation** Scalar words obey pragmatic constraints, e.g., scalar implicature [Gazdar 1979; Levinson 1983; Hirshberg 1985] (see Section 1.7.4). When a natural language system knows that a word is scalar, it can detect (for understanding) or be aware of (for generation) the existence of such constraints. If the position of the word on the scale is known, the system can draw the implied pragmatic inferences during text analysis, or use them for appropriate lexical choice decisions during generation. In particular, such information can be used for the proper analysis and generation of expressions with explicit or implicit negation. For example, *not hot* usually means *warm* (an opposition in degree), but *not warm* usually means *cold* (an opposition in orientation).

In addition to the applications listed above that take advantage of the lexical semantic information extracted by the system, the methods discussed in this thesis can also be applied to the task of collecting syntactic and morphological information for the other components of the lexicon. This would be an application of the knowledge extraction techniques and tools developed in the context of the present system that are not specific to semantic or scalar information. For example, prepositions that introduce the complement of particular adjectives or verbs (e.g., *afraid of* but *averse to*) can be automatically identified from the corpus with only small extensions to the finite-state grammar described in the Appendix. Although this is a direction that I have not pursued in the thesis, the extension of the system to syntactic and morphological lexical knowledge can supplement other methods that have been previously proposed for such knowledge acquisition tasks at the syntactic, morphological, and phonological levels (e.g., [Atwell and Drakos 1987; Calzolari and Bindi 1990; Brent 1991; Hindle and Rooth 1991; McKee and Maloney 1992; Brent 1993; Manning 1993]). An automatically or semi-automatically constructed, domain-dependent, and dynamic lexicon could then replace the knowledge sources that are currently used to plug this gap.

The list of applications given above is not an exaustive one. The system that I have described in this dissertation can be used to dynamically extract lexical semantic information pertinent to any given domain and construct a partial lexical database, which in turn can be exploited by other specialized, domain-dependent systems. In the next section, I describe three such additional applications in the context of two medical informatics projects that are currently been developed by the Center for Medical Informatics at the Columbia Presbyterian Medical Center (CPMC) with the collaboration of our natural language group, as well as within the context of a large multimedia effort for the efficient presentation of medical patient data.

Plasma Glucose Test	
MED Code:	1601
Lab Name:	"GLUC"
Lab Code:	"CC000006"
Part of:	*Presbyterian CHEM-7*
Has Specimen:	*Plasma Specimen*
Substance Measured:	*Glucose*
Low Norm:	"50"
High Norm:	"110"
Units:	"mg/dl"

Figure 11.1: Sample MED entry for the concept *Plasma Glucose Test*. Literal string values are enclosed in quotation marks, while values that indicate pointers to other concepts in the MED database are shown in italics. Adapted from [Cimino *et al.* 1994, p. 39].

11.2 Applications from the medical domain

Extending the coverage of the MED project The MED (*M*edical *E*ntities *D*ictionary) is a controlled vocabulary of medical terms developed at Columbia University from previously available lists of medical terms [Cimino *et al.* 1994]. Each concept (term) is represented by a collection of features, which can take either string values that cannot be further analyzed (such as the name of the concept or several recognized synonyms) or pointers to other concepts in the database. In other words, the MED is a frame-based system, with the concepts forming a semantic network encoding relationships between medical terms. Figure 11.1 shows a sample entry for the concept *Plasma Glucose Test*, while Figure 11.2 shows a part of the hierarchical representation of terms in MED and a graphical representation of the links in the entry for *Plasma Glucose Test*.

The explicit representation of such relationships has several benefits, both for avoiding

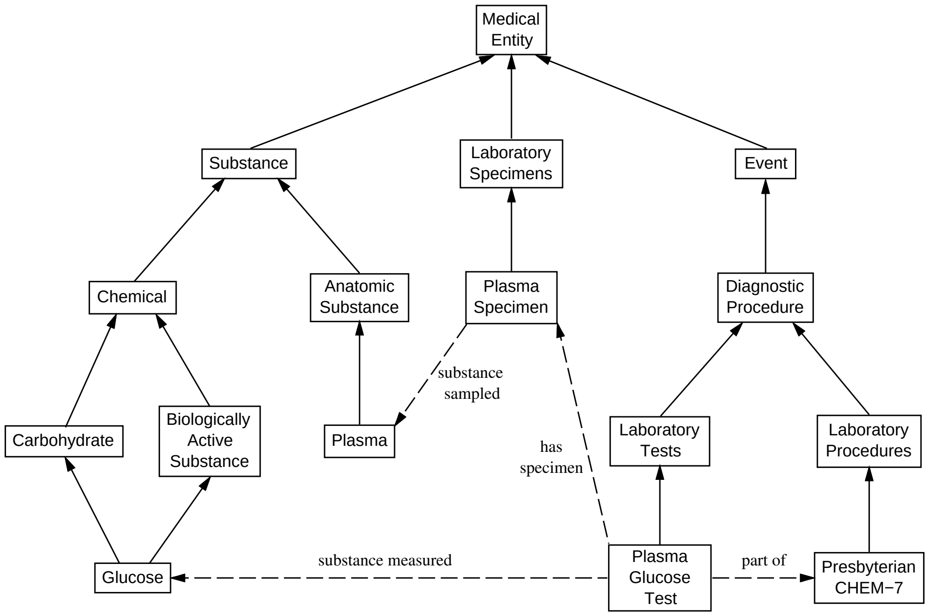


Figure 11.2: Representation of medical terms in the MED. Only a small part of the MED is depicted, including the concepts and links related to the entry for *Plasma Glucose Test* (Figure 11.1). Solid lines indicate hierarchical "is-a" relationships, while dashed lines represent non-hierarchical semantic links. The figure was reproduced from [Cimino *et al.* 1994, p. 39].

redundancy and enforcing consistency, and for facilitating the operation of automated reasoning tools that use these terms. The data in MED has been entered mostly by hand. Automatic tools have been used to support the data entry functions, e.g., by ensuring that two different concepts are not assigned the same name, but the decisions of what values to use for the various slots and which frames to link have been made by human experts, mostly based on previously compiled indices [United States National Center for Health Statistics 1980; American Society of Hospital Pharmacists 1988; National Library of Medicine 1991]. A classification program helps the human experts by suggesting equivalences between candidate terms and terms already in the database, and by alerting domain experts to potentially new concepts or semantic classes [Barrows *et al.* 1994].

The techniques for identifying semantically related words (first stage of the system) and in particular synonymous ones (third stage) can be used to augment the available mappings between terms in the MED. Such techniques can be especially useful when new terms are introduced in the database. As mentioned above, a classification program [Barrows *et al.* 1994] has already been developed for this task. This classification program operates on very different (and much shallower) principles than the ones described in this dissertation: Matches are found either on the basis of lexical similarities between the forms of the terms viewed as character strings (e.g., between *Gonorrhea* and *Acute Gonorrheal Infection of Lower Genitourinary Tract*) or on the basis of shared semantic features (which are explicitly listed for each term). Although the existence of semantic information greatly simplifies the classification task, such information is not always available (particularly for new candidate terms), and inconsistencies exist in the databases that are used to populate MED [Cimino *et al.* 1989]. Therefore, the system developed for this dissertation can provide an alternate term classification tool, handling the cases for which the simpler methods fail because of the lack of semantic data and formal derivational similarities. The need for such an automatic approach may be more acute in particular specialized subdomains, for which little semantic information has been explicitly precomputed.

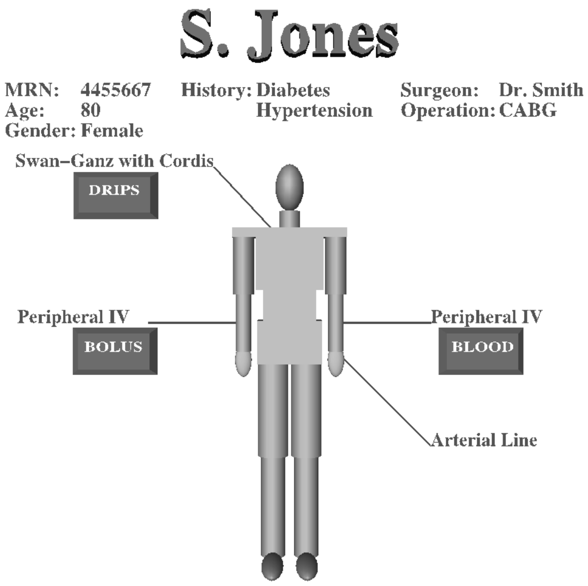
Identifying semantically similar modifiers in compound terms A second application of the techniques of the thesis to the problem of medical terminology classification is to use the grouping stage to identify classes of similar modifiers that are employed in different compound terms. This would allow for the extrapolation of the meaning of new terms on the basis of compositionality principles. This application is more in line with the rest of the thesis, where basic lexical knowledge acquisition work is emphasized. Although it is well known that compositionality cannot explain the meaning of terms in all cases [Smadja 1993], in many cases it can provide the basis for the synthesis of lexical information for new terms from the information already extracted or supplied for known terms. The results of this automated process can be subjected to a domain expert's verification before final inclusion in the lexicon; such verification is less costly than having the expert identify the terms from scratch.

Generating medical reports and presentations of medical data Another project currently under way in the field of medical informatics is the automatic generation of medical reports from data entered from a variety of health care personnel (lab technicians, nurses, primary care and admitting physicians, surgeons). Not only different sources of information contribute to the report for the same patient, but also different kinds of reports are needed for each class of users, necessitating the application of user modeling techniques. During the construction of the report, numerical indicators from possibly multiple sources such as multiple laboratory tests for the same patient need to be converted to natural language expressions. For example, the blood pressure of hypertensive patients that are prepared for surgery is assessed as poorly controlled if it is over 180 mm Hg (systolic) or over 110 mm Hg (diastolic).

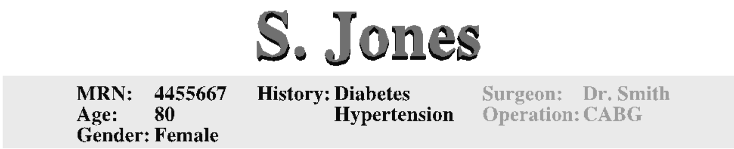
An expert system can be used to place each clinical case in the appropriate category and assign the corresponding label if the categories and their labels are known. However, not only the relevant numerical indicators change from disease to disease, but also so do the labels that are used to describe the severity of the patient's condition. For example, the terms *historical*, *quiescent*, and *active* are used for patients suffering from asthma, while the terms *subclinical*, *mild*, and *overt* are used for hypothyroid patients. The system described in this dissertation can be used on a suitable corpus of previously generated medical reports to identify groups of related descriptors of the severity of the patient's condition, forming classes such as the ones described above. Then, either a text processing tool can further analyze the reports to form links between the groups of descriptors and the actual diseases and/or relevant laboratory tests, or a medical domain expert can supply this information. In both cases, the task of constructing the lexicon for the medical report generation system is significantly simplified.

This application of the lexical knowledge acquisition capabilities of the present system fits especially well in the framework of MAGIC (*M*ultimedia *A*bstract *G*eneration for *I*ntensive *C*are) [Dalal *et al.* 1996a; Dalal *et al.* 1996b; McKeown *et al.* 1997]. MAGIC is a multimedia system that extracts information about a coronary bypass patient's record and current status and presents it using a variety of different media and modalities: graphics, text, and speech. Speech is temporally co-ordinated with graphics and with highlighting of the relevant, usually more detailed, textual information; Figure 11.3 shows a (necessarily static) snapshot of MAGIC's output. During the ongoing development of MAGIC, a large amount of time is spent in consultation with medical experts for building and refining an ontology of medical terms.[7] For example, there are subtle differences in appropriate usage between closely related terms such as the verbs "control", "support", "maintain", and

[7] Kathy McKeown, personal communication, 1996.



(a) Initial state of the presentation.



(b) *Ms. Jones is an eighty-year old, diabetic, hypertensive, female patient ...*

(c) ... *of Dr. Smith undergoing CABG.*

Figure 11.3: Sample output of the MAGIC multimedia generation and co-ordination system. The initial state of the presentation is shown in (a), while (b) and (c) indicate the changes in text highlighting (in bold font) as the accompanying speech (in italics) is produced. Adapted from [Dalal *et al.* 1996b, p. 58].

"sustain"; when a patient's blood pressure is brought to within normal limits through the administration of drugs, it is *supported* if it was too low before treatment, but *maintained* if it was previously too high. The latter term is also used if the drugs are administered in a preventive manner, while the patient's blood pressure is already within the normal range. These differences must be reflected in MAGIC's lexicon which is used for the generation of text and speech. A portion of these development costs could be saved if the techniques of this dissertation are employed to find such related terms automatically and present them to the experts only for confirmation and assignment at the proper ontology level.

Chapter 12

Current limitations and future extensions

Throughout the previous chapters I have described in detail a system that addresses a series of related problems on lexical semantic knowledge acquisition. The issues discussed in the body of this thesis certainly cover more ground than could be exhaustively addressed in a single dissertation. I have elected to examine at a reasonable depth, and propose solutions for, a significant number of the issues related to the automatic identification and further analysis of semantic and scalar word groups. I had thus to leave out some topics and questions that are peripherally related to the original lexical semantics problems, and to necessarily limit the coverage of the implemented system. In this chapter, I discuss first these limitations of the thesis, originating mostly from the amount of time available for its completion. Then, in Section 12.2, I present some additional extensions to the present work that I intend to pursue in the future.

12.1 Limitations of the current system

12.1.1 Ordering the identified subscales

Although a provision has been made in the system's architecture (cf. Figure 1.3 on page 59) for a fifth stage that analyzes the retrieved subscales (after the scalar groups have been partitioned according to semantic orientation) and orders their elements according to their relative semantic strength, this component has not yet been implemented. There seem to be few promising lexical co-occurrence properties that could help solve this subproblem with automatic, corpus-based methods. The methods that appear to be relevant to this task tend to apply to fewer words and result in a lower amount of information extracted from the corpus. Furthermore, the correlation of that information with relative semantic order remains to be experimentally verified. Thus, the final ordering subtask may be the hardest of the steps required to fully analyze linguistic scales.

Nevertheless, some clues informative of the relative position of a word in a subscale can be automatically extracted from corpora, and are worth exploring in the future. These clues, which may be sufficient for extracting at least some partial order information if not fully ordering the subscale, include certain specialized text patterns, frequency, and modification by constructs that indicate extreme values on the scale.

Some patterns involving two scalar words establish order relationships between the two words in them (i.e., that word X is semantically less strong than word Y, or, in the terminology of Section 1.7, that $X \prec Y$). Such patterns include the prototypical test pattern for scales, "*X, even Y*" (cf. Section 1.8). They also include other cases of apposition, which is a linguistic device frequently used to introduce a more specialized term; for example, the patterns "*X, although not Y*" (e.g., *She was good, although not excellent*), and "*X, in fact Y*" (e.g., *The weather was warm, in fact hot*). Some conjunctions involving *but* and negation of the second word also tend to provide information that the negated term is a semantically stonger member of the same scalar group the first word belongs to, e.g., *difficult but not impossible* and *rare but not unprecedented* (cf. Section 7.3). The frequency of such patterns in the corpus and their tendency to lead to valid inferences for the ordering task will determine, through an empirical analysis, which of them would actually be useful for the ordering substage.

In addition to the above general patterns, we can exploit supplementary methods for separating the extreme elements of a subscale (words such as *frigid* and *gigantic*) from the more typical ones (e.g., *cold* and *big*). I have identified two criteria that apply to this problem. The first is the frequency of the words in the subscale. I expect that more extreme words would tend to be less frequent than words which express values that are nearer the middle of the scale and thus more typical.

The second criterion for the identification of extreme elements is modification by a particular set of intensifiers (including, for example, *extremely*) that magnify the semantic content of the word to the largest possible degree. It is reasonable to expect that such intensifiers would be used only with a subset of the scalar words. The words that indicate medium values of the underlying property would not normally accept them because it is more economical and appropriate to use a semantically stronger element of the scale; e.g., we would not say ?*extremely warm* because it is simpler to say *hot*, or, if an extreme value is truly implied, *extremely hot*. The same constraint would apply, for a different reason, to words that are at the endpoints of a scale, as such scalar terms already imply that the property is at its extreme value; consider, for example, the phrase ?*extremely gigantic* which is unacceptable (or at least odd) because *gigantic* already pushes the notion of size to its limit. This leaves only terms that are neither too close to the middle nor too close to the endpoints of the scale as suitable candidates for modification by words that indicate extreme magnification.

Separation of the members of these three categories (words of medium, high but not extreme, and extreme semantic strength) may then be accomplished with a combination of the frequency and modification criteria. It is possible that the modification criterion will lead to a partition of each subscale into two subsets: elements that express high but not really extreme values of the property, and elements that express either medium or truly extreme values of the property. Then the frequency criterion discussed above can be used to further separate the subsets of the second type.

12.1.2 Coverage of syntactic classes

The present implementation provides for the automatic classification of adjectives and nouns according to semantic relatedness criteria, and the further analysis of adjectival groups (determining which of them are scalar, partitioning them according to orientation, and labeling the identified subscales as positive or negative). Although adjectives are the prototypical syntactic class that expresses opposition and related properties such as orientation, these concepts apply to a subset of the nouns as well. Many of the system's components, including the initial one that retrieves semantic groups, could also operate on verbs or adverbs.

The current limitation of the syntactic classes processed by the system is necessitated by its reliance on syntactic information that must first be extracted from the corpus. Unlike many statistical natural language processing systems, which treat the words as character strings with no regard to their syntactic class, I have adopted an approach that utilizes syntactic clues to improve performance. The resulting improvement can be significant, as the empirical analysis of Chapter 5 attests. Unfortunately, the downside of this approach is that additional syntactic rules need to be incorporated into the system for processing new syntactic classes (see Section A.4.2 in the Appendix).

If verbs, for example, are added to the set of classes that the system processes, the finite-state parser of Section A.4.2 will need to be extended to retrieve relationships between verbs and nouns in subject and object position, as well as between adverbs and verbs. The noun-verb syntagmatic relationships would also be useful for the semantic classification of nouns, for which only information from adjectives modifying them and other nouns in the same noun phrase is presently employed. The relationships that are expected to be useful for each syntactic class have already been analyzed (Sections 2.1 and 2.2), and the underlying mechanisms such as computation of dissimilarities, clustering, and extraction of semantic orientation information would remain largely unchanged.

Evidence that has been extracted from the corpus, whether by the application of the finite-state grammar or by simpler methods, is filtered and converted by postprocessing steps before it is actually used in the main modules of the system. These postprocessing stages, which are described in Section A.5 of the Appendix, increase the quality of the data on which subsequent methods operate by, among other steps, analyzing it morphologically and converting it to a canonical form. Such transformations increase both the expected and the actual counts of words and pairs of words, thus allowing for a higher accuracy of the statistical methods that operate on them. Currently, plural nouns are converted to singular ones and adjectives in the comparative or superlative degree are converted to their base form. An extension of the system's coverage to verbs (or the use of indicators for classifying nouns that involve verbs) would also require the conversion of inflected forms of verbs to the infinitive form.

12.1.3 Extension to new domains and non-English texts

Conversion to a common format and extraction of some basic information is necessary before the grammar is able to operate on the text. This basic preprocessing involves the identification of word and sentence breaks, and the assignment of part of speech tags to each word. Before even these tasks can be done, conversion to a standardized format in which these tools expect their input is needed.

I have implemented tools that convert the input to a standard format starting from the SGML-style representation used in the Wall Street Journal (WSJ) corpus and recognize the embedded information about sentence and paragraph breaks. Other tools perform minor editing of the raw text, to remove for example quotation marks. Finally, the pre-processed text is interfaced to the PARTS part-of-speech tagger[1] [Church 1988], and the output of this program is filtered by additional programs that correct some of the mistakes commonly made by the tagger. The tools I have implemented for these purposes are described in Sections A.3.1–A.3.4 and A.3.6 in the Appendix.

Many of these preprocessing steps, including the part-of-speech tagging phase, are performed on text that is in a standardized format, and consequently no additional work except running the appropriate programs is needed when a different corpus is used. However, the first preprocessing step is to bring the text to this standard format, and thus a new interfacing tool needs to be developed for every corpus that comes in a format not yet recognized by the system. I have already interfaced the system to the formats of the Wall Street Journal corpus and of a corpus of Associated Press newswire reports. Work is currently under way to enable the use of a corpus of medical discharge summaries, which will facilitate the application of the techniques developed in this thesis to populate the lexicon of the MAGIC system (cf. Section 11.2). But in general, porting the system to a new domain requires some adaptation to the characteristics of the corpora available in that domain. Note that this adaptation is at the lowest level, depending on the format of the data; the algorithms that process the extracted data are domain-independent.

Moving to texts from a language other than English combines the difficulties mentioned in this subsection and the previous one: We need to provide for a low-level interface to the

[1] Implemented by Ken Church at AT&T Bell Laboratories and kindly made available to us by its author.

new corpus, and also to rewrite the syntactic rules that retrieve pairs of syntagmatically associated words. The morphological analysis processes, which are used to identify formal relationships between words and to reduce words to their standard citation form, need also to be adapted. Thus, although the design of the system and the algorithms employed by it are independent of the language, adapting the front end that retrieves the data to a new language is a major undertaking.

12.1.4 Feedback between components

A limitation of the present system's design is that no feedback paths exist between different stages. Thus, each stage produces its own analysis of the data using the results from previous stages but without revising its decisions in the light of conclusions drawn or inconsistencies detected by a subsequent stage. There are several such potential negative feedback paths in the system; for example, if the second stage finds that some of the members of a group produced by the first stage are gradable and some are not, it might be possible for this information to be passed back to the first stage which would then revise that particular group. However, incorporation of such feedback paths would considerably complicate the system, since the decomposition of the problem into independent stages would be compromised.

Introduction of feedback and of revisions of earlier decisions may necessitate the use of conflict resolution techniques to arrive at a solution that is globally consistent and in maximum agreement with the suggestions of individual system modules. Conflict resolution already takes place at the word, word pair, and word group levels within each module. Initial decisions by the various algorithms are local; for example, a particular word may look gradable, or two words may look strongly semantically related or of the same orientation. To ensure the consistency of the overall solution, however, algorithms such as the majority method for gradability, or clustering for word grouping and partitioning according to orientation, are used. Such algorithms guarantee a consistent solution while optimizing the agreement with the independent local decisions.[2]

When constraints across system stages are considered, the complexity of finding an optimal or near-optimal solution increases. These problems belong in the general class of constraint satisfaction problems (CSP) [Tsang 1993], although I don't expect that a consistent solution that satisfies all constraints placed by the local decisions can be found; this is due to the imperfect local decisions, and the imperfections are in turn attributable in part to the modules' reliance on statistical methods. Nevertheless, methods developed for CSPs and related techniques such as relaxation are likely to prove helpful. Two such methods are iterative deepening search and heuristic optimization.

Iterative deepening search [Rich and Knight 1991; Russell and Norvig 1995] starts from the generally inconsistent solution produced by joining the independent decisions together and modifies it until a consistent solution is reached. Because one of the consistent solutions that involve the minimum number of modifications to (and thus the maximum agreement with) the original inconsistent solution will be located first, this method will produce an

[2]The solution is guaranteed to be optimal in some of the simpler cases, such as when the majority method is used to determine the gradability or orientation label of word groups. The solution is necessarily approximate when the problem is NP-complete, as is the case with clustering.

optimal solution.

If, however, the search method proves to be infeasible, we may have to resort to a suboptimal iterative algorithm that again starts with the inconsistent solution produced by joining the independent decisions together and repeatedly modifies it until it arrives at a consistent solution. But unlike the exhaustive search method, an objective function is used to evaluate the "degree of consistency" of each proposed solution, and modifications to the current solution are made in the direction that indicates maximum improvement. The algorithm may need to include a randomized component to limit dependence on local minima, and a backtracking mechanism to handle cases where a big initial improvement leads to a dead end.

12.1.5 Hard versus soft clustering

The first stage of the system creates word groups by placing each word in a single cluster; membership in multiple groups, which may provide a solution for the classification of polysemous words where different senses suggest different semantic groups, is not allowed. In Section 2.7.4 I described alternative soft clustering methods, and argued that such methods are not as important for lexical semantic knowledge acquisition purposes as they might look at first glance. Soft clustering methods generally sacrifice some performance since they have to estimate more parameters (in the order of the number of groups or categories being considered) from the same amount of data. Furthermore, most words occur with low frequencies and typically with only one sense; it is the few frequent words in the core portion of the vocabulary that exhibit polysemy to a large degree (see the discussion and empirical evidence presented in Section 2.7.4). From a knowledge acquisition perspective, we are most interested in the words of the former class, i.e., the more numerous, rarer, and usually unambiguous words. The usefulness of fuzzy classification methods is even less when the analysis is performed within a fixed domain, where words tend to have a predominant sense even if they are potentially polysemous in the language at large.

Despite these observations, it is sometimes beneficial to consider semantic neighbors of a word that are not its most stongly related neighbors. I described a technique that performs this task in Section 9.3 for the purpose of identifying antonymous adjectives. The method aggregates placement decisions in hard clusters to produce soft measures of word association. It could be readily applied to the general task of classifying words into groups, producing a real-valued normalized measure of group membership for each word and group according to the strength of the associations between that specific word and the members of the group.

12.1.6 Dependence of semantic groups on the context

Semantic groups and other lexical semantic attributes are currently assumed to be independent of the context of their use. While the context ultimately determines which words go into each particular group (through the co-occurrence of the words with other words in appropriate syntagmatic relationships), and similar constraints drive the other types of knowledge acquisition covered in the thesis, the formulated groups or learned values for semantic attributes are supposed to be constant, at least throughout the entire training corpus.

Yet this is not necessarily true, even for homogeneous corpora drawn from a single domain. The assumption of constancy can be violated when the extracted lexical knowledge is dependent on some other constituent of the sentence. As an example, an adjectival scale which is used to express different values for a given property can change as it is applied to different modified nouns. This may be due to collocational restrictions, or to semantic restrictions imposed by the modified term (recall the example of **hot ice-cream* of Section 1.7.1).

One possible solution to this problem is to stratify the clustering (or, more generally, any lexical semantic knowledge acquisition task) of modifiers on separate groups of modified items. The clustering techniques would first be applied on nouns to determine the noun classes, and the clustering of adjectives would take place within each such noun class. In this manner, adjectives that do not modify nouns from a particular semantic class would not be considered as candidates during the formulation of adjectival groups related to those nouns; e.g., *hot* would have been eliminated from the candidates for the temperature scale when that scale is applied to *ice-cream* and related nouns. Naturally, this approach would require a significantly larger corpus, so that enough data is provided for each stratum.

Combining the clustering of modifiers and modified items in a single step is also likely to be helpful. This approach would employ one of the *two-way* or *block clustering* methods mentioned in Section 1.11 [Good 1965; Hartigan 1972; McCormick *et al.* 1972; Hartigan

1975; Chanson and Pinson 1981; Kaufman and Plastria 1981], which utilize learned generalizations about one syntactic class (e.g., nouns) to refine the groups constructed for the other class (e.g., adjectives) and vice versa, in an iterative manner.

12.2 Other future extensions

In addition to the applications of the present work to a number of open research problems that I discussed in the preceding chapter, I have mentioned in the body of the dissertation several extensions to the already developed methods and the currently implemented system that I am interested in pursuing in the future. I summarize these extensions below.

12.2.1 Determining the optimal number of clusters

As discussed in Section 2.7.3, determining the optimal number of clusters in a partition from the data is a very hard problem in cluster analysis in general. However, experimental results from the grouping of adjectives show that humans tend to place approximately the same number of words in each group independently of the total number of words that are grouped.[3] Thus, an alternative method for finding the appropriate number of clusters is to extrapolate it on the basis of previous human-made groupings and the number of words

[3] The corresponding data for the number of clusters in human-prepared groupings of nouns exhibits greater variability than that for groupings of adjectives (see Section 4.2.2); but since I have collected human-made models for only two sets of nouns, I must regard the findings for this syntactic class as preliminary.

being grouped. This heuristic approach has served well for the partitions of words discussed in Sections 4.1 and 4.2; but it would be interesting to further analyze its performance on additional groups of words, given the psycholinguistic implications that any method that predicts the number of groups humans will use has.

I have also implemented a more general technique for the problem of selecting the correct number of clusters, based on cluster silhouettes [Rousseeuw 1987]. However, this method has produced less than satisfactory results so far. At the same time, silhouettes have proved useful as a secondary optimization criterion when only two clusters, with a reasonable cardinality, exist (cf. Section 7.6). I have argued in Section 2.7.3 that the lackluster performance of silhouettes in actual partitions of words according to semantic relatedness can be traced to the way the method treats singleton groups and the existence of numerous such groups in semantic word clusterings. The good performance of the silhouette indicator for the second task, where each of the two groups has a significant number of members (on the order of 100-500), further supports this thesis. Consequently, I plan to experiment in the future with alternative re-weighting schemes for singleton clusters, especially as this problem is not limited to computational linguistics applications.

12.2.2 Identifying semantically empty words

I have observed that words with relatively little semantic content (such as the adjective *real*) tend to move significantly from cluster to cluster as the number of clusters in the partition changes (Section 4.1.2). Such semantically "empty" words are frequently misplaced in the wrong cluster, and they can affect the placement of the other words in the partition.

Using criteria such as the variability in the semantic neighbors of a word X (the members of the same group X is placed in), we may be able to automatically identify these semantically empty words. This is interesting in its own right, and can also help increase the classification accuracy for the words with higher semantic content, by eliminating the dependence of the clustering decisions on the tenuous associations of the more informative words with the words from the former category.

12.2.3 General properties of word association measures

While considering alternative means for measuring the semantic affinity of two words on the basis of the similarity of contextual vectors corresponding to them, I have identified general properties that such measures should satisfy. For example, I have noted that cases where both words appear with some other marker word are fundamentally more significant than cases where neither of them appears with a potential marker word. This necessitates an asymmetry in the way different kinds of matches between the two corresponding vectors should be treated, and makes some of the proposed association metrics, such as information theory-based mutual information, less suitable for measuring lexical association.

These ideas were pursued further in a theoretical direction and were empirically verified in another context as well, that of the CHAMPOLLION system [Smadja *et al.* 1996, pp. 7–15], which identifies bilingual mappings between collocations. Given that measurements of lexical association lie in the core of many statistical corpus-based natural language systems, I intend to look for further theoretical and empirical justification for such general desiderata for word association metrics, and discuss appropriate metrics that satisfy these desiderata in a general context, independent of particular applications.

12.2.4 Generality of the proposed evaluation methodology

The applicability of the evaluation methodology developed in the thesis for evaluating partitions and incorporating simultaneous multiple models is not limited to the present system. The evaluation approach is generally applicable to partitions of any kind of objects and not just words. Furthermore, the extensions that provide for multiple reference models can be used for any collection of "yes"–"no" questions. Thus, the same techniques can be generally applied in information retrieval.

I would like to analyze and expand this evaluation methodology further, with the goal of using it for other tasks as well. One such extension is to carry out a formal evaluation of the proposed evaluation mechanism itself, and in particular to quantitatively compare it to previous alternatives such as the majority method [Gale *et al.* 1992a], discussed in Sections 3.4 and 10.3. This comparative analysis may be performed in the context of an actual system (e.g., the first stage of the system described in this thesis, or an information retrieval system that relies on the basic concepts of precision and recall) or through simulation under prespecified distributional assumptions for the system's responses and the models' specifications.

12.2.5 An alternative evaluation method

In Section 3.5.1 I discussed constraints that the structure of a partition and the presence of multiple evaluation models place on the evaluation metrics. The overall effect of these constraints is the introduction of non-linearities in the behavior of the evaluation measures, i.e., small changes in a grouping of words may result in a large change in the assigned score.

An alternative evaluation method that does not suffer from this non-linearity disadvantage is based on an explicit definition of some notion of distance between the system's output and the optimal solution that could have been produced given the models that have been constructed by the human experts. However, there are several subproblems that need to be addressed before we can actually employ this alternative evaluation method:

- We need to precisely define the distance metric that will be used to measure discrepancies from the optimal solution (or any other solution that is used as a reference point; for example, we may want to measure discrepancies between any two models constructed by humans). One way to measure discrepancy is to find the minimum number of elementary operations that should be applied to the evaluated solution to produce the reference solution. In this case, we need to define exactly what the "elementary" operations are. When two partitions are compared, this is far from clear. Considering the example of Figure 3.2 on page 155, should we treat the transformation from the left side to the right side as the result of the application of one operator (splitting cluster 3 into two clusters) or as the result of the application of two operators (moving elements H and I out of cluster 3)? Both approaches have problems: If we allow a multi-element move, we equate transformations that involve few elements (small clusters) to transformations that involve many elements. And if we disallow operators that work on the cluster level, we treat an operation such as moving three related elements from their current cluster to another cluster the same as moving these three elements to three different clusters, i.e., the original solution does not receive any credit for having placed the three related elements in the same group, even if that particular group was inappropriate.

- Even after a proper distance metric between solutions has been defined, we need a method to compute it. If the minimum number of necessary transformations is used as the metric, no analytic method for its computation seems apparent. Rather, it seems that we need to resort to applying an approximate hill-climbing (gradient-descent) method that will start from the solution that is evaluated and attempt various transformations until it reaches the target (reference) solution.

- Finally, it is not clear how the optimal solution will be found. One approach that has been outlined in Section 3.5.2 is to use the clustering method of Section 2.7.1 (or, in general, the solution-constructing method employed by the system) to find such an optimal solution by making available to the system all the human-constructed models that will be used during evaluation. In this way, the corpus-extracted information that would be normally used for the construction of the solution (e.g., word dissimilarities) is replaced with the "correct" values. This approach, however, suffers from the disadvantage that the system is not guaranteed to find the optimal solution, even when randomization techniques are used.[4] The problem is exacerbated when more than one optimal solution exists for the same problem instance.

Given these difficulties, the implementation of this alternative evaluation method is an open research problem. It would, however, be interesting to compare the results produced by it to the scores assigned by the information retrieval-based approach of Sections 3.2–3.4.

12.2.6 Estimating an upper bound on system performance

The adopted methodology of Section 3.5.2 provides an upper bound on the performance of the system on a given task by averaging the performance of the human-constructed solutions (models) when the latter are evaluated against all other such models. Obviously, at least one of the n human-constructed models will achieve the bound when evaluated against the remaining $n-1$ models; but there is no guarantee that a single solution that achieves the average performance against all n models is in principle constructible.

An extension of the present evaluation methodology is to find the optimal single solution given the models, calculating the best score that *any* single-partition solution could ever achieve. This maximum score, which provides a true upper bound on system (or human) performance, can be computed on the basis of the agreements between the reference models

[4] We can increase the probability of the system's finding the optimal solution to any number arbitrarily close to 1, but we can never be absolutely sure. Furthermore, it is hard to even know what this probability is for a given number of random repetitions of the search procedure.

at each decision point.[5] Because of the structural constraints that partitions impose, it may not be always possible to obtain this score by a closed formula; then, an approximation algorithm that uses the same clustering techniques as the actual system but has access to the reference models can search the solution space for the optimal assignment of words to groups.

12.2.7 Extending the analysis of the significance of linguistic knowledge

In Chapter 5 I measured the performance improvement that linguistic knowledge offers in the present system, and concluded that this improvement is significant both statistically and quantitatively. I also argued that these results should carry over to other statistical natural language learning systems which share common elements with the system described in the thesis.

To further substantiate this claim, an extension of the present work would be to apply the comparative evaluation methodology of Chapter 5 to an entirely different natural language processing problem, such as sense disambiguation, collocation extraction, or machine translation. This would require either re-implementing or having access to apropriately modified versions of sense disambiguation, collocation extraction, etc., systems.

In the thesis, I have looked into several parameters that affect the quality of the produced

[5] Unless there is a single model, or perfect agreement between the models, the maximum score will be less than 100%. In decision points where some dissent exists, any choice would be penalized, as explained in Section 3.5.1.

solution. In addition, interactions between several of these parameters can be studied, especially as other system parameters vary in a controlled manner. For example, it would be interesting to measure the benefit that linguistic knowledge provides as the training corpus size increases, and the relative importance of each system module for more or less ambiguous or frequent words. This would require the stratification of the currently available experimental data and its augmentation with observations at additional points of the parameter space (e.g., finer distinctions for the corpus size parameter), so that trends in the behavior of the system can be established. Preliminary evidence indicates, for example, that the significance of the data extraction method (finite-state grammar versus simpler methods) remains high as the the corpus data increases, and the accumulation of more training material does not appear to completely offset the advantage of the more linguistically informed version of the clustering system.

12.2.8 Fractional values of semantic orientation

The algorithms of Chapters 7 and 8 currently classify an adjective as either of positive or negative semantic orientation, without any gradation of the result. Yet, the evidence might be conflicting, and certainly is not equally strong in all cases. Cluster silhouettes [Rousseeuw 1987], which are already computed by the system to aid in the placement of the words in the two potential classes, can be used (after normalization) as a "goodness-of-fit" measure for each word. This measure ranks the words according to how well they fit in their group, and can thus be used as a *quantitative* measure of orientation, refining the binary *positive–negative* distinction.

Not only would this extension provide more informational output, it would also be useful as a means of increasing the precision of the system, at the cost of sacrificing some coverage. Assuming that the system is better in classifying words that fit better in their assigned groups (a reasonable assumption that needs nevertheless to be empirically verified), the accuracy of the system would increase if we restricted the labeling decisions to words with high values of this measure and refrained from judging the orientation of the remaining words.

A related area of the thesis where precision could be significantly increased at an expected modest cost in coverage is the automatic identification of antonyms. As noted in Section 9.3, the system typically either locates the correct antonym for an adjective near the top of the list of related words for that adjective or misses it completely (i.e., places it near the end of the list). Empirical evidence for this claim has already been provided in Table 9.4 on page 355. Thus, we could extract a variable number of candidate antonyms for each adjective, using the heuristics discussed in Section 9.3 (which are based on real-valued lexical association measures), and keep precision high without severely impairing recall.

Antonym identification would also benefit from fractional values for gradability or semantic orientation. If such values were available, the system could apply the constraints that gradability and orientation impose only in the cases where the values were sufficiently high, i.e., above an appropriate prespecified threshold. Then the risky operation of eliminating some of the candidate antonyms for each word would be performed only for the words for which the system's confidence in these constraints would be high. Gradability assignments already receive fractional values, as they are produced from a log-linear model (see Section 6.3), but the system does not currently take advantage of this. Orientation decisions could be extended to fractional values using the already computed silhouette coefficients with a suitable normalization, as noted earlier.

12.2.9 Markedness and orientation values for polysemous words

In Chapter 8 I analyzed the semantic markedness values of 449 distinct adjectives participating in 344 pairs expressing oppositions. Many adjectives participated in more than one pair, and frequently this was due to a change (sometimes dramatic) in the sense of the adjective. For example, *deep* participated in six oppositions with different senses. These senses are indicated by the opposing adjective and the sample modified noun listed below: *deep/shallow river*, *deep/short kick* (in sports), *deep/light sleep*, *deep/pale color*, *deep/high sound*, and *deep/superficial cut*.

While analyzing the markedness values assigned to these polysemous adjectives, I noted that in most cases an adjective tended to have the same markedness value (either marked or unmarked) throughout its different senses, regardless of the other member of the opposition. In the example given above, *deep* is the unmarked term in five of its six senses, with one of the cases (the sound sense) having both adjectives marked.[6] Although I have not performed

[6]Note that the other member of the opposition in this exception, *high*, also tends to be unmarked in its other senses. Thus a conflict between the preferences of the two words exists, which shifts their status in

a formal statistical analysis, this property of markedness consistency across senses appeared to be statistically significant. Thus, semantic markedness (and consequently semantic orientation) apparently remains constant across different senses of the same word, in the same way that syntactic features typically do and unlike other lexical semantic features. I plan to formally validate this observation through a more detailed study in the future.

12.2.10 Additional indicators of semantic orientation and antonymy

After words have been separated into groups of different semantic orientation, the current implementation relies on semantic markedness indicators to pick the group that contains the positive terms. This essentially boils down to the relative frequencies in an opposition pair of words or word groups (cf. Section 8.10).

An additional indicator that could be useful for the determination of positive or negative orientation is positional information in conjunctions. It has been noted that, at least when antonyms are conjoined with a central coordinator (*and* or *or*), there is a tendency for the positive term to appear first. For example, we say *good and bad*, *strong and weak*, *rich and poor*, and not the other way around. Malkiel [1959] noted this apparent preference for the positive term to precede the negative one in such constructs which he called *irreversible binomials*, but he made his observations only on a limited set of intuitive examples. I intend to quantitatively explore the extent of this phenomenon in observed examples from the

---

that particular opposition to marked for both of them.

corpus, and, if this tendency of the positive term to be first is verified from the experimental data, to include this indicator as one of the criteria for determining semantic orientation.

Similarly, Justeson and Katz [1991] have empirically established that antonymous adjectives appear together in the same sentence much more frequently than random pairs of adjectives. If this higher co-occurence rate holds when antonyms are contrasted with other kinds of semantically related words, it can provide a direct indicator of antonymy and complement the indirect indicators (such as different semantic orientation) of Chapter 7.

12.2.11 Psycholinguistic implications of human performance

The analysis of the performance of humans during the construction of models for evaluation purposes reveals several interesting observations of a psycholinguistic nature. These include estimates of the difficulty humans have with the task of constructing model solutions for the various subproblems in the thesis and the degree of agreement among humans. It is especially illuminating to study measurements of task difficulty and inter-evaluator agreement as they relate to observable parameters of the problem instance such as the size of the test set of words that are clustered and the frequency and specificity of its members. For example, humans easily produce a solution for the semantic clustering subproblem when the size of the collection of words remains small; but for sets containing more than 50 words, the humans need significantly more time and actually more time than the automated system takes for the same problem instance. Clearly the relationship between difficulty for the humans as measured by the time needed for the completion of the task and the size of the problem instance is superlinear (and possibly not polynomial).

The difficulty is also influenced by the specificity of the words that are being grouped, with the evaluators finding it harder to process polysemous or ambiguous words. As Table 4.6 on page 190 indicates, similar observations hold for inter-evaluator agreement measures, which tend to fall as the task becomes inherently more difficult (i.e., more words are clustered) but are higher when the grouped words are unambiguous.

Another observation with psycholinguistic significance is the emergence of a near constant ratio of the total number of words clustered to the number of groups used as humans partitioned different collections of words. This has enabled the use of a heuristic to predict the appropriate number of clusters for each problem instance (cf. Section 2.7.3). It also implies certain restrictions on what levels of granularity for word classifications (as measured via the number of words in each class) the human evaluators consider appropriate.

Finally, the distributional characteristics of positive and negative terms in the language also have psycholinguistic implications. In Section 8.10 I analyzed a large, representative set of 1,336 adjectives according to their semantic orientation. I noted that slightly more than half of those had a negative orientation, and yet the positive terms were significantly more common in the corpus. In other words, people have more negative than positive expressions at their disposal, but they prefer to use the positive expressions much more than the negative ones.

I intend to investigate these preliminary observations more formally in the future, using a larger collection of human-constructed models and for additional combinations of problem instance parameters such as test set size, word frequency, and word specificity. The collected data can then be used to statistically test the hypotheses discussed above.

Chapter 13

Contributions

I have presented an implemented system which proposes solutions for a number of related difficult problems from lexical semantics in a unified way. The system is also the first that addresses most of these problems from a computational perspective. In the introductory chapter I explained why the automatic acquisition of lexical semantic knowledge is a particularly hard problem and why the use of corpus-based methods offers significant advantages (Sections 1.3 and 1.4). The present system provides solutions to problems of this type, compiling semantic entries for the input words without human intervention. The knowledge extraction methods do not depend on particular theories of semantics, but rather are based on empirical evidence. As a result, the system is both fully portable across domains[1], and utilizes any domain-specific information present in the input corpus, automatically adapting to the domain sublanguage.

The system offers solutions to several problems that have not been addressed before with computational methods, or have been addressed in a more superficial, simplified, and

[1]Only a front-end component needs to be adapted to the format of a new corpus; all the internal methods are independent of the domain.

ad hoc manner (see Section 10.2); these include the automatic identification of semantic and scalar groups, antonyms, and gradability, semantic markedness, and semantic orientation information. The system also addresses the problem of separating words into groups of semantically related ones with more powerful, detailed, and linguistically justified methods than those used by other systems that have been independently developed for this task (Section 10.1).

The system's flexible architecture enables appropriate statistical inference modules to operate on corpus data extracted according to various linguistic sources and with the guidance of shallow linguistic knowledge encoded in the system. During the system's design, the following two broad goals were set and are reflected in the adopted architecture:

- **Modularity**. Each component of the system communicates with other components through a well-specified interface. Thus, any component can be modified internally (for example, by changing the similarity metric used or by adopting a different clustering technique) without affecting other modules of the system.
- **Extensibility**. The architecture for each of the stages of the system allows the future incorporation of additional knowledge sources with minimal effort. Only one module in each stage, the one which combines the results of individual knowledge sources, needs to be changed if new knowledge sources are added in the system.

The end result of the system's output is also important, as it can be used to automatically enrich a lexicon with semantic features. The lack of such information is a major bottleneck in current natural language processing systems, at least when deployment in practical situations and real-world domains is sought. Several potential applications of the system's output have been described in Chapter 11. In particular, the availability of an automatic system for the extension of medical terminologies or the identification of disease-dependent modifying expressions, both of which are currently being done by hand, is a useful application in the context of medical informatics.

The technical contributions of the present work can be summarized in the following five areas:

Novel applications of statistical methods Sophisticated statistical methods are employed in the system, including a robust, non-parametric self-smoothing method for the estimation of the semantic similarity between words. This method has not been used in computational linguistics applications before. Theoretical properties of the measures that are appropriate for the estimation of lexical affinity have been identified and currently popular association measures are evaluated in light of these observations. Statistical modeling of the data, such as with generalized linear models, is employed to assess the significance of competing knowledge sources and combine them. This is in contrast to the usual practice of using either ad hoc methods or arbitrarily chosen weights in a linear sum formula whenever information from more than one source is used.

Algorithms and knowledge sources for extracting new kinds of lexical semantic information The dissertation discusses several types of lexical semantic knowledge applicable to semantic and scalar word groups, such as gradability, antonymy, semantic markedness, and semantic orientation. For many of these, no previous attempt at automatically extracting the relevant information from corpus data has been made; for the subproblems for which previous attempts exist, such earlier approaches are usually limited to direct matching of text patterns that cover only a fraction of the cases we are interested in. The current work has explored several deeper linguistic sources that can still be efficiently extracted from the corpus, and has analyzed empirically several potential knowledge sources that have not been proposed before (for example, negative examples for the word grouping task or conjunctions between adjectives for the identification of synonyms and antonyms) in order to verify which of the potential indicators are in practice useful for lexical knowledge acquisition. The validity of most of these knowledge sources has not been analyzed in a computational context before.

Use of shallow linguistic knowledge in addition to the statistics Unlike most statistical natural language systems which place the emphasis on the statistical component only and often neglect to exploit as much linguistic knowledge as possible, the present system incorporates several linguistic components. These components process the corpus data and analyze it in order to maximize the information that can be gained from it and minimize the number of spurious or erroneous relationship instances that are extracted. To this end, several preprocessing steps are applied to the corpus, and a detailed finite-state grammar is used for data extraction instead of the simpler methods typically employed in other systems. Shallow linguistic knowledge is used not only in the grammar rules but also in several post-processing steps that filter and transform the retrieved data items, resulting in higher accuracy without sacrificing parts of the corpus that are difficult to analyze.

Quantitative assessment of the relative importance of linguistic knowledge sources The thesis also addresses the issue of quantitatively measuring the relative significance of shallow linguistic knowledge. This is a topic of considerable interest, as it is expected that a combination of the knowledge-based and statistical paradigms will lead to improved performance, but this assertion is usually based on intuition and not on empirical evidence. Extensive experiments in the context of the system have been performed and the results have been analyzed in a statistical framework. In this way, hard quantitative evidence has been collected, showing the improved performance of the hybrid approach over statistics alone. The analysis also reveals which components of linguistic knowledge are responsible for most of the observed increase in performance and thus should be given the most weight during development.

Evaluation and meta-evaluation In addition to the core issues of the system's design, the thesis includes a detailed discussion of how the output of the system is evaluated using simultaneously multiple reference models constructed by human experts. Issues such as why the evaluation methodology employed in the system is preferable to the more common method of scoring the system's results *a posteriori* are covered. More importantly, the evaluation methodology includes three significant new elements, which are motivated in a general setting and mathematically formalized. First, a general method is developed for comparing complex structured results such as set partitions by reducing them to parallel lists of binary decisions. Second, multiple reference models are introduced into the evaluation methodology by extending commonly used metrics from information retrieval. Finally, the importance of meta-evaluation is discussed at length, and general methods for computing upper and lower limits on the system's potential performance are discussed.

The dissertation also contributes to computational linguistics at large, as well as related fields, such as non-computational linguistics, psycholinguistics, and information retrieval. For both computational and non-computational linguistics, it provides evidence on the power (and some of the limitations) of corpus-based methods for the automated learning of lexical semantic information. As a result of a deliberate design decision, the system uses a minimal amount of lexical knowledge, none of it specialized, domain-dependent, directly related to the words being analyzed, or semantic in nature. The material that the system can access in order to learn (parts of) the meaning of the words being analyzed is limited to the corpus. Therefore, the system provides a test case for the hypothesis that semantic knowledge, including fine distinctions such as difference in semantic orientation, can be automatically learned from large amounts of free text. The system performs quite well, thus validating this hypothesis. Particular types of consistent failures (or of unexpected successes) can be further analyzed, and the results can be potentially generalized to other applications, providing a basis for a better understanding of what corpus-based systems can and cannot do.

The general learning, estimation, validation, and evaluation mechanisms employed in the present work are not limited to the specific problems addressed in the thesis; rather, they can be transferred to many other computational approaches on other natural language problems. The empirical validation studies, which analyze quantitative measurements related to linguistic phenomena on the basis of indicators automatically extracted from the corpus, are of particular significance to theoretical linguistics. Frequently, these analyses essentially test the validity of hypotheses that have been formulated in the linguistics literature. These hypotheses are based on and supported by particular linguistic theories of semantics[2], but have not been empirically verified. Consequently, the results of the exploratory analyses that determine the usefulness of indicators for linguistic properties also offer currently unavailable evidence relevant to pure linguistic problems. For example, during the analysis of tests that have been proposed in the linguistics literature for the determination of semantic markedness in pairs of antonymous gradable adjectives (Sections 8.2–8.9), it was found that one of the simplest tests, text frequency, outperforms other more complicated tests; that frequency contains essentially all the information extractable from the other tests; and that some of the proposed tests cannot be distinguished from a random classifier. These surprising results are certainly of interest to linguists who study semantic markedness. Even when the analysis is not directly testing hypotheses proposed in linguistics, frequently the results offer insight into complex linguistic phenomena (e.g., antonymy). Furthermore, this approach represents a different methodology paradigm than the one traditionally used in linguistics, where evaluation of a proposed theory on a small set of convenient examples is common. I believe that the use of this methodology in linguistics helps satisfy the criterion of *falsifiability* [Popper 1965; Lakatos 1972; Kempson 1977], and thus moves the field towards a more scientific basis.

The evaluation methodology that was presented in Chapter 3 is also not limited to computational linguistic problems. Formal methods operating on generic, abstract objects

[2]And, not infrequently, vehemently rejected by competing linguistic theories.

are proposed in the thesis for this task, and the subproblems addressed include how to evaluate structured collections of objects such as set partitions, how to introduce multiple reference models, and how to assess the relative meaning of a given evaluation score. These problems are of significant interest in information retrieval, and the methods developed in the thesis can be directly applied to tasks from that field.

Finally, results obtained as a by-product of the evaluation process include data on how much disagreement exists between solutions proposed by different humans and how difficult it is for each evaluator to produce a solution. Relationships between these results and parameters of the problem instance such as the size of the word set that is partitioned and the frequency or specificity of the words in that set can then be projected; several observations of this type have already been noted in the thesis in connection with the system's output for particular word sets (Sections 2.7.3 and 4.1.4). For example, I noted that humans find more difficult to group polysemous adjectives than less ambiguous ones, that proportionately less agreement is reached on the more frequent and polysemous words, and that the time the humans need for grouping words increases with the size of the word set faster than the system's corresponding time does. In this way, the current research offers several types of data about the performance of humans on the tasks that the system addresses. Since the data on human performance can lead to several hypotheses explaining the experimental evidence, the dissertation contributes indirectly to a study of linguistic performance rather than competence [Chomsky 1965], and is thus relevant to the field of psycholinguistics.

I have summarized above the contributions of this work to related fields outside computational linguistics, such as traditional linguistics, information retrieval, and psycholinguistics. These contributions, as well as the many potential extensions to this work that I outlined in the previous chapter, indicate the richness of the underlying topics, and their relevance to multiple areas of natural language understanding, modeling, generation, and evaluation. I have examined many of the topics in considerable depth, developed new methodologies and techniques, provided implemented solutions, formally evaluated their results, and investigated alternatives. Nevertheless, the topics certainly merit further exploration from different perspectives, and I intend to pursue some of these directions in the future.

Bibliography

[Aarts and Korst 1989] Emile H. L. Aarts and Jan Korst. *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing.* Wiley, New York, 1989.

[Abney 1991] Steven P. 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, Dordrecht, The Netherlands, and Boston, Massachusetts, 1991.

[Abraços and Lopes 1997] José Abraços and Gabriel Pereira Lopes. Statistical Methods for Retrieving Most Significant Paragraphs in Newspaper Articles. In *Proceedings of the Workshop on Intelligent Scalable Text Summarization*, pages 51–57, Madrid, Spain, August 1997. Association for Computational Linguistics.

[Adriaens 1992] G. Adriaens. From COGRAM to ALCOGRAM: Towards a Controlled English Grammar Checker. In Christian Boitet, editor, *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 595–601, Nantes, France, August 1992.

[Aho *et al.* 1988] Alfred V. Aho, Brian W. Kernighan, and Peter J. Weinberger. *The AWK Programming Language.* Addison-Wesley, Reading, Massachusetts, 1988.

[Aït-Mokhtar and Chanod 1997] Salah Aït-Mokhtar and Jean-Pierre Chanod. Incremental Finite-State Parsing. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing (ANLP-97)*, pages 72–79, Washington, D.C., April 1997.

[American National Standards Institute 1986] American National Standards Institute. *American Standard Code for Information Interchange*, 1986. ANSI X3.4-1986.

[American Society of Hospital Pharmacists 1988] American Society of Hospital Pharmacists, Bethesda, Maryland. *American Hospital Formulatory Service Drug Information*, 1988.

[Anderberg 1973] Michael R. Anderberg. *Cluster Analysis for Applications.* Academic Press, New York, 1973. Revised version of the author's thesis, University of Texas at Austin, 1971.

[Anscombre and Ducrot 1983] Jean-Claude Anscombre and Oswald Ducrot. *L' Argumentation dans la Langue.* Philosophie et Langage. Pierre Mardaga, Brussels, Belgium, 1983.

[Atkins *et al.* 1992] B. H. Atkins, J. Clear, and Nicholas Ostler. Corpus Design Criteria. *Literary and Linguistic Computing*, **7**:1–16, 1992.

[Atwell and Drakos 1987] Eric S. Atwell and Nikos F. Drakos. Pattern Recognition Applied to the Acquisition of a Grammatical Classification System from Unrestricted English Text. In Bente Maegaard, editor, *Proceedings of the Third European Conference of the ACL*, pages 56–62, Copenhagen, Denmark, April 1987. Association for Computational Linguistics.

[Ayer 1952] Alfred J. Ayer. Negation. *The Journal of Philosophy*, **49**(26):797–815, 1952.

[Bache and Davidsen-Nielsen 1997] Carl Bache and Niels Davidsen-Nielsen. *Mastering English: An Advanced Grammar for Non-native and Native Speakers.* Mouton de Gruyer, Berlin and New York, 1997.

[Bahl *et al.* 1977] Lalit R. Bahl, J. K. Baker, Fred Jelinek, and Robert L. Mercer. Perplexity—A Measure of the Difficulty of Speech Recognition Tasks. In *Program of the 94th Meeting of the Acoustical Society of America*, page S63, Miami Beach, Florida, December 1977. American Institute of Physics, New York. Supplement no. 1 to the *Journal of the Acoustical Society of America*, volume **62**(S), Fall 1977.

[Baker and Nelder 1978] R. J. Baker and John A. Nelder. *The* GLIM *System, Release 3: Generalized Linear Interactive Modeling.* Numerical Algorithms Group, Oxford, 1978.

[Baker 1995] Carl L. Baker. *English Syntax.* MIT Press, Cambridge, Massachusetts, 2nd edition, 1995.

[Bar-Hillel and Carnap 1952] Yehoshua Bar-Hillel and Rudolf Carnap. An Outline of a Theory of Semantic Information. Technical Report 257, MIT Research Laboratory of Electronics, Cambridge, Massachusetts, 1952. Reprinted in [Bar-Hillel 1964, pp. 221–274].

[Bar-Hillel 1964] Yehoshua Bar-Hillel. *Language and Information.* Addison-Wesley, Reading, Massachusetts, 1964.

[Barrows *et al.* 1994] Randolph C. Barrows Jr., James J. Cimino, and Paul D. Clayton. Mapping Clinically Useful Terminology to a Controlled Medical Vocabulary. In *Proceedings of AMIA-94*, pages 211–215. American Medical Informatics Association, 1994.

[Barwise and Perry 1983] Jon Barwise and John Perry. *Situations and Attitudes.* MIT Press, Cambridge, Massachusetts, 1983.

[Basili *et al.* 1994] Roberto Basili, Maria Teresa Pazienza, and Paola Velardi. The Noisy Channel and the Braying Donkey. In *Proceedings of the ACL Workshop The Balancing Act: Combining Symbolic and Statistical Approaches to Language*, pages 21–28, Las Cruces, New Mexico, July 1994. Association for Computational Linguistics.

[Batagelj *et al.* 1992] Vladimir Batagelj, Tomaž Pisanski, and Damijana Keržič. Automatic Clustering of Languages. *Computational Linguistics*, **18**(3):339–352, 1992. Technical correspondence.

[Bates and Watts 1988] Douglas M. Bates and Donald G. Watts. *Nonlinear Regression Analysis and its Applications.* Wiley, New York, 1988.

[Battistella 1990] Edwin L. Battistella. *Markedness: The Evaluative Superstructure of Language.* State University of New York Press, Albany, New York, 1990.

[Bear *et al.* 1992] John Bear, John Dowding, and Elizabeth Shriberg. Integrating Multiple Knowledge Sources for Detection and Correction of Repairs in Human-Computer Dialog. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 56–63, Newark, Delaware, June 1992. Association for Computational Linguistics.

[Berger *et al.* 1994] Adam L. Berger, Peter F. Brown, Stephen A. della Pietra, Vincent J. della Pietra, John R. Gillet, John D. Lafferty, Robert L. Mercer, Harry Printz, and Luboš Ureš. The Candide System for Machine Translation. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 157–162, Plainsboro, New Jersey, March 1994. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Bezdek 1973] James C. Bezdek. Cluster Validity with Fuzzy Sets. *Journal of Cybernetics*, **3**(3):58–72, July–September 1973.

[Bierwisch 1969] Manfred Bierwisch. On Certain Problems of Semantic Representation. *Foundations of Language*, **5**(2):153–184, May 1969.

[Black *et al.* 1991] Ezra Black, Steven Abney, D. Flickenger, Ralph Grishman, P. Harrison, Donald Hindle, R. Ingria, Fred Jelinek, Judith Klavans, Mark Liberman, Mitchell Marcus, Salim Roukos, B. Santorini, and Tomek Strzalkowski. A Procedure for Quantitatively Comparing the Syntactic Coverage of English Grammars. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 306–311, Pacific Grove, California, 1991. Morgan Kaufmann, San Mateo, California.

[Black *et al.* 1993] Ezra Black, Fred Jelinek, John Lafferty, David M. Magerman, Robert L. Mercer, and Salim Roukos. Towards History-Based Grammars: Using Richer Models for Probabilistic Parsing. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 31–37, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Bock 1974] Hans H. Bock. *Automatische Klassifikation.* Studia Mathematica. Vandenhoek und Ruprecht, Göttingen, Germany, 1974.

[Bolinger 1977] Dwight Bolinger. *Neutrality, Norm, and Bias.* Indiana University Linguistics Club, Bloomington, Indiana, 1977.

[Boucher and Osgood 1969] T. Boucher and Charles E. Osgood. The Polyanna Hypothesis. *Journal of Verbal Learning and Verbal Behavior*, **8**:1–8, 1969.

[Bourbeau *et al.* 1990] Laurent Bourbeau, Denis Carcagno, E. Goldberg, Richard Kittredge, and Alain Polguère. Bilingual Generation of Weather Forecasts in an Operations Environment. In Hans Karlgren, editor, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, volume 3, pages 318–320, Helsinki, Finland, 1990.

[Bourigault 1992] D. Bourigault. Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases. In Christian Boitet, editor, *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 977–981, Nantes, France, August 1992.

[Boyer and Lapalme 1985] M. Boyer and Guy Lapalme. Generating Paraphrases from Meaning-Text Semantic Networks. *Computational Intelligence*, **1**(3 and 4):103–117, August–November 1985.

[Breiman *et al.* 1984] Leo Breiman, J. H. Friedman, R. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth International Group, Belmont, California, 1984.

[Brent 1991] Michael R. Brent. Automatic Acquisition of Subcategorization Frames from Untagged Free-text Corpora. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 209–214, Berkeley, California, June 1991. Association for Computational Linguistics.

[Brent 1993] Michael R. Brent. From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. *Computational Linguistics*, **19**(2):243–262, June 1993.

[Brill 1993] Eric Brill. Automatic Grammar Induction and Parsing Free Text: A Transformation-Based Approach. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 237–242, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Briscoe and Waegner 1992] Ted Briscoe and Nick Waegner. Robust Stochastic Parsing Using the Inside-Outside Algorithm. In *Proceedings of the AAAI-92 Workshop on Statistically-Based NLP Techniques*, pages 39–47, San Jose, California, July 1992. American Association for Artificial Intelligence.

[Brown *et al.* 1990] Peter F. Brown, John Cocke, Stephen A. della Pietra, Vincent J. della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roosin. A Statistical Approach to Machine Translation. *Computational Linguistics*, **16**(2):79–85, June 1990.

[Brown *et al.* 1991] Peter F. Brown, Stephen A. della Pietra, Vincent J. della Pietra, and Robert L. Mercer. Word-Sense Disambiguation Using Statistical Methods. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 264–270, Berkeley, California, June 1991. Association for Computational Linguistics.

[Brown *et al.* 1992a] Peter F. Brown, Stephen A. della Pietra, Vincent J. della Pietra, Jennifer C. Lai, and Robert L. Mercer. An Estimate of an Upper Bound for the Entropy of English. *Computational Linguistics*, **18**(1):31–40, March 1992.

[Brown *et al.* 1992b] Peter F. Brown, Vincent J. della Pietra, Peter V. de Souza, Jennifer C. Lai, and Robert L. Mercer. Class-Based n-gram Models of Natural Language. *Computational Linguistics*, **18**(4):467–479, 1992.

[Brown *et al.* 1993] Peter F. Brown, Stephen A. della Pietra, Vincent J. della Pietra, and Robert L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, **19**(2):263–311, June 1993.

[Brucker 1978] P. Brucker. On the complexity of clustering problems. In Rudolf Henn, Bernhard H. Korte, and W. Oletti, editors, *Optimierung und Operations Research*, Lecture Notes in Economics and Mathematical Systems. Springer, Berlin, 1978.

[Buitelaar 1997] Paul Buitelaar. A Lexicon for Underspecified Semantic Tagging. In *Proceedings of the 1997 Applied Natural Language Conference (ANLP-97) Workshop "Tagging Text with Lexical Semantics: Why, What, and How?"*, pages 25–33, Washington, D.C., April 1997.

[Calzolari and Bindi 1990] Nicoletta Calzolari and Remo Bindi. Acquisition of Lexical Information from a Large Textual Italian Corpus. In Hans Karlgren, editor, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, volume 3, pages 54–59, Helsinki, Finland, 1990.

[Canvar and Trenkle 1994] William B. Canvar and John M. Trenkle. N-Gram-Based Text Categorization. In *Proceedings of the 1994 Symposium on Document Analysis and Information Retrieval*, Las Vegas, Nevada, 1994.

[Carcagno and Iordanskaja 1993] Denis Carcagno and Lidija Iordanskaja. Content Determination and Text Structuring: Two Interrelated Processes. In Helmut Horacek and Michael Zock, editors, *New Concepts in Natural Language Generation: Planning, Realization, and Systems*, pages 10–26. Frances Pinter, London and New York, 1993.

[Carnap 1947] Rudolf Carnap. Meaning Postulates. In *Meaning and Necessity.* University of Chicago Press, Chicago, Illinois, 1947.

[Carroll and Charniak 1992] Glenn Carroll and Eugene Charniak. Learning Probabilistic Dependency Grammars from Labelled Text. In *Probabilistic Approaches to Natural Language: Papers from the 1992 Fall Symposium*, pages 25–32, Cambridge, Massachusetts, October 1992. American Association for Artificial Intelligence, AAAI Press, Menlo Park, California.

[Chafe 1992] Wallace L. Chafe. The Importance of Corpus Linguistics to Understanding the Nature of Language. In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of the Nobel Symposium 82*, pages 79–97. Mouton de Gruyter, Berlin and New York, 1992. Papers from the Symposium held in Stockholm, Sweden in August 1991.

[Chang *et al.* 1995] Jing-Shin Chang, Yi-Chung Lin, and Keh-Yih Su. Automatic Construction of a Chinese Electronic Dictionary. In David Yarowsky and Kenneth W. Church, editors, *Proceedings of the Third Workshop on Very Large Corpora (WVLC-3)*, pages 107–120, Cambridge, Massachusetts, June 1995.

[Chanson and Pinson 1981] Jean-Louis Chanson and Suzanne Pinson. *Analyse Typologique: Théories et Applications.* Masson, Paris, 1981.

[Charniak *et al.* 1993] Eugene Charniak, Curtis Hendrickson, Neil Jacobson, and Mike Perkowitz. Equations for Part-of-Speech Tagging. In *Proceedings of the 11th National Conference on Artificial Intelligence (AAAI-93)*, pages 784–789, Washington, D.C., July 1993. American Association for Artificial Intelligence.

[Chen and Goodman 1996] Stanley F. Chen and Joshua Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 310–318, Santa Cruz, California, June 1996. Association for Computational Linguistics.

[Chinchor 1992] Nancy A. Chinchor. MUC-4 Evaluation Metrics. In *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pages 22–29, McLean, Virginia, June 1992. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Mateo, California.

[Chodorow *et al.* 1985] Martin S. Chodorow, Roy J. Byrd, and George E. Heidron. Extracting Semantic Hierarchies from a Large On-Line Dictionary. In *Proceedings of the 23rd Annual Meeting of the ACL*, pages 299–304, Chicago, Illinois, July 1985. Association for Computational Linguistics.

[Chomsky 1965] Noam Chomsky. *Aspects of the Theory of Syntax.* MIT Press, Cambridge, Massachusetts, 1965.

[Chow and Schwartz 1989] Y. Chow and R. Schwartz. The N-Best Algorithm: An Efficient Search Procedure for Finding Top N Sentence Hypotheses. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 199–202, Cape Cod, Massachusetts, 1989.

[Church and Gale 1991] Kenneth W. Church and William A. Gale. A Comparison of the Enhanced Good-Turing and Deleted Estimation Methods for Estimating Probabilities of English Bigrams. *Computer Speech and Language*, **5**:19–54, 1991.

[Church and Hanks 1990] Kenneth W. Church and Patrick Hanks. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*, **16**(1):22–29, March 1990.

[Church 1988] Kenneth W. Church. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In *Proceedings of the Second Conference on Applied Natural Language Processing (ANLP-88)*, pages 136–143, Austin, Texas, February 1988. Association for Computational Linguistics.

[Cimino *et al.* 1989] James J. Cimino, George Hripcsak, Stephen B. Johnson, and Paul D. Clayton. Designing an Introspective, Controlled Medical Vocabulary. In Lawrence C. Kingsland, editor, *Proceedings of the Thirteenth Annual Symposium on Computer Applications in Medical Care*, pages 513–518, Washington, D.C., November 1989. IEEE Computer Society Press.

[Cimino *et al.* 1994] James J. Cimino, Paul D. Clayton, George Hripcsak, and Stephen B. Johnson. Knowledge-Based Approaches to the Maintenance of a Large Controlled Medical Terminology. *Journal of the American Medical Informatics Association (JAMIA)*, **1**(1):35–50, January/February 1994.

[Clémencin 1996] G. Clémencin. Integration of a CL-Checker in an Operational SGML Authoring Environment. In *Proceedings of the First International Workshop on Controlled Language Applications*, pages 32–40, Katholieke Universiteit Leuven, Belgium, 1996.

[Coltheart 1981] Max Coltheart. The MRC Psycholinguistic Database. *Quarterly Journal of Experimental Psychology*, **33A**:497–505, 1981.

[Conover 1980] W. J. Conover. *Practical Nonparametric Statistics*. Wiley, New York, 2nd edition, 1980.

[Cover and Thomas 1991] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[Cutting *et al.* 1992] Douglas R. Cutting, Julian M. Kupiec, Jan O. Pedersen, and Penelope Sibun. A Practical Part-of-Speech Tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing (ANLP-92)*, pages 133–140, Trento, Italy, April 1992. Association for Computational Linguistics.

[Dagan and Church 1994] Ido Dagan and Kenneth W. Church. *Termight*: Identifying and Translating Technical Terminology. In *Proceedings of the Fourth Conference on Applied Natural Language Processing (ANLP-94)*, pages 34–40, Stuttgart, Germany, October 1994. Association for Computational Linguistics.

[Dagan and Itai 1994] Ido Dagan and Alon Itai. Word Sense Disambiguation Using a Second Language Monolingual Corpus. *Computational Linguistics*, **20**(4):563–596, December 1994.

[Dagan *et al.* 1991] Ido Dagan, Alon Itai, and Ulrike Schwall. Two Languages Are More Informative Than One. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 130–137, Berkeley, California, June 1991. Association for Computational Linguistics.

[Dagan *et al.* 1993] Ido Dagan, Shaul Marcus, and Shaul Markovitch. Contextual Word Similarity and Estimation from Sparse Data. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 164–171, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Daille 1994] Béatrice Daille. *Approche Mixte pour l' Extraction Automatique de Terminologie: Statistique Lexicale et Filtres Linguistiques*. PhD thesis, Université Paris VII, Paris, 1994.

[Dalal *et al.* 1996a] Mukesh Dalal, Steven K. Feiner, Kathleen R. McKeown, Desmond A. Jordan, Barry Allen, and Yasser al Safadi. MAGIC: An Experimental System for Generating Multimedia Briefings about Post-Bypass Patient Status. In *Proceedings of the 1996 Annual Fall Symposium of the American Medical Informatics Association (AMIA-96)*, pages 684–688, Washington, D.C., October 26–30 1996.

[Dalal *et al.* 1996b] Mukesh Dalal, Steven K. Feiner, Kathleen R. McKeown, Shimei Pan, Michelle Zhou, Tobias H. Höllerer, James Shaw, Yong Feng, and Jeanne Fromer. Negotiation for Automated Generation of Temporal Multimedia Presentations. In *Proceedings of 1996 ACM Conference on Multimedia*, pages 55–64, Boston, Massachusetts, November 18–22 1996.

[Daniel 1990] Wayne W. Daniel. *Applied Nonparametric Statistics.* PWS-KENT Publishing Company, Boston, Massachusetts, 2nd edition, 1990.

[Danlos 1986] Laurence Danlos. *The Linguistic Basis of Text Generation.* Studies in Natural Language Processing. Cambridge University Press, Cambridge, England, 1986.

[Davidson 1967] Donald Davidson. Truth and Meaning. *Synthese*, **17**:304–323, 1967. Reprinted in [Rosenberg and Travis 1971].

[Davis 1987] Lawrence Davis, editor. *Genetic Algorithms and Simulated Annealing.* Pitman and Morgan Kaufmann, London and Los Altos, California, 1987.

[de Marcken 1990] Carl G. de Marcken. Parsing the LOB corpus. In *Proceedings of the 28th Annual Meeting of the ACL*, pages 243–251, Pittsburgh, Pennsylvania, June 1990. Association for Computational Linguistics.

[de Rose 1988] Steven J. de Rose. Grammatical Category Disambiguation by Statistical Optimization. *Computational Linguistics*, **14**(1):31–39, Winter 1988.

[de Saussure 1983] Ferdinand de Saussure. *Course in General Linguistics.* Duckworth, London, 1983. Edited by Charles Bally and Albert Sechehaye, with the collaboration of Albert Riedlinger. Translated and annotated by Roy Harris. French text edited by Eisuke Komatsu. Translation from *Troisieme Cours de Linguistique Générale (1910–1911): D' aprés les cahiers d' Emile Constantin.*

[Deese 1964] James Deese. The Associative Structure of Some Common English Adjectives. *Journal of Verbal Learning and Verbal Behavior*, **3**(5):347–357, 1964.

[Dice 1945] Lee R. Dice. Measures of the Amount of Ecologic Association between Species. *Journal of Ecology*, **26**:297–302, 1945.

[Dolby *et al.* 1963] James L. Dolby, Howard L. Resnikoff, and F. L. MacMurray. A Tape Dictionary for Linguistic Experiments. In *Proceedings of the 1963 Fall Joint Computer Conference*, volume 24, pages 419–423, Las Vegas, Nevada, November 1963. American Federation of Information Processing Societies (AFIPS), Spartan Books, Baltimore, Maryland.

[Douglas and Hurst 1996] S. Douglas and M. Hurst. Controlled Language Support for Perkings Approved Clear English. In *Proceedings of the First International Workshop on Controlled Language Applications*, pages 93–105, Katholieke Universiteit Leuven, Belgium, 1996.

[Dowty 1979] David R. Dowty. *World Meaning and Montague Grammar: The Semantics of Verbs and Times in Generative Semantics and in Montague's PTQ.* Reidel, Dordrecht, The Netherlands, 1979.

[Draper and Smith 1981] Norman R. Draper and Harry Smith. *Applied Regression Analysis.* Wiley, New York, 2nd edition, 1981.

[Duda and Hart 1973] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis.* Wiley, New York, 1973.

[Dumas 1955] F. N. Dumas. *Manifest Structure Analysis.* Montana State University Press, Missoula, Montana, 1955.

[Dunn 1973] J. C. Dunn. A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, **3**(3):32–57, July–September 1973.

[Dunning 1993] Ted Dunning. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, **19**(1):61–74, March 1993.

[Dunning 1994] Ted Dunning. Statistical Identification of Language. Technical Report MCCS-94-273, CRL, New Mexico State University, Las Cruces, New Mexico, 1994.

[Edwards and Cavalli-Sforza 1965] A. W. F. Edwards and L. L. Cavalli-Sforza. A Method for Cluster Analysis. *Biometrics*, **21**:362–375, 1965.

[Edwards 1992] Jane Anne Edwards. Survey of Electronic Corpora and Related Resources for Language Researchers. In Jane Anne Edwards and Martin D. Lampert, editors, *Talking Data: Transcription and Coding in Discourse Research*, chapter 10, pages 263–310. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1992.

[Elhadad and McKeown 1990] Michael Elhadad and Kathleen R. McKeown. A Procedure for Generating Connectives. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, Helsinki, Finland, July 1990.

[Elhadad 1991] Michael Elhadad. Generating Adjectives to Express the Speaker's Argumentative Intent. In *Proceedings of the 9th National Conference on Artificial Intelligence (AAAI-91)*, pages 98–104, Anaheim, California, July 1991. American Association for Artificial Intelligence.

[Elhadad 1993] Michael Elhadad. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation.* PhD thesis, Department of Computer Science, Columbia University, New York, 1993.

[Evans and Zhai 1996] David A. Evans and Chengxiang Zhai. Noun-Phrase Analysis in Unrestricted Text for Information Retrieval. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 17–24, Santa Cruz, California, June 1996. Association for Computational Linguistics.

[Everitt 1980] Brian S. Everitt. *Cluster Analysis.* Halsted Press, New York, 1980.

[Fano 1961] Robert M. Fano. *Transmission of Information: A Statistical Theory of Communications.* MIT Press, Cambridge, Massachusetts, 1961.

[Federici *et al.* 1996] Stefano Federici, Simonetta Montemagni, and Vito Pirrelli. Shallow Parsing and Text Chunking: A View on Underspecification in Syntax. In *Proceedings of the ESSLLI-96 Workshop on Robust Parsing*, pages 12–16, Prague, Czech Republic, August 1996.

[Fillenbaum and Rapoport 1971] Samuel Fillenbaum and Amnon Rapoport. *Structures in the Subjective Lexicon.* Academic Press, New York, 1971.

[Fillmore 1992] Charles J. Fillmore. "Corpus Linguistics" or "Corpus-Based Armchair Linguistics". In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of the Nobel Symposium 82*, pages 35–60. Mouton de Gruyter, Berlin and New York, 1992. Papers from the Symposium held in Stockholm, Sweden in August 1991.

[Firth 1957] John R. Firth. Modes of Meaning. In *Papers in Linguistics 1934–1951.* Oxford University Press, London and New York, 1957.

[Fisher 1934] Ronald A. Fisher. *Statistical Methods for Research Workers.* Oliver and Boyd, Edinburgh, United Kingdom, 5th edition, 1934.

[Fleiss 1981] Joseph L. Fleiss. *Statistical Methods for Rates and Proportions.* Wiley, New York, 2nd edition, 1981.

[Florek *et al.* 1951] K. Florek, Jozef Łukaszewicz, Julian Perkal, Hugo Steinhaus, and Stefan Zubrzycki. Sur la Liaison et la Division des Points d' un Ensemble Fini. *Colloquium Mathematicum*, **2**(3–4):282–285, 1951.

[Frakes and Baeza-Yates 1992] William B. Frakes and Ricardo Baeza-Yates, editors. *Information Retrieval: Data Structures and Algorithms.* Prentice Hall, Englewood Cliffs, New Jersey, 1992.

[Francis and Kučera 1982] W. Nelson Francis and Henry Kučera. *Frequency Analysis of English Usage: Lexicon and Grammar.* Houghton Mifflin, Boston, Massachusetts, 1982.

[Fung 1995] Pascale Fung. A Pattern Matching Method for Finding Noun and Proper Noun Translations from Noisy Parallel Corpora. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 236–243, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Gale and Church 1991] William A. Gale and Kenneth W. Church. Identifying Word Correspondences in Parallel Texts. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, pages 152–157, Asilomar, California, February 1991. Morgan Kaufmann, San Mateo, California.

[Gale *et al.* 1992a] William A. Gale, Kenneth W. Church, and David Yarowsky. Estimating Upper and Lower Bounds on the Performance of Word-Sense Disambiguation Programs. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 249–256, Newark, Delaware, June 1992. Association for Computational Linguistics.

[Gale *et al.* 1992b] William A. Gale, Kenneth W. Church, and David Yarowsky. Work on Statistical Methods for Word Sense Disambiguation. In *Probabilistic Approaches to Natural Language: Papers from the 1992 Fall Symposium*, pages 54–60, Cambridge, Massachusetts, October 1992. American Association for Artificial Intelligence, AAAI Press, Menlo Park, California.

[Gale *et al.* 1992c] William A. Gale, Kenneth W. Church, and David Yarowsky. One Sense Per Discourse. In *Proceedings of the DARPA Speech and Natural Language Workshop*, February 1992.

[Garey and Johnson 1979] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, San Francisco, California, 1979.

[Garside and Leech 1985] Roger Garside and Fanny Leech. A Probabilistic Parser. In *Proceedings of the Second European Conference of the ACL*, pages 166–170, Geneva, Switzerland, March 1985. Association for Computational Linguistics.

[Gazdar 1979] Gerald Gazdar. *Pragmatics: Implicature, Presupposition, and Logical Form*. Academic Press, New York, 1979.

[Gellerstam 1988] Martin Gellerstam, editor. *Studies in Computer-Aided Lexicology*. Almqvist and Wiksell International, Stockholm, Sweden, 1988.

[Geman and Geman 1984] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**:721–741, 1984.

[Gibbons and Chakraborti 1992] Jean Dickinson Gibbons and Subhabrata Chakraborti. *Nonparametric Statistical Inference*. Marcel Dekker, New York, 3rd edition, 1992.

[Golub and van Loan 1989] Gene H. Golub and Charles F. van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, 2nd edition, 1989.

[Good 1965] Irving J. Good. A Categorization of Classification. In *Mathematics and Computer Science in Biology and Medicine: Proceedings of the Conference held by the Medical Research Council in Association with the Health Departments, Oxford, July 1964*, pages 115–128. Her Majesty's Stationery Office, London, 1965.

[Gordon 1981] A. D. Gordon. *Classification: Methods for the Exploratory Analysis of Multivariate Data*. Chapman and Hall, London and New York, 1981.

[Gower and Ross 1969] J. C. Gower and Gavin J. S. Ross. Minimum Spanning Trees and Single Linkage Cluster Analysis. *Applied Statistics*, **18**(1):54–64, 1969.

[Greenbaum 1996] Sidney Greenbaum. *The Oxford English Grammar*. Oxford University Press, London, 1996.

[Greenberg 1966] Joseph H. Greenberg. *Language Universals*. Mouton, The Hague, The Netherlands, 1966.

[Grefenstette 1992a] Gregory Grefenstette. Finding Semantic Similarity in Raw Text: The Deese Antonyms. In *Probabilistic Approaches to Natural Language: Papers from the 1992 Fall Symposium*, pages 61–66, Cambridge, Massachusetts, October 1992. American Association for Artificial Intelligence, AAAI Press, Menlo Park, California.

[Grefenstette 1992b] Gregory Grefenstette. Use of Syntactic Context to Produce Term Association Lists for Text Retrieval. In Nicholas J. Belkin, Peter Ingwersen, and Annelise M. Pejtersen, editors, *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-92)*, Copenhagen, Denmark, June 1992. Association for Computing Machinery, Special Interest Group on Information Retrieval, ACM Press, New York.

[Grice 1967] H. Paul Grice. Logic and Conversation. Unpublished manuscript of the William James Lectures, Harvard University, 1967.

[Grice 1975] H. Paul Grice. Logic and Conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics 3: Speech Acts*, pages 41–58. Academic Press, New York, 1975. Part of [Grice 1967].

[Grice 1978] H. Paul Grice. Further Notes on Logic and Conversation. In Peter Cole, editor, *Syntax and Semantics 9: Pragmatics*, pages 113–127. Academic Press, New York, 1978. Part of [Grice 1967].

[Guthrie *et al.* 1991] Joe A. Guthrie, Louise Guthrie, Yorick Wilks, and Homa Aidinejad. Subject-Dependent Co-occurrence and Word Sense Disambiguation. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 146–152, Berkeley, California, June 1991. Association for Computational Linguistics.

[Halliday 1992] Michael A. K. Halliday. Language as System and Language as Instance: The Corpus as a Theoretical Construct. In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of the Nobel Symposium 82*, pages 61–77. Mouton de Gruyter, Berlin and New York, 1992. Papers from the Symposium held in Stockholm, Sweden in August 1991.

[Harman 1993] Donna Harman. Overview of TREC-1. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 61–65, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Harris 1951] Zellig S. Harris. *Structural Linguistics*. Chicago University Press, Chicago, Illinois, 1951.

[Harris 1968] Zellig S. Harris. *Mathematical Structures of Language*. Number 21 in Interscience Tracts in Pure and Applied Mathematics. Wiley-Interscience, New York, 1968.

[Hartigan 1972] John A. Hartigan. Direct Clustering of a Data Matrix. *Journal of the American Statistical Association*, **67**(337):123–129, March 1972.

[Hartigan 1975] John A. Hartigan. *Clustering Algorithms*. Wiley-Interscience, New York, 1975.

[Haruno and Yamazaki 1996] Masahiko Haruno and Takefumi Yamazaki. High-Performance Bilingual Text Alignment Using Statistical and Dictionary Information. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 131–138, Santa Cruz, California, June 1996. Association for Computational Linguistics.

[Hastie and Pregibon 1990] Trevor Hastie and D. Pregibon. Shrinking Trees. Technical report, AT&T Bell Laboratories, 1990.

[Hatzivassiloglou and Knight 1995] Vasileios Hatzivassiloglou and Kevin Knight. Unification-Based Glossing. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, volume 2, pages 1382–1389, Montréal, Québec, Canada, August 1995. Morgan Kaufmann, San Mateo, California.

[Hatzivassiloglou and McKeown 1993] 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 ACL*, pages 172–182, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Hatzivassiloglou and McKeown 1995] Vasileios Hatzivassiloglou and Kathleen McKeown. A Quantitative Evaluation of Linguistic Tests for the Automatic Prediction of Semantic Markedness. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 197–204, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Hatzivassiloglou and McKeown 1998] Vasileios Hatzivassiloglou and Kathleen R. McKeown. An Automatic Tool for the Sanitization of Medical Texts, 1998. In preparation.

[Hayes *et al.* 1996] P. Hayes, S. Maxwell, and L. Schmandt. Controlled English Advantages for Translated and Original English Documents. In *Proceedings of the First International Workshop on Controlled Language Applications*, pages 84–92, Katholieke Universiteit Leuven, Belgium, 1996.

[Hayes 1994] P. Hayes. NameFinder: Software that Finds Names in Text. In *Proceedings of RIAO-94*, pages 762–774, New York, October 1994.

[Hearst 1992] Marti A. Hearst. Automatic Acquisition of Hyponyms from Large Text Corpora. In Christian Boitet, editor, *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 539–545, Nantes, France, August 1992.

[Hearst 1994] Marti A. Hearst. Multi-Paragraph Segmentation of Expository Text. In *Proceedings of the 32nd Annual Meeting of the ACL*, pages 9–16, Las Cruces, New Mexico, June 1994. Association for Computational Linguistics.

[Heeman and Allen 1994] Peter A. Heeman and James F. Allen. Detecting and Correcting Speech Repairs. In *Proceedings of the 32nd Annual Meeting of the ACL*, pages 295–302, Las Cruces, New Mexico, June 1994. Association for Computational Linguistics.

[Heeman and Allen 1997] Peter A. Heeman and James F. Allen. Intonational Boundaries, Speech Repairs and Discourse Markers: Modeling Spoken Dialog. In *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL*, pages 254–261, Madrid, Spain, July 1997. Association for Computational Linguistics.

[Herder 1877] Johann G. Herder. Abhandlung über den Ursprung der Sparche (1772). In Bernhard L. Suphan, editor, *Herder's Sämmtliche Werke*, volume 2. Berlin, 1877.

[Hicks 1982] Charles R. Hicks. *Fundamental Concepts in the Design of Experiments*. Holt, Rinehart, and Wilson, New York, 3rd edition, 1982.

[Hindle and Rooth 1991] Donald Hindle and Mats Rooth. Structural Ambiguity and Lexical Relations. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 229–236, Berkeley, California, June 1991. Association for Computational Linguistics.

[Hindle 1983] Donald Hindle. User Manual for FIDDITCH, a Deterministic Parser. Technical Memorandum 7590-142, Naval Research Laboratory, 1983.

[Hindle 1990] Donald Hindle. Noun Classification from Predicate-Argument Structures. In *Proceedings of the 28th Annual Meeting of the ACL*, pages 268–275, Pittsburgh, Pennsylvania, June 1990. Association for Computational Linguistics.

[Hindle 1994] Donald Hindle. A Parser for Text Corpora. In B. T. Sue Atkins and Antonio Zampoli, editors, *Computational Approaches to the Lexicon*, pages 103–151. Oxford University Press, Oxford, England, 1994.

[Hintikka 1962] Jaakko J. Hintikka. *Knowledge and Belief.* Cornell University Press, Ithaca, New York, 1962.

[Hintikka 1969] Jaakko J. Hintikka. Semantics for Propositional Attitudes. In J. W. Davis, D. J. Hockney, and W. K. Wilson, editors, *Philosophical Logic*, pages 21–45. Reidel, Dordrecht, The Netherlands, 1969.

[Hirshberg 1985] Julia B. Hirshberg. *A Theory of Scalar Implicature.* PhD thesis, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, Pennsylvania, 1985.

[Hoard *et al.* 1992] James E. Hoard, R. H. Wojcik, and K. Holzhauser. An Automated Grammar and Style Checker for Writers of Simplified English. In Patrik O. Holt and Noel Williams, editors, *Computers and Writing: State of the Art.* Kluwer Academic Publishers, Dordrecht, The Netherlands, and Boston, Massachusetts, 1992.

[Hobbs *et al.* 1993] Jerry R. Hobbs, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, and Mabry Tyson. FASTUS: A System for Extracting Information from Text. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 133–137, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Horn 1969] Laurence R. Horn. A Presuppositional Analysis of *Only* and *Even*. In *Papers from the Fifth Regional Meeting*, pages 98–107, Chicago, Illinois, 1969. Chicago Linguistics Society.

[Horn 1972] Laurence R. Horn. *On the Semantic Properties of the Logical Operators in English.* Indiana University Linguistics Club, Bloomington, Indiana, 1972.

[Huber 1981] Peter J. Huber. *Robust Statistics.* Wiley, New York, 1981.

[Hubert 1973] L. J. Hubert. Monotone Invariant Clustering Procedures. *Psychometrika*, **38**:47–62, 1973.

[Hubler 1983] Axel Hubler. *Understatements and Hedges in English.* Pragmatics and Beyond. John Benjamins Publishing Company, Amsterdam and Philadelphia, 1983.

[International Standards Organization 1983] International Standards Organization. *ISO 646 – Information Processing Systems: 7-Bit Coded Character Set for Information Processing Interchange*, 1983. Reference No. ISO 646-1983(E); ISO 646 RVC (Reference Version Character) Set.

[International Standards Organization 1986] International Standards Organization. *ISO 8879 – Information Processing Systems – Text and office systems: Standard Generalized Markup Language (SGML)*, 1986.

[Iordanskaja *et al.* 1994] Lidija Iordanskaja, M. Kim, Richard Kittredge, B. Lavoie, and Alain Polguère. Generation of Extended Bilingual Statistical Reports. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-94)*, Kyoto, Japan, 1994.

[Iyer *et al.* 1994] Rukmini Iyer, Mari Ostendorf, and J. Robin Rohlicek. Language Modeling with Sentence-Level Mixtures. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 82–87, Plainsboro, New Jersey, March 1994. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Jaccard 1908] Paul Jaccard. Nouvelles Recherches sur la Distribution Florale. *Bulletin de la Societé Vaudoise des Sciences Naturelles*, **44**(163):223–270, 1908.

[Jackendoff 1985] Ray Jackendoff. *Semantics and Cognition.* MIT Press, Cambridge, Massachusetts, 1985.

[Jackendoff 1990] Ray Jackendoff. *Semantic Structures.* MIT Press, Cambridge, Massachusetts, 1990.

[Jakobson 1962] Roman Jakobson. *Phonological Studies*, volume 1 of *Selected Writings.* Mouton, The Hague, The Netherlands, 1962.

[Jakobson 1984a] Roman Jakobson. The Structure of the Russian Verb (1932). In Linda R. Waugh and Morris Halle, editors, *Russian and Slavic Grammar Studies 1931-1981*, pages 1–14. Mouton, Berlin, 1984.

[Jakobson 1984b] Roman Jakobson. Zero Sign (1939). In Linda R. Waugh and Morris Halle, editors, *Russian and Slavic Grammar Studies 1931-1981*, pages 151–160. Mouton, Berlin, 1984.

[Jelinek and Mercer 1980] Fredrick Jelinek and Robert L. Mercer. Interpolated Estimation of Markov Source Parameters from Sparse Data. In Edzard S. Gelsema and Laveen N. Kanal, editors, *Pattern Recognition in Practice: Proceedings of an International Workshop held in Amsterdam, May 1980.* North-Holland, Amsterdam, The Netherlands, 1980.

[Jing *et al.* 1997] Hongyan Jing, Vasileios Hatzivassiloglou, Rebecca J. Passonneau, and Kathleen R. McKeown. Investigating Complementary Methods for Verb Sense Pruning. In *Proceedings of the 1997 Applied Natural Language Conference (ANLP-97) Workshop "Tagging Text with Lexical Semantics: Why, What, and How?"*, pages 58–65, Washington, D.C., April 1997.

[Johnson and Kotz 1970] Norman L. Johnson and Samuel Kotz. *Continuous Univariate Distributions*, volume 1. Houghton-Mifflin, Boston, Massachusetts, 1970.

[Jones and Eisner 1992] Mark A. Jones and Jason M. Eisner. A Probabilistic Parser and Its Application. In *Proceedings of the AAAI-92 Workshop on Statistically-Based NLP Techniques*, pages 20–27, San Jose, California, July 1992. American Association for Artificial Intelligence.

[Justeson and Katz 1991] John S. Justeson and Slava M. Katz. Co-occurrences of Antonymous Adjectives and Their Contexts. *Computational Linguistics*, **17**(1):1–19, 1991.

[Justeson and Katz 1995a] John S. Justeson and Slava M. Katz. Principled Disambiguation: Discriminating Adjective Senses with Modified Nouns. *Computational Linguistics*, **21**(1):1–27, March 1995.

[Justeson and Katz 1995b] John S. Justeson and Slava M. Katz. Technical Terminology: Some Linguistic Properties and an Algorithm for Identification in Text. *Natural Language Engineering*, **1**(1):9–27, 1995.

[Kamp 1981] Johan A. W. Kamp. A Theory of Truth and Semantic Representation. In Jeroen A. G. Groenendijk, Theo M. V. Janssen, and Martin B. J. Stokhof, editors, *Formal Methods in the Study of Language*, number 135 in Mathematical Centre Tracts. Mathematisch Centrum, Amsterdam, 1981. Proceedings of a colloquium held in Amsterdam, March 25–28, 1980.

[Kaplan and Bresnan 1982] Ronald M. Kaplan and Joan Bresnan. Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Joan Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–281. MIT Press, Cambridge, Massachusetts, 1982.

[Katz 1972] Jerrold J. Katz. *Semantic Theory.* Harper and Row, New York, 1972.

[Kaufman and Plastria 1981] Leonard Kaufman and Frank Plastria. Non-hierarchical Twoway Clustering. Working paper, University of Brussels, Brussels, Belgium, 1981.

[Kaufman and Rousseeuw 1987] Leonard Kaufman and Peter J. Rousseeuw. Clustering by Means of Medoids. In Yadolah Dodge, editor, *Statistical Data Analysis Based on the L_1 Norm*, pages 405–416. Elsevier/North-Holland, Amsterdam, The Netherlands, 1987.

[Kaufman and Rousseeuw 1990] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, New York, 1990.

[Keenan and Faltz 1985] Edward L. Keenan and Leonard M. Faltz. *Boolean Semantics for Natural Language.* Number 23 in Synthese Language Library. Reidel, Dordrecht, The Netherlands, 1985.

[Kempson 1977] Ruth M. Kempson. *Semantic Theory.* Cambridge University Press, Cambridge, England, 1977.

[Kendall 1938] Maurice G. Kendall. A New Measure of Rank Correlation. *Biometrika*, **30**(1–2):81–93, June 1938.

[Kendall 1975] Maurice G. Kendall. *Rank Correlation Methods.* Griffin, London, 4th edition, 2nd impression, 1975.

[Kittredge and Grishman 1986] Richard Kittredge and Ralph Grishman, editors. *Analyzing Language in Restricted Domains: Sublanguage Description and Processing.* L. Erlbaum Associates, Hillsdale, New Jersey, 1986.

[Kittredge and Lehrberger 1982] Richard Kittredge and John Lehrberger, editors. *Sublanguage: Studies of Language in Restricted Semantic Domains.* Foundations of communication. Walter de Gruyter, Berlin and New York, 1982.

[Kjellmer 1984] Goran Kjellmer. Some Thoughts on Collocational Distinctiveness. In Jan M. Aarts and Willem Meijs, editors, *Corpus Linguistics: Recent Developments in the Use of Computer Corpora in English Language Research*, pages 163–171. Rodopi, Amsterdam, 1984.

[Klastorin 1985] Theodore D. Klastorin. The p-median Problem for Cluster Analysis: A Comparative Test Using the Mixture Model Approach. *Management Science*, **31**(1):84–95, January 1985.

[Klavans and Chodorow 1992] Judith L. Klavans and Martin S. Chodorow. Degrees of Stativity: The Lexical Representation of Verb Aspect. In Christian Boitet, editor, *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 1126–1131, Nantes, France, August 1992.

[Klavans and Resnik 1996] Judith L. Klavans and Philip S. Resnik, editors. *The Balancing Act: Combining Symbolic and Statistical Approaches to Language.* The MIT Press, Cambridge, Massachusetts, 1996.

[Klavans and Tzoukermann 1990] Judith L. Klavans and Evelyne Tzoukermann. The BICORD System: Combining Lexical Information from Bilingual Corpora and Machine Readable Dictionaries. In Hans Karlgren, editor, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, volume 3, pages 174–179, Helsinki, Finland, 1990.

[Knight and Hatzivassiloglou 1995] Kevin Knight and Vasileios Hatzivassiloglou. Two-Level, Many-Paths Generation. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 252–260, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Knight and Luk 1994] Kevin Knight and Steve K. Luk. Building a Large-Scale Knowledge Base for Machine Translation. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-94)*, volume 1, pages 773–778, Seattle, Washington, July–August 1994. American Association for Artificial Intelligence.

[Knuth 1973] Donald E. Knuth. *Fundamental Algorithms*, volume 1 of *The Art of Computer Programming.* Addison-Wesley, Reading, Massachusetts, 2nd edition, 1973.

[Kohonen 1984] Teuvo Kohonen. *Self-organization and Associative Memory.* Number 8 in Series in Information Sciences. Springer-Verlag, Berlin, 1984.

[Kukich 1983] Karen K. Kukich. Design of a Knowledge-Based Report Generator. In *Proceedings of the 21st Annual Meeting of the ACL*, pages 145–150, Cambridge, Massachusetts, June 1983. Association for Computational Linguistics.

[Kupiec 1992] Julian M. Kupiec. Robust Part-of-Speech Tagging Using a Hidden Markov Model. *Computer Speech and Language*, **6**:225–242, 1992.

[Kupiec 1993] Julian M. Kupiec. An Algorithm for Finding Noun Phrase Correspondences in Bilingual Corpora. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 17–22, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Kučera and Francis 1967] Henry Kučera and W. Nelson Francis. *Computational Analysis of Present-Day American English.* Brown University Press, Providence, Rhode Island, 1967.

[Kučera 1982] Henry Kučera. Markedness and Frequency: A Computational Analysis. In Ján Horecký, editor, *Proceedings of the Ninth International Conference on Computational Linguistics (COLING-82)*, pages 167–173, Prague, Czechoslovakia, July 1982. North-Holland, Amsterdam, The Netherlands.

[Lakatos 1972] Imre Lakatos. Falsification and the Methodology of Scientific Research Programmes. In Imre Lakatos and Alan Musgrave, editors, *Criticism and the Growth of Knowledge.* Cambridge University Press, Cambridge, England, 1972. Papers from the International Colloquium in the Philosophy of Science, Bedford College, 1965.

[Lakoff 1987] George Lakoff. *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind.* University of Chicago Press, Chicago, Illinois, 1987.

[Lance and Williams 1966] Godfrey N. Lance and G. T. Williams. Computer Programs for Hierarchical Polythetic Classification. *Computer Journal*, **9**:60–64, 1966.

[Lance and Williams 1979] G. N. Lance and W. T. Williams. INVER: A Program for the Computation of Distance-Measures Between Attributes of Mixed Types. *Australian Computer Journal*, **11**:27–28, 1979.

[Lauer 1995] Mark Lauer. Corpus Statistics Meet the Noun Compound: Some Empirical Results. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 47–54, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Leacock *et al.* 1993] Claudia Leacock, Geoffrey Towell, and Ellen Voorhees. Corpus-Based Statistical Sense Resolution. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 260–265, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Lee 1997] Lillian Lee. *Similarity-Based Approaches to Natural Language Processing.* PhD thesis, Department of Computer Science, Harvard University, Cambridge, Massachusetts, 1997. Also Harvard University Technical Report TR-11-97.

[Leech 1975] Geoffrey N. Leech. *Semantics.* Penguin, 1975.

[Leech 1991] Geoffrey N. Leech. The State of the Art in Corpus Linguistics. In Karin Aijmer and Bengt Altenberg, editors, *English Corpus Linguistics: Studies in Honour of Jan Svartvik*, pages 8–29. Longman, London, 1991.

[Leech 1992] Geoffrey N. Leech. Corpora and Theories of Linguistic Performance. In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of the Nobel Symposium 82*, pages 105–122. Mouton de Gruyter, Berlin and New York, 1992. Papers from the Symposium held in Stockholm, Sweden in August 1991.

[Lehrer 1974] Adrienne Lehrer. *Semantic Fields and Lexical Structure.* North Holland, Amsterdam and New York, 1974.

[Lehrer 1985] Adrienne Lehrer. Markedness and Antonymy. *Journal of Linguistics*, **31**(3):397–429, September 1985.

[Lesk and Schmidt 1975] Michael E. Lesk and Eric Schmidt. *Lex — A Lexical Analyzer Generator.* AT&T Bell Laboratories, Murray Hill, New Jersey, July 1975. Unix Research System Programmer's Manual.

[Lesk 1986] Michael E. Lesk. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 1986 ACM SIGDOC Conference*, pages 24–26, Toronto, Ontario, Canada, 1986. Association for Computing Machinery, Special Interest Group on Documentation.

[Levelt 1983] Willem J. M. Levelt. Monitoring and Self-Repair in Speech. *Cognition*, **14**:41–104, 1983.

[Levin 1993] Beth Levin. *English Verb Classes and Alternations: A Preliminary Investigation.* University of Chicago Press, Chicago, Illinois, 1993.

[Levine *et al.* 1992] John R. Levine, Tony Mason, and Doug Brown. *Lex and Yacc.* Nutshell Handbooks. O'Reilly and Associates, Sebastopol, California, 2nd edition, 1992.

[Levinson 1983] Stephen C. Levinson. *Pragmatics.* Cambridge University Press, Cambridge, England, 1983.

[Lewis and Tong 1992] David D. Lewis and Richard M. Tong. Text Filtering in MUC-3 and MUC-4. In *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pages 51–66, McLean, Virginia, June 1992. DARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Mateo, California.

[Lewis 1972] David K. Lewis. General Semantics. In Gilbert Harman and Donald Davidson, editors, *Semantics of Natural Language*, pages 169–218. Reidel, Dordrecht, The Netherlands, 1972.

[Lewis 1973] David K. Lewis. *Counterfactuals.* Blackwell, Oxford, 1973.

[Li and Abe 1996] Hang Li and Naoki Abe. Clustering Words with the MDL Principle. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 4–9, Copenhagen, Denmark, August 1996.

[Libert and Roubens 1982] G. Libert and M. Roubens. Non-Metric Fuzzy Clustering Algorithms and their Cluster Validity. In Madan M. Gupta and Elie Sanchez, editors, *Approximate Reasoning in Decision Analysis.* North-Holland, Amsterdam, The Netherlands, 1982.

[Liddy and Paik 1992] Elizabeth D. Liddy and Woojin Paik. Statistically-Guided Word Sense Disambiguation. In *Probabilistic Approaches to Natural Language: Papers from the 1992 Fall Symposium*, pages 98–107, Cambridge, Massachusetts, October 1992. American Association for Artificial Intelligence, AAAI Press, Menlo Park, California.

[Lindeman *et al.* 1980] Richard H. Lindeman, Peter F. Merenda, and Ruth Z. Gold. *Introduction to Bivariate and Multivariate Analysis.* Scott, Foresman, and Company, Glenview, Illinois, 1980.

[Luk 1995] Alpha K. Luk. Statistical Sense Disambiguation with Relatively Small Corpora Using Dictionary Definitions. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 181–188, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Lyon and Brown 1997] Caroline Lyon and Stephen Brown. Evaluating Parsing Schemes with Entropy Indicators. In *Proceedings of the 5th Meeting on Mathematics of Language (MOL-5)*, Saarbrücken, Germany, August 1997.

[Lyons 1977] John Lyons. *Semantics*, volume 1. Cambridge University Press, Cambridge, England, 1977.

[MacNaughton-Smith *et al.* 1964] P. MacNaughton-Smith, W. T. Williams, M. B. Dale, and L. G. Mockett. Dissimilarity Analysis: A New Technique of Hierarchical Sub-division. *Nature*, **202**:1034–1035, 1964.

[MacQueen 1967] J. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In Lucien M. Le Cam and Jerzy Neyman, editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, Berkeley, California, 1967.

[Magerman and Marcus 1990] David M. Magerman and Mitchell P. Marcus. Parsing a Natural Language Using Mutual Information Statistics. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, volume 2, pages 984–989. American Association for Artificial Intelligence, July–August 1990.

[Magerman and Marcus 1991] David M. Magerman and Mitchell P. Marcus. PEARL: A Probabilistic Chart Parser. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, Asilomar, California, February 1991. Morgan Kaufmann, San Mateo, California.

[Magerman 1995] David M. Magerman. Statistical Decision-Tree Models for Parsing. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 276–283, Boston, Massachusetts, June 1995. Association for Computational Linguistics.

[Mahalanobis 1936] Prasanta C. Mahalanobis. On the Generalized Distance in Statistics. *Proceedings of the National Institute of Sciences of India*, **12**:49–55, 1936.

[Malkiel 1959] Yakov Malkiel. Studies in Irreversible Binomials. *Lingua*, **8**(2):113–160, May 1959. Reprinted in [Malkiel 1968].

[Malkiel 1968] Yakov Malkiel. *Essays on Linguistic Themes*. Blackwell, Oxford, 1968.

[Manning 1993] Christopher D. Manning. Automatic Acquisition of a Large Subcategorization Dictionary from Corpora. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 235–242, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Massart *et al.* 1983] Désiré L. Massart, Frank Plastria, and Leonard Kaufman. Non-hierarchical Clustering with MASLOC. *Pattern Recognition*, **16**(5):507–516, 1983.

[McCormick *et al.* 1972] William T. McCormick, Jr., Paul J. Schweitzer, and Thomas W. White. Problem Decomposition and Data Reorganization by a Clustering Technique. *Operations Research*, **20**(5):993–1009, September–October 1972.

[McCullagh and Nelder 1989] Peter McCullagh and John A. Nelder. *Generalized Linear Models*. Chapman and Hall, London, 2nd edition, 1989.

[McDonald *et al.* 1990] James E. McDonald, Tony Plate, and Roger W. Schvaneveldt. Using Pathfinder to Extract Semantic Information from Text. In Roger W. Schvaneveldt, editor, *Pathfinder Associative Networks: Studies in Knowledge Organization*. Ablex, Norwood, New Jersey, 1990.

[McKee and Maloney 1992] Doug McKee and John Maloney. Using Statistics Gained from Corpora in a Knowledge-Based NLP System. In *Proceedings of the AAAI-92 Workshop on Statistically-Based NLP Techniques*, pages 81–89, San Jose, California, July 1992. American Association for Artificial Intelligence.

[McKeown *et al.* 1997] Kathleen R. McKeown, Shimei Pan, James Shaw, Desmond A. Jordan, and Barry Allen. Language Generation for Multimedia Healthcare Briefings. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing (ANLP-97)*, pages 277–282, Washington, D.C., April 1997.

[McMahon and Smith 1996] John G. McMahon and Francis J. Smith. Improving Statistical Language Model Performance with Automatically Generated Word Hierarchies. *Computational Linguistics*, **22**(2):217–247, June 1996.

[McQuitty 1960] L. L. McQuitty. Hierarchical Linkage Analysis for the Isolation of Types. *Educational and Psychological Measurement*, **20**:55–67, 1960.

[Mel'čuk and Pertsov 1987] Igor A. Mel'čuk and Nikolaj V. Pertsov. *Surface Syntax of English: A Formal Model within the Meaning-Text Framework*. Benjamins, Amsterdam and Philadelphia, 1987.

[Mel'čuk and Polguère 1987] Igor A. Mel'čuk and Alain Polguère. A Formal Lexicon in the Meaning-Text Theory. *Computational Linguistics*, **13**(3–4):261–275, 1987.

[Mel'čuk 1981] Igor A. Mel'čuk. Meaning-Text Models: A Recent Trend in Soviet Linguistics. In *The Annual Review of Anthropology*. 1981.

[Merialdo 1991] Bernard Merialdo. Tagging Text with a Probabilistic Model. In *Proceedings of ICASSP-91*, pages 809–812, Toronto, Canada, 1991.

[Merialdo 1994] Bernard Merialdo. Tagging English Text with a Probabilistic Model. *Computational Linguistics*, **20**(2):155–171, 1994.

[Meteer *et al.* 1991] M. Meteer, R. Schwartz, and R. Weischedel. Empirical Studies in Part of Speech Labelling. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, pages 331–336, Asilomar, California, February 1991. Morgan Kaufmann, San Mateo, California.

[Metropolis *et al.* 1953] Nicholas C. Metropolis, A. W. Rosenbluth, Marshall N. Rosenbluth, A. H. Teller, and Edward Teller. Equations of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*, **21**(6):1087–1091, June 1953.

[Miller *et al.* 1990] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to WordNet: An On-Line Lexical Database. *International Journal of Lexicography (special issue)*, **3**(4):235–312, 1990.

[Miller 1969] George A. Miller. A Psychological Method to Investigate Verbal Concepts. *Journal of Mathematical Psychology*, **6**:169–191, 1969.

[Miller 1990] Alan J. Miller. *Subset Selection in Regression.* Number 40 in Monographs on Statistics and Applied Probability. Chapman and Hall, London, 1990.

[Mitamura and Nyberg 1995] T. Mitamura and E. H. Nyberg. Controlled English for Knowledge-Based MT: Experience with the KANT System. In *Proceedings of the 6th International Conference on Theoretical and Methodological Issues in Machine Translation*, Leuven, Belgium, 1995.

[Montague 1970] Richard Montague. Universal Grammar. *Theoria*, **36**:373–398, 1970. Reprinted in [Montague 1974].

[Montague 1973] Richard Montague. The Proper Treatment of Quantification in Ordinary English. In Jaakko J. Hintikka, Julius M. E. Moravcsik, and Patrick Suppes, editors, *Approaches to Natural Language: Proceedings of the 1970 Stanford Workshop on Grammar and Semantics.* Reidel, Dordrecht, The Netherlands, 1973. Reprinted in [Montague 1974].

[Montague 1974] Richard Montague. *Formal Philosophy: Selected Papers of Richard Montague.* Yale University Press, New Haven, Connecticut, 1974. Edited and with an introduction by Richmond H. Thomason.

[Moore 1993] Johanna D. Moore, 1993. Personal communication.

[Nakatani and Hirshberg 1993] Christine Nakatani and Julia Hirshberg. A Speech-First Model for Repair Detection and Correction. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 46–53, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[National Library of Medicine 1991] National Library of Medicine, Bethesda, Maryland. *UMLS Knowledge Sources*, Second Experimental edition, 1991.

[Neumann *et al.* 1997] Günter Neumann, Rolf Backofen, Judith Baur, Marcus Becker, and Christian Braun. An Information Extraction Core System for Real World German Text Processing. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing (ANLP-97)*, pages 209–216, Washington, D.C., April 1997.

[Ogden 1932] Charles K. Ogden. *Opposition.* Indiana University Press, Bloomington, Indiana, 1932. Reprinted in 1968.

[Osgood *et al.* 1957] Charles E. Osgood, George J. Suci, and Percy H. Tannenbaum. *The Measurement of Meaning.* University of Illinois Press, Urbana, Illinois, 1957. 8th printing, 1971.

[Palmer and Day 1997] David D. Palmer and David S. Day. A Statistical Profile of the Named Entity Task. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing (ANLP-97)*, pages 190–193, Washington, D.C., April 1997.

[Papoulis 1984] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, New York, 2nd edition, 1984.

[Passonneau and Litman 1993] Rebecca J. Passonneau and Diane J. Litman. Intention-Based Segmentation: Human Reliability and Correlation with Linguistic Cues. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 148–155, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Passonneau and Litman 1997] Rebecca J. Passonneau and Diane J. Litman. Discourse Segmentation by Human and Automated Means. *Computational Linguistics*, **23**(1):103–139, March 1997.

[Pereira and Schabes 1992] Fernando Pereira and Yves Schabes. Inside-Outside Reestimation from Partially Bracketed Corpora. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 128–135, Newark, Delaware, June 1992. Association for Computational Linguistics.

[Pereira *et al.* 1993] Fernando Pereira, Naftali Tishby, and Lillian Lee. Distributional Clustering of English Words. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 183–190, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Popper 1965] Karl Raimund Popper. *Conjectures and Refutations: The Growth of Scientific Knowledge.* Basic Books, New York, 2nd edition, 1965.

[Procter 1978] Paul Procter (editor in chief). *Longman Dictionary of Contemporary English.* Longman, Essex, United Kingdom, 1978.

[Pustejovsky and Boguraev 1996] James D. Pustejovsky and Branimir Boguraev, editors. *Lexical Semantics: The Problem of Polysemy.* Oxford University Press, London and New York, 1996.

[Pustejovsky 1995] James D. Pustejovsky. *The Generative Lexicon.* MIT Press, Cambridge, Massachusetts, 1995.

[Quigley 1995] Ellie Quigley. *PERL by Example.* Prentice Hall, Englewood Cliffs, New Jersey, 1995.

[Quinlan 1986] John R. Quinlan. Induction of Decision Trees. *Machine Learning*, **1**(1):81–106, 1986.

[Quirk *et al.* 1985] Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. *A Comprehensive Grammar of the English Language.* Longman, London and New York, 1985.

[Quirk 1992] Randolph Quirk. On Corpus Principles and Design. In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of the Nobel Symposium 82*, pages 457–469. Mouton de Gruyter, Berlin and New York, 1992.

[Ramshaw and Marcus 1995] Lance A. Ramshaw and Mitchell P. Marcus. Text Chunking Using Transformation-Based Learning. In David Yarowsky and Kenneth W. Church, editors, *Proceedings of the Third Workshop on Very Large Corpora (WVLC-3)*, pages 82–94, Cambridge, Massachusetts, June 1995.

[Ravin 1993] Yael Ravin. Grammar Errors and Style Weaknesses in a Text-Critiquing System. In K. Jensen, G. Heidorn, and S. Richardson, editors, *Natural Language Processing: The PNLP Approach*, pages 65–76. Kluwer Academic Publishers, Dordrecht, The Netherlands, and Boston, Massachusetts, 1993.

[Resnik and Hearst 1993] Philip Resnik and Marti A. Hearst. Structural Ambiguity and Conceptual Relations. In *Proceedings of the ACL Workshop on Very Large Corpora (WVLC-1)*, pages 58–64, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Resnik 1993] Philip Resnik. Semantic Classes and Syntactic Ambiguity. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 278–283, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Resnik 1995] Philip Resnik. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, volume 1, pages 448–453, Montréal, Québec, Canada, August 1995. Morgan Kaufmann, San Mateo, California.

[Rich and Knight 1991] Elaine Rich and Kevin Knight. *Artificial Intelligence.* McGraw-Hill, New York, second edition, 1991.

[Richardson and Braden-Harder 1993] S. Richardson and L. Braden-Harder. The Experience of Developing a Large-Scale Natural Language Processing System: Critique. In K. Jensen, G. Heidorn, and S. Richardson, editors, *Natural Language Processing: The PNLP Approach*, pages 77–89. Kluwer Academic Publishers, Dordrecht, The Netherlands, and Boston, Massachusetts, 1993.

[Robin and McKeown 1996] Jacques Robin and Kathleen R. McKeown. Empiricially Designing and Evaluating a New Revision-Based Model for Summary Generation. *Artificial Intelligence*, **85**, August 1996.

[Romesburg 1984] H. Charles Romesburg. *Cluster Analysis for Researchers.* Wadsworth, Belmont, California, 1984.

[Rosenberg and Travis 1971] Jay F. Rosenberg and Charles Travis, editors. *Readings in the Philosophy of Language.* Prentice-Hall, Englewood Cliffs, New Jersey, 1971.

[Ross 1987] John R. Ross. Islands and Syntactic Prototypes. In Barbara Need, Eric Schiller, and Anna Bosch, editors, *Papers from the 23rd Annual Regional Meeting of the Chicago Linguistic Society (Part I: The General Session)*, pages 309–320. Chicago Linguistic Society, Chicago, Illinois, 1987.

[Roubens 1978] Marc Roubens. Pattern Classification Problems and Fuzzy Sets. *Fuzzy Sets and Systems*, **1**:239–253, 1978.

[Rousseeuw 1987] Peter J. Rousseeuw. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, **20**:53–65, 1987.

[Rubinstein 1981] Reuven Y. Rubinstein. *Simulation and the Monte Carlo Method.* Wiley, New York, 1981.

[Russell and Norvig 1995] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, Englewood Cliffs, New Jersey, 1995.

[Sadler 1989] Victor Sadler. *Working with Analogical Semantics: Disambiguation Techniques in DLT.* Foris Publications, Dordrecht, The Netherlands, and Providence, Rhode Island, 1989.

[Salton and McGill 1983] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, New York, 1983.

[Santner and Duffy 1989] Thomas J. Santner and Diane E. Duffy. *The Statistical Analysis of Discrete Data.* Springer-Verlag, New York, 1989.

[Sapir 1944] Edward Sapir. On Grading: A Study in Semantics. *Philosophy of Science*, **2**:93–116, 1944. Reprinted in [Sapir 1949].

[Sapir 1949] Edward Sapir. *Selected Writings in Language, Culture and Personality.* University of California Press, Berkeley, California, 1949. Edited by David G. Mandelbaum.

[Schütze 1992] Hinrich Schütze. Word Sense Disambiguation With Sublexical Representations. In *Proceedings of the AAAI-92 Workshop on Statistically-Based NLP Techniques*, pages 109–113, San Jose, California, July 1992. American Association for Artificial Intelligence.

[Schütze 1993] Hinrich Schütze. Word Space. In S. J. Hanson, J. D. Cowan, and C. L. Giles, editors, *Advances in Neural Information Processing Systems*, volume 5. Morgan Kaufmann, San Francisco, California, 1993. Collected papers from the 1992 IEEE Conference on Neural Information Processing Systems—Natural and Synthetic.

[Sells 1987] Peter Sells. *Lectures on Contemporary Syntactic Theories.* Number 3 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, California, 1987.

[Shannon 1948] Claude E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423 and 623–656, July and October 1948.

[Sharman *et al.* 1990] R. A. Sharman, Fred Jelinek, and Robert L. Mercer. Generating a Grammar for Statistical Training. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 267–272, Hidden Valley, Pennsylvania, June 1990. Defense Advanced Research Projects Agency, Information Science and Technology Office, Morgan Kaufmann, San Mateo, California.

[Siegel 1997a] Eric V. Siegel. Learning Methods for Combining Linguistic Indicators to Classify Verbs. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, Providence, Rhode Island, August 1997.

[Siegel 1997b] Eric V. Siegel. *Linguistic Indicators for Language Understanding: Using Machine Learning Methods to Combine Corpus-Based Indicators for Aspectual Classification of Clauses.* PhD thesis, Department of Computer Science, Columbia University, New York, October 1997.

[Silverman 1986] Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, London, 1986.

[Sinclair and Kirby 1990] John M. Sinclair and D. M. Kirby. Progress in English Computational Lexicography. *World Englishes*, **9**:21–36, 1990.

[Sinclair 1987a] John M. Sinclair, editor. *Looking up: An Account of the COBUILD Project in Lexical Computing.* Collins, London, 1987.

[Sinclair 1987b] John M. Sinclair (editor in chief). *Collins COBUILD English Language Dictionary.* Collins, London, 1987.

[Smadja *et al.* 1996] Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. Translating Collocations for Bilingual Lexicons: A Statistical Approach. *Computational Linguistics*, **22**(1):1–38, March 1996.

[Smadja 1991] Frank Smadja. *Extracting Collocations from Text. An Application: Language Generation.* PhD thesis, Department of Computer Science, Columbia University, New York, 1991.

[Smadja 1993] Frank Smadja. Retrieving Collocations from Text: Xtract. *Computational Linguistics*, **19**(1):143–177, March 1993.

[Sneath and Sokal 1973] Peter H. A. Sneath and Robert R. Sokal. *Numerical Taxonomy: The Principles and Practice of Numerical Classification.* W. H. Freeman, San Francisco, California, 1973.

[Sneath 1957] Peter H. A. Sneath. Some Thoughts on Bacterial Classifications. *Journal of General Microbiology*, **17**:184–200, 1957.

[Sokal and Michener 1958] Robert R. Sokal and Charles D. Michener. A Statistical Method for Evaluating Systematic Relationships. *University of Kansas Science Bulletin*, **38**:1409–1438, 1958.

[Sokal and Sneath 1963] Robert R. Sokal and Peter H. A. Sneath. *Principles of Numerical Taxonomy*. W. H. Freeman, San Francisco, California, 1963.

[Sörensen 1948] Thorvald J. Sörensen. A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and its Application to Analysis of the Vegetation of Danish Commons. *Biologiske Skrifter*, **5**(4):1–34, 1948.

[Sparck-Jones 1986] Karen Sparck-Jones. *Synonymy and Semantic Classification*. Edinburgh University Press, Edinburgh, Great Britain, 1986. Based on the author's Ph.D. thesis, University of Cambridge, 1964.

[Späth 1985] Helmuth Späth. *Cluster Dissection and Analysis: Theory, FORTRAN Programs, Examples*. Ellis Horwood, Chichester, West Sussex, England, 1985.

[Spearman 1904] Charles E. Spearman. The Proof and Measurement of Association between Two Things. *American Journal of Psychology*, **15**(1):72–101, January 1904.

[Spearman 1906] Charles E. Spearman. A Footrule for Measuring Correlation. *British Journal of Psychology*, **2**:89ff, 1906.

[Su *et al.* 1994] Keh-Yih Su, Ming-Wen Wu, and Jing-Shin Chang. A Corpus-based Approach to Automatic Compound Extraction. In *Proceedings, 32nd Annual Meeting of the ACL*, pages 242–247, Las Cruces, New Mexico, June 1994. Association for Computational Linguistics.

[Sundheim and Chinchor 1993] Beth M. Sundheim and Nancy A. Chinchor. Survey of the Message Understanding Conferences. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 56–60, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Sundheim 1992] Beth M. Sundheim. Overview of the Fourth Message Understanding Evaluation and Conference. In *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pages 3–21, McLean, Virginia, June 1992. DARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Mateo, California.

[Swets 1969] J. A. Swets. Effectiveness of Information Retrieval Methods. *American Documentation*, **20**:72–89, January 1969.

[Szu and Hartley 1987] H. Szu and R. Hartley. Fast Simulated Annealing. *Physics Letters*, **122**(3, 4):157–162, 1987.

[Tarski 1935] Alfred Tarski. Der Wahrheitsbegriff in den Formalisierten Sprachen. *Studia Philosophica*, **1**:261–405, 1935. English translation in [Tarski 1956].

[Tarski 1956] Alfred Tarski. *Logic, Semantics and Metamathematics*. Oxford University Press, London, 1956.

[Thorndike and Lorge 1944] Edward L. Thorndike and Irving Lorge. *The Teacher's Word Book of 30,000 Words.* Teacher's College, Columbia University, New York, 1944.

[Tichy *et al.* 1995] Walter F. Tichy, Paul Lukowicz, Lutz Prechelt, and Ernst A. Heinz. Experimental Evaluation in Computer Science: A Quantitative Study. *Journal of Systems and Software*, **28**(1):9–18, January 1995.

[Trier 1934] Jost Trier. Das Sprachliche Feld. Eine Auseinandersetzung. *Neve Jahrbücher für Wissenschaft und Jugendbildung*, **10**:428–449, 1934.

[Trubetzkoy 1939] Nikolai S. Trubetzkoy. *Grundzuger der Phonologie.* Travaux du Cercle Linguistique de Prague 7, Prague, 1939. English translation in [Trubetzkoy 1969].

[Trubetzkoy 1969] Nikolai S. Trubetzkoy. *Principles of Phonology.* University of California Press, Berkeley and Los Angeles, California, 1969. Translated into English from [Trubetzkoy 1939].

[Tsang 1993] Edward Tsang. *Foundations of Constraint Satisfaction.* Academic Press, London and San Diego, California, 1993.

[Tzoukermann and Radev 1996] Evelyne Tzoukermann and Dragomir R. Radev. Using Word Class for Part-of-Speech Disambiguation. In *Proceedings of the Fourth Workshop on Very Large Corpora (WVLC-4)*, pages 1–13, Copenhagen, Denmark, August 1996.

[Ullmann 1957] Stephen Ullmann. *The Principles of Semantics.* Blackwell, Oxford, 2nd edition, 1957.

[United States National Center for Health Statistics 1980] United States National Center for Health Statistics, Washington, D.C. *International Classification of Diseases*, Ninth Revision, with Clinical Modifications, 1980.

[van der Eijk *et al.* 1996] Pim van der Eijk, M. de Koning, and G. van der Steen. Controlled Language Correction and Translation. In *Proceedings of the First International Workshop on Controlled Language Applications*, pages 64–73, Katholieke Universiteit Leuven, Belgium, 1996.

[van der Eijk 1993] Pim van der Eijk. Automating the Acquisition of Bilingual Terminology. In *Proceedings of the Sixth Conference of the European Chapter of the ACL*, pages 113–119, Utrecht, The Netherlands, April 1993. Association for Computational Linguistics.

[van Laarhoven 1988] Peter J. M. van Laarhoven. *Theoretical and Computational Aspects of Simulated Annealing.* Centrum voor Wiskunde en Informatica, Amsterdam, 1988.

[van Rijsbergen 1979] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 2nd edition, 1979.

[Vendler 1967] Zeno Vendler. Verbs and Times. *Linguistics in Philosophy*, pages 97–121, 1967.

[Veronis and Ide 1990] Jean Veronis and Nancy M. Ide. Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionaries. In Hans Karlgren, editor, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, volume 2, pages 389–394, Helsinki, Finland, 1990.

[Viterbi 1967] Andrew J. Viterbi. Error Bounds for Convolution Codes and an Asymptotically Optimal Decoding Algorithm. *IEEE Transactions on Information Theory*, **13**:260–269, 1967.

[von Humboldt 1836] Wilhelm F. von Humboldt. *Über die Verschiedenheit des Menschlichen Sprachbaues.* Berlin, 1836. Reprinted in 1949 by Claasen and Roether, Darmstadt.

[Voutilainen 1993] Atro Voutilainen. NPTool, a Detector of English Noun Phrases. In *Proceedings of the ACL Workshop on Very Large Corpora (WVLC-1)*, pages 48–57, Columbus, Ohio, June 1993. Association for Computational Linguistics.

[Wacholder *et al.* 1997] Nina Wacholder, Yael Ravin, and Misook Choi. Disambiguation of Proper Names in Text. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing (ANLP-97)*, pages 202–208, Washington, D.C., April 1997.

[Wahba 1990] Grace Wahba. *Spline Models for Observational Data.* CBMS-NSF Regional Conference series in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pennsylvania, 1990.

[Waibel and Lee 1990] Alex Waibel and Kai-Fu Lee, editors. *Readings in Speech Recognition.* Morgan Kaufmann, San Mateo, California, 1990.

[Walker and Hockney 1991] Donald E. Walker and Susan Hockney. The Text Encoding Initiative. *Bulletin du CID*, 1991.

[Walker 1987] Donald E. Walker. Knowledge Resource Tools for Accessing Large Text Files. In Amedeo Cappelli, Laura Cignoni, and Carol Peters, editors, *Studies in Honour of Roberto Busa*, pages 279–300. Giardini, Pisa, Italy, 1987.

[Wall *et al.* 1996] Larry Wall, Tom Christiansen, and Randal L. Schwartz. *Programming Perl.* Nutshell Handbooks. O'Reilly and Associates, Sebastopol, California, 2nd edition, 1996.

[Ward 1963] Joe H. Ward, Jr. Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association*, **58**(301):236–244, March 1963.

[Waugh 1982] Linda R. Waugh. Marked and Unmarked: A Choice Between Unequals in Semiotic Structure. *Semiotica*, **38**(3–4):299–318, 1982.

[White 1988] J. S. White. Determination of Lexical-Semantic Relations for Multi-Lingual Terminology Structures. In Martha W. Evens, editor, *Relational Models of the Lexicon: Representing Knowledge in Semantic Networks.* Cambridge University Press, Cambridge, England, 1988.

[Whittemore *et al.* 1990] Greg Whittemore, Kathleen Ferrara, and Hans Brunner. Empirical Study of Predictive Powers of Simple Attachment Schemes for Post-Modifier Prepositional Phrases. In *Proceedings of the 28th Annual Meeting of the ACL*, pages 23–30, Pittsburgh, Pennsylvania, June 1990. Association for Computational Linguistics.

[Wilson 1987] Michael Wilson. MRC Psycholinguistic Database: Machine Usable Dictionary. Version 2.00. Technical Report IKBS/SN16/86, Informatics Division, Science and Engineering Research Council, Rutherford Appleton Laboratory, Oxon, England, April 1987.

[Winters 1990] Margaret Winters. Toward a Theory of Syntactic Prototypes. In Savas L. Tsohatzidis, editor, *Meanings and Prototypes: Studies in Linguistic Categorization*, pages 285–307. Routledge, London, 1990.

[Wittgenstein 1972] Ludwig Wittgenstein. *Philosophical Investigations.* Macmillan, New York, 3rd edition, 1972. Translated from *Philosophische Untersuchungen.*

[Wojcik *et al.* 1990] Richard H. Wojcik, J. E. Hoard, and K. Holzhauser. The Boeing Simplified English Checker. In *Proceedings of the International Conference on Human Machine Interaction and Artificial Intelligence in Aeuronatics and Space*, pages 43–57, Toulouse, France, 1990. Centre d'Etudes et de Recherches de Toulouse.

[Yamron *et al.* 1994] Jonathan Yamron, James Cant, Anne Demedts, Taiko Dietzel, and Yoshiko Ito. The Automatic Component of the LINGSTAT Machine-Aided Translation System. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 163–168, Plainsboro, New Jersey, March 1994. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Yarowsky 1993] David Yarowsky. One Sense per Collocation. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 266–271, Plainsboro, New Jersey, March 1993. ARPA Software and Intelligent Systems Technology Office, Morgan Kaufmann, San Francisco, California.

[Yarowsky 1994] David Yarowsky. Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In *Proceedings of the 32nd Annual Meeting of the ACL*, pages 88–95, Las Cruces, New Mexico, June 1994. Association for Computational Linguistics.

[Yates 1934] Frank Yates. Contingency Tables Involving Small Numbers and the χ^2 Test. *Journal of the Royal Statistical Society, Supplement*, **1**:217–235, 1934.

[Zhou and Dapkus 1995] Joe Zhou and Pete Dapkus. Automatic Suggestion of Significant Terms for a Predefined Topic. In David Yarowsky and Kenneth W. Church, editors, *Proceedings of the Third Workshop on Very Large Corpora (WVLC-3)*, pages 131–147, Cambridge, Massachusetts, June 1995.

[Zipf 1945] George K. Zipf. The Meaning-Frequency Relationship of Words. *Journal of General Psychology*, **3**:251–256, 1945.

[Zipf 1949] George K. Zipf. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology.* Addison-Wesley, Reading, Massachusetts, 1949.

[Zubin 1938] T. Zubin. A Technique for Measuring Like-Mindedness. *Journal of Abnormal and Social Psychology*, **33**:508–516, 1938.

[Zwicky 1978] A. Zwicky. On Markedness in Morphology. *Die Sprache*, **24**:129–142, 1978.

Appendix

Data extraction

A.1 Introduction

The present appendix discusses in detail the data extraction mechanisms used by the lexical semantic knowledge acquisition system I presented in the body of the dissertation. That system extracts lexical semantic information from the text, identifying groups of semantically related words, scalar groups, antonyms and synonyms, and assigning semantic orientation and markedness values to each scalar word. The system operates on free text in a completely automatic and unsupervised manner, using a combination of statistical methods guided by linguistic principles and shallow linguistic knowledge, and produces solutions that are adapted to the particular domain from which the training corpus has been drawn.

In Chapters 2 and 6–8 I presented several syntagmatic relationships, linguistic properties, and text-level indicators for them that are useful for the problems of lexical semantic knowledge acquisition addressed by the system and briefly outlined above. In this appendix, I discuss in more detail the mechanisms by which this linguistic knowledge, consisting of syntagmatic and morphological relationships between words and information on syntactic and morphological features of words, is extracted from free text.

In the next section, I discuss the corpora that I have used and explain why I selected them. Then, I address issues related to the preprocessing of the text, namely how sentence and word boundaries are identified, how parts of speech are automatically assigned to each word, and how the output of the tagging phase is filtered and transformed to increase its quality. In Section A.4 I present in detail a grammar that retrieves the syntagmatic relationships between words that are used by the various stages of the system. I also discuss how ambiguities in the grammar are resolved by the application of heuristic rules, and describe the actual implementation of the grammar via a collection of nested finite state automata. Methods for processing the output of this parser to increase its accuracy and combine instances of syntagmatic relationships that are only superficially different are discussed in Section A.5. Finally, in the last section I describe how additional text-level indicators, needed for the separation of words into subscales and for the determination of semantic markedness, are retrieved from the corpus.

A.2 The corpora

During the development of the system I have used two corpora. The first is a collection of stock market reports from the Associated Press newswire during the period 1987–1988, containing approximately 8.2 million words.[1] Unfortunately, it also contains some repeated text (multiple occurrences of articles), which adversely affect the statistical methods used. This corpus was processed by an earlier version of the preprocessing and extracting tools described in the next sections, and was also parsed using Donald Hindle's robust parser FIDDITCH [Hindle 1983; Hindle 1994].[2]

Because of the repetitions that were present in this corpus, and the limitations of its size, I later switched to a different, larger, and cleaner corpus. I used this corpus for developing and evaluating the current version of the word grouping subsystem, as well as for training and testing the modules that subsequently analyze the retrieved groups. This corpus is the September 1991 CD-ROM version of the Wall Street Journal (WSJ) articles from the years 1987 to 1989[3], provided by the Data Collection Initiative of the Association for Computational Linguistics (ACL/DCI CD-ROM 1). I have relied mostly on the 1987 articles, which contain approximately 21 million words, but I have preprocessed and used for some of the analyses the 1988 articles as well, for an additional 23.5 million words. The Wall Street Journal corpus offers the following advantages:

- Relatively high quality and low error rate.
- Extensibility in terms of size if this becomes necessary, since we can always add additional years of Wall Street Journal articles. In fact, three more years are already

[1] I thank Frank Smadja and Karen Kukich for making this corpus available to me.

[2] I thank Diane Litman and Donald Hindle for arranging for the parsing of the corpus at AT&T Bell Labs.

[3] © 1987, 1988, 1989 Dow Jones Inc.

available from ACL/DCI in a slightly different format.

- Some preprocessing already done at the University of Pennsylvania and incorporated in the version of the corpus that ACL/DCI distributes. This includes separating datelines, headlines, and other information belonging to the article's header from each article's body, and marking sentence and paragraph breaks.

- A clearly defined SGML-like markup language.

- Wide availability at a low cost, which has made this corpus a de facto standard in the computational linguistics community. The availability of a common corpus guarantees that reported results can be independently verified, and that different approaches to the same problem can be compared on the same data.

In addition to the Wall Street Journal corpus, I have also partially processed and intend to use in the future a collection of medical discharge summaries from the Columbia Presbyterian Medical Center (CPMC). Each discharge summary contains a synopsis of a patient's course throughout the hospital, including descriptions of past medical history, diagnosis, administered treatment, overall result, and recommended follow-up procedures. The current collection at my disposal consists of 223,503 summaries containing a total of approximately 121 million words. I have focused on the subcorpus of cardiology reports which provides text from a more specialized domain; this part consists of 7,374 discharge summaries (about 4 million words).[4]

[4]The number of words mentioned above for the medical corpora includes headers that are removed during preprocessing, as discussed in Section A.3.4. Thus, fewer words than the listed number are actually available as part of the useful text.

The medical corpus is converted to a format similar to that used in the Wall Street Journal corpus through the application of a preprocessing front end I developed, before it is presented to the semantic knowledge extraction system. It is also sanitized with the application of a separate tool [Hatzivassiloglou and McKeown 1998], removing sensitive information such as patient and doctor names, operation dates, and addresses. In the future, I intend to collect semantic groups of nouns from this corpus with the methods described in Chapter 2, and incorporate this lexical semantic knowledge into the lexicon of the MAGIC multimedia generation and co-ordination system [Dalal *et al.* 1996a; Dalal *et al.* 1996b; McKeown *et al.* 1997] (see Section 11.2).

All the above corpora are used as sources of cooccurrence information for the probabilistic methods described in Chapters 2 and 6–9, as well as for the comparative study of Chapter 5. I have also used a separate corpus to draw data for exploratory analyses. The corpus I used for this purpose is the Brown University corpus[5] [Kučera and Francis 1967; Francis and Kučera 1982]. While the Brown corpus contains only about 1 million words, it is carefully designed and balanced across several genres of written text, thus making any results established from it plausible for the language in general and not dependent on a particular domain. In Chapter 8 I discussed how data from this corpus was used to quantitatively evaluate some of the tests proposed in the linguistics literature for determining semantic markedness.

[5]Officially called the Standard Corpus of Present-Day Edited American English.

A.3 Preprocessing the corpus

Before turning my attention on how to extract from the corpus the pairs of words that participate in the various relationships that were identified in Sections 2.1–2.2, 6.2, 7.3, and 8.2, I need to consider how to process the raw corpus so that extraction methods become applicable. This preprocessing phase encompasses several issues, on which I elaborate below.

A.3.1 Identifying encoding conventions in the machine-readable text

A machine-readable text corpus can rarely be used "off the shelf" for data extraction tasks. While the text itself is usually represented by the standard 7-bit ASCII code [American National Standards Institute 1986], which is almost identical to the international standard character set [International Standards Organization 1983], frequently other symbols are interspersed in it. This is a result of adopting a particular set of text formatting conventions or of some preprocessing already performed by the people who make the machine-readable version available. For example, Table A.1 shows some of the symbol sequences that have special meaning in the standard version of the Brown corpus and Figure A.1 shows an excerpt from the CD-ROM version of the Wall Street Journal corpus. Note that the Wall Street Journal example includes markup symbols such as <*s*>–</*s*> and <*p*>–</*p*> pairs, which delimit sentences and paragraphs respectively.

In general, the meaning of some non-alphanumeric characters and character sequences in

Symbol sequence	Meaning
**A	Apostrophe
**F	Mathematical formula
**H	Ellipsis (...)
**Q	Beginning of quoted material
**U	End of quoted material
*=	Beginning of text region that was emphasized in the original source
*$	End of text region that was emphasized in the original source

Table A.1: Some of the encoding conventions used in the standard version of the Brown corpus.

the corpus is not always clear. For example, a newline character may be used to indicate the end of a sentence, or the end of the line in some printed edition of the text, or have no special meaning at all but simply be placed at random word boundaries or after a fixed number of words or characters to prevent extremely long lines. Conversely, different means may be employed to indicate the end of a sentence (if such information is explicitly present in the corpus at all), e.g., a special tag as in the Wall Street Journal corpus, a newline character as in some versions of the Brown corpus, or a null byte (ASCII 0). Extra-linguistic, visual information such as paragraph and page breaks which is immediately available to the reader of the printed version of the corpus may be removed from the machine-readable version, or encoded with special ad hoc character sequences or with the use of blank lines.

Usually such conventions are described in the documentation that accompanies the text files, but even then a new front end needs to be implemented for each new raw text format, to translate the raw text in a common internal format that is expected by subsequent stages. While UNIX tools such as *awk* [Aho *et al.* 1988] and *lex* [Lesk and Schmidt 1975; Levine *et al.* 1992], and programming languages with built-in pattern matching capabilities such

```
<DOC>
<DOCNO> 870513-0054. </DOCNO>
<HL> Weather Threatens U.S., Soviet Grain
---
Conditions May Soon
Boost Commodity Prices
---
By Jean Marie Brown and Wendy L. Wall
Staff Reporters of The Wall Street Journal</HL>
<DD> 05/13/87</DD>
<SO> WALL STREET JOURNAL (J)</SO>
<IN> FREST GOVMT
COMMODITY NEWS, FARM PRODUCTS (CMD)
MONETARY NEWS, FOREIGN EXCHANGE, TRADE (MON) </IN>
<TXT>
<p>
<s> Warm, dry weather in some U.S. agricultural regions and a wet spring in the Soviet
Union are threatening grain crops in both countries and may push commodity prices
higher in coming weeks. </s>
</p>
<p>
<s> The dry Midwestern spring, coming on the heels of an unusually mild winter, has
severely depleted topsoil moisture in some areas, setting the stage for a summer drought.
</s>
<s> Although it is far too early in the season to make firm predictions, much of Illinois
and northern Iowa -- the heart of the Corn Belt -- in the past six weeks has received less
than half of the rainfall normal for this time of year. </s>
<s> Most vulnerable are corn and soybeans, the nation's two most valuable crops with a
combined projected value this year of more than $23 billion. </s>
</p>

...

The article includes 16 more paragraphs, containing 42 additional sentences, and a table.
These have been removed from the figure because of space restrictions.

</TXT>
</DOC>
```

Figure A.1: Excerpt from the Wall Street Journal corpus.

as *perl* [Quigley 1995; Wall *et al.* 1996], can help with this process, introducing a new raw text format nevertheless involves a considerable amount of work. In the present system, I have already implemented two such front ends for two different formats, and I am in the midst of developing a third front end for use with the medical discharge summaries from which relationships between medical terms will be extracted (see Section 11.2). This need for customizing to different text formats would be eliminated if the text providers adhered to a common standard. Steps for the development of such a standard are being taken by the Text Encoding Initiative (TEI) [Walker and Hockney 1991], which uses the formally defined Standard Generalized Markup Language (SGML) [International Standards Organization 1986]. However, to the date of this writing, no major corpora employ a TEI-conformant markup language (although some, like the ACL/DCI Wall Street Journal corpus, use SGML variants for their mark-up conventions).

A.3.2 Separating the corpus into sentences

Several punctuation marks, usually periods, question marks, and exclamation marks, are used to delimit sentences; other symbols such as dashes, semicolons, and colons partition a sentence into more or less independent segments. However, identifying the cases that these punctuation marks are used as sentence terminators is a non-trivial task. Periods, which are by far the most frequent sentence terminator[6], are also employed for different reasons in the text. These other uses include abbreviations (*U.S.A.*, *T. S. Elliot*, *etc.*, *Dr.*

[6] Of the 869,992 sentences automatically identified in the 1987 Wall Street Journal corpus with the methods of the present section, 857,375 (98.55%) are terminated with a period. 9,017 (1.04%) of these sentences are terminated with a question mark and 1,362 (0.16%) with an exclamation mark. The remaining 2,238 cases (0.26%) correspond to errors made during the identification of sentence breaks or to improperly terminated sentences. The percentages given do not sum exactly to 100% because of rounding.

Frankestein), fractional parts of numbers (*0.2*), list items (*1.*), and ellipsis (..., three dots that can end a sentence absorbing the final period, or appear in the middle of the sentence to indicate hesitation, a pause in speech, etc.). Dashes present similar problems for the task of identifying relatively independent subsentential fragments.[7] Sometimes, interpreting a period correctly can be very hard for an automatic system that does not use syntax or semantics, as the following example shows:

I thought the book on C. Marvin was helpful.	*one sentence*
I lost the book on C. Marvin was helpful.	*two shorter sentences*

In the above example, the subcategorization of *think* for a clause complement forces the period to be interpreted as an abbreviation mark. Since *lose* cannot have a clause complement, in the second case the period has to be a sentence terminator.[8]

Sentence separation also depends on what information is already encoded in the markup language used in the machine-readable version of the corpus. If the corpus already contains explicit information about sentence breaks, as the Wall Street Journal corpus does, I use a simple filter that extracts this information and converts it to the uniform, corpus-independent format that I use for subsequent steps. Otherwise, I employ the algorithm for sentence boundary detection that comes as part of the part-of-speech tagger [Church 1988] that I use at a later stage.

---

[7]In typography, dashes of different length are used according to the purpose the dash serves, e.g., - (an interword dash as in *top-down approach*), – (separating ranges of numbers, e.g., 12–15), and — (the dash as a punctuation symbol separating sentence segments). Unfortunately, this information is not usually preserved in the machine-readable version of the corpus.

[8]Note the difference in the space left after the period, which in carefully typeset text is wider in the second case. Unfortunately, once again this information is not usually present in the machine-readable version of the corpus.

A.3.3 Separating the sentences into words

In English, two closely related words are sometimes joined in special contracted word forms, such as *can't* and *I'll.* If such word forms are left intact in the corpus, two problems arise: First, the complexity of the part-of-speech tagging task increases, since these forms must be assigned composite tags such as *PPSS+MD* for *I'll* (personal pronoun + modal). This would increase the number of valid tags and the complexity of the search for the optimal tag sequence. Second, the use of composite tags in the output of the tagger complicates the design of the syntactic extractor modules which analyze this output, as different rules need to be used for handling *I'll* and *I will.*

For these reasons, I have built a filter program that retrieves such contracted word forms and replaces them with the sequence of the individual words that make up the contracted form. This filter also replaces abbreviated forms of a single word (such as *'n* for *and*), and separates pairs of words that are joined together without an apostrophe (such as *cannot*).

A.3.4 Further preprocessing before part-of-speech tagging

Before the corpus is passed on to the part-of-speech tagger that will assign a unique grammatical category to each word, I employ several additional filters that remove or repair unwanted or problematic parts of the text. These filters first strip text that does not belong to the main body of an article, but is part of a figure or table, or of a title, date, source, article number, or keywords line. These parts of an article use a highly compact, elliptic syntax that does not fully conform to the general syntactic rules of English [Quirk *et al.* 1985, pp. 845–846], and is not likely to contain the relationships I am looking for. Leaving these parts in the corpus would almost certainly result in misclassification errors by the tagger and/or the subsequent extractor of word pairs.

The filters also remove quote markers from the text, leaving the quoted material in place. This does not affect the grammaticality of the text, but avoids having the quote symbols processed by the tagger (which has great difficulty with them) and simplifies the output so that the grammar which subsequently processes the tagged text will not have to worry about quoted text. Implementing this is easy for the “, ”, ", and ‘ forms of quotes, but hard for ’ which can also be used as an apostrophe or possessive marker. Several rules are employed for appropriately classifying occurrences of ’ according to its position in the word and the surrounding context.

Finally, sequences of two or more periods that signify an ellipsis are removed. Ellipsis markers are not informative for our purposes, and their removal seldom affects the grammaticality of the text.[9] If left in the text, the tagger will always interpret them as sentence-final punctuation, which is not necessarily true. Once again the problem is not trivial, since ellipsis may absorb adjacent punctuation symbols. Rules are used to detect such cases and reinsert the original punctuation symbol (which I assume was a period) in the text.

[9]Only if they are used as an apposition separator, or to indicate a repair in transcribed speech. In the latter case, the sentence is ungrammatical anyway.

A.3.5 Assigning parts-of-speech to words

In order to be able to extract pairs of words based on their grammatical categories, I need a means to identify these categories (parts-of-speech) in the corpus. The problem is hard since many words (and especially the most frequent ones) may appear with several parts of speech. For example, *light* can be used as adjective, noun, or verb depending on the context. To identify the appropriate part of speech for each word in the corpus, I use a *probabilistic part of speech tagger*, which assigns the most likely grammatical category for each word occurrence.

Such part-of-speech taggers [Church 1988; de Rose 1988; de Marcken 1990; Merialdo 1991; Meteer *et al.* 1991; Kupiec 1992; Cutting *et al.* 1992; Charniak *et al.* 1993; Merialdo 1994; Tzoukermann and Radev 1996] access a *language model* which consists of probabilities for any short sequence of words, typically bigrams or trigrams. These probabilities are estimated from a corpus that has been tagged with part-of-speech information by hand, or with a hidden Markov model (HMM) approach [Kupiec 1992]. The initial estimates are also typically smoothed, to guard against sparse data problems for rare words and word sequences.[10] At the end of the training phase, the probabilities are known to the tagger. When new text is presented to it, the tagger tries to assign valid part-of-speech tags to each word in the text in such a way that the probability of the sequence of tags given the words is maximized. Because this probability cannot be directly computed for arbitrarily large word sequences, the tagger approximates the stochastic processes that generates strings of

[10]See, for example, [Jelinek and Mercer 1980], [Church and Gale 1991], [Hatzivassiloglou and Knight 1995], and [Chen and Goodman 1996] for a discussion of quite sophisticated smoothing methods.

words with a low order Markov chain. The precomputed bigram or trigram conditional probabilities then give the transition probabilities of the chain. The most likely sequence of states of the chain is found with an iterative algorithm such as the Viterbi algorithm [Viterbi 1967] or the N-best algorithm [Chow and Schwartz 1989]. This sequence immediately gives the best (in the maximum likelihood sense) assignment of tags, since a part-of-speech tag is associated with each state.

I use such a part-of-speech tagger, designed by Ken Church at AT&T Bell Labs[11] [Church 1988]. The tagger uses a trigram model with simple smoothing based on a word list and typically produces output that is 95–97% accurate. The tagger assigns to each word in the text one of the tags of Table A.2, in a way that maximizes the probability of the tag sequence assigned to each sentence in the corpus. It should be noted here that the decision to use such a tool is consistent with the stated goals of full automation and avoidance of dependencies on precompiled lexical information (Section 1.5). Furthermore, this method is quite efficient, tagging about 126,000 words per minute on a Sun SparcStation 10.[12]

[11] I wish to thank Ken Church for making the tagger available to Columbia University, especially for providing the source code. This allowed me to tinker with it, incorporating the filters that were discussed in Sections A.3.2–A.3.4.

[12] The tagging phase takes about 4.76 seconds per 10,000 words; an additional 0.58 seconds per 10,000 words is spent on the preprocessing and postprocessing steps of Sections A.3.2, A.3.3, A.3.4, and A.3.6.

Tag	Meaning	Examples
ABN	Predeterminer	*all, half*
AT	Article	*a, the, no*
CD	Cardinal number	*three, 41*
DT	Singular demonstrative pronoun	*this, that*
DTS	Plural demonstrative pronoun	*these, those*
DTI	Indefinite pronoun/determiner	*some, any*
DTX	"Dual" indefinite pronoun, i.e. involving two phrases	*both, either*
NN	Singular noun	*correction, maturity*
NNS	Plural noun	*investors, people*
NP	Proper noun	*Korea, IRS*
JJ	Adjective	*final, usual*
EX	Existential *there*	*there*
PN	Compound indefinite pronoun	*something, anyone*
PPS	Personal pronoun, subjective case, singular	*she, it*
PPSS	Personal pronoun, subjective case, plural	*you, they*
PPO	Personal pronoun, objective case	*him, it, them*
PP$	Personal pronoun, genitive case[13]	*his, their*
PPL	Reflexive pronoun, singular	*myself, herself*
PPLS	Reflexive pronoun, plural	*ourselves, themselves*
QL	Qualifier	*more, so, too*
RB	Adverb	*ahead, generally*
WDT	Relative/interrogative pronoun/determiner	*what, whichever*
WPS	Relative/interrogative personal pronoun, subjective or objective case	*who, whom, that*
WP$	Relative/interrogative personal pronoun, genitive case	*whose*
WRB	Relative/interrogative adverb	*when, where, why*
CC	Coordinating conjunction	*and, but*
CS	Subordinating conjunction	*as, because, since*
IN	Preposition	*at, for, to*
TO	*to* introducing an infinitive	*to*
UH	Interjection	*ah, oops*

Table A.2: Word tags employed by the part-of-speech tagger. Continued on the next page.

[13] This tag does not belong in the set originally used by the tagger, but is introduced later during the post-processing of the tagged corpus; see Section A.3.6.

Tag	Meaning	Examples
MD	Modal verb	*may*, *should*, *will*
VB	Uninflected verb (infinitive, or present tense but not third person singular)[14]	*walk*, *break*
VBZ	Verb, third person singular[15]	*walks*, *breaks*
VBD	Verb, past tense[16]	*walked*, *broke*
VBG	Verb, present participle	*walking*, *breaking*
VBN	Verb, past participle[17]	*walked*, *broken*
$	Possessive marker	*'s*, *'*
(	Punctuation marking the start of a parenthetical unit	*(*, *[*, {
)	Punctuation marking the end of a parenthetical unit	*)*, *]*, }
,	Comma	*,*
:	Punctuation separating a sentence into independent clauses	*:*, *;*
.	Punctuation marking the end of a sentence	*.*, *!*, *?*

Table A.2: Word tags employed by the part-of-speech tagger. Continued from the previous page.

A.3.6 Post-processing the tagger's output

A sequence of filters postprocess the tagger's output, to remedy some mistakes that are consistently made during the tagging process. First, words belonging in some categories that are handled in a special way at two separate places within the tagger come out with a double tag (*/NP/NP* for proper nouns and */$/$* for the possessive marker *'* after nouns ending in *s*). A filter removes the extraneous tag, to simplify the algorithm that extracts the tag from a tagged word. The filter also handles a related problem, involving words that

[14] Additional special tags exist for the three primary verbs of English: BE for *be*, DO for *do*, and HV for *have*. *Be* is the only English verb where the infinitive is different from the simple present in number and person combinations other than the third person singular. Consequently, there are the specialized tags BEM for *am* and BER for *are*.

[15] Additional special tags: BEZ for *is* and HVZ for *has*.

[16] Additional special tags: BED for *were*, BEDZ for *was*, and HVD for *had*.

[17] A special tag, BEN, is assigned to *been*.

contain a period. This period is placed by the tagger after the tag, as in *Mr/NP./NP*; the filter fixes the problem by putting these periods in their correct place, producing *Mr./NP*.

Another filter fixes some consistenly wrong assignments of tags to words which I identified by inspecting the output of the tagger. These include the assignment of the */JJ* (adjective) tag to possessive pronouns, a shortcoming of the tag set which I fixed by introducing a new tag */PP$*; the misclassification of *'s* as a possessive marker instead of an auxiliary verb (*is* or *has*) even in contexts that preclude the former interpretation, e.g., after a subjective case personal pronoun such as *he*; and the mysterious assignment of the tag */ILLEGAL* for plural nouns ending in *-ians* such as *politicians* and *Australians*. Also, some tag assignments that are not technically wrong but can be replaced with more informative specialized tags are changed by the filter. These include changing the tag of *less*, *more*, *least*, and *most* from adjective to adverb when the word is followed by an adjective or adverb, and compacting special noun phrases such as *a bit* and *a little* that function as adverbs. Finally, interjections such as *ah* and *oh* are removed from the tagged text, since they offer no information for our purposes.

Figure A.2 shows the result of applying the preprocessing, part-of-speech tagging, and postprocessing steps discussed above to the sample extract of the Wall Street Journal corpus that was given in Figure A.1.

```
.Start of File 1/.
.Start of File 2/.
.Start of File 3/.
.Start of Sentence/.
Warm/JJ ,/, dry/JJ weather/NN in/IN some/DTI U.S./NP agricultural/JJ
regions/NNS and/CC a/AT wet/JJ spring/NN in/IN the/AT Soviet/NP Union/NP
are/BER threatening/VBG grain/NN crops/NNS in/IN both/DTX countries/NNS
and/CC may/MD push/VB commodity/NN prices/NNS higher/JJ in/IN coming/VBG
weeks/NNS ./.
.End of Sentence/.
.Start of Sentence/.
The/AT dry/JJ Midwestern/NN spring/NN ,/, coming/VBG on/IN the/AT heels/NNS
of/IN an/AT unusually/RB mild/JJ winter/NN ,/, has/HVZ severely/RB
depleted/VBN topsoil/NN moisture/NN in/IN some/DTI areas/NNS ,/, setting/VBG
the/AT stage/NN for/IN a/AT summer/NN drought/NN ./.
.End of Sentence/.
.Start of Sentence/.
Although/CS it/PPS is/BEZ far/RB too/QL early/RB in/IN the/AT season/NN
to/TO make/VB firm/JJ predictions/NNS ,/, much/JJ of/IN Illinois/NP and/CC
northern/JJ Iowa/NP -/: the/AT heart/NN of/IN the/AT Corn/NP Belt/NP -/: in/IN
the/AT past/NN six/CD weeks/NNS has/HVZ received/VBN less/JJ than/IN
half/ABN of/IN the/AT rainfall/NN normal/JJ for/IN this/DT time/NN of/IN
year/NN ./.
.End of Sentence/.
.Start of Sentence/.
Most/QL vulnerable/JJ are/BER corn/NN and/CC soybeans/NNS ,/, the/AT
nation/NN 's/$ two/CD most/RB valuable/JJ crops/NNS with/IN a/AT combined/JJ
projected/VBN value/NN this/DT year/NN of/IN more/JJ than/CS $23/CD
billion/CD ./.
.End of Sentence/.
.End of Discourse/.
.End of File 1/.
.End of File 2/.
.End of File 3/.
```

Figure A.2: Sample text with automatically assigned part-of-speech tags. The selected tag follows each word, with a slash (/) used as a separator. To improve readability, the end-of-line character that separates tagged words has been replaced by a space.

A.4 Extracting pairs of related words

A.4.1 Syntagmatic relationships and surface forms

In Section 2.2 I discussed which of the realizations of the syntagmatic relationships of Section 2.1 should be included in the subsystem that identifies semantic groups. I concluded that for adjectives, I need to collect modified nouns, modified and modifying adjectives, other adjectives that modify a noun in common with the adjective of interest, and possibly modifying adverbs. For a noun, the relationships imply extracting modifying and modified nouns, modifying adjectives, verbs that take the noun as a subject or object, and other nouns that modify a common noun with the original one. In addition, for the purposes of Chapters 6 and 7, I also need to extract scaling and intensifying adverbs modifying adjectives and conjunctions of adjectives[18] respectively (see Sections 6.2 and 7.3 for a description of how these pairs are used for the subsequent processing of the retrieved semantic groups). The entire list of pairs of words that are used in the system and need to be extracted from the text is given in Table A.3.

Once I have specified the types of pairs that I want to extract from the tagged corpus, I need to identify the surface syntactic patterns that correspond to each such type of pairs (i.e., to each syntagmatic relationship). Finding such mappings between syntagmatic relationships and their surface forms is not in general easy, although we can use a detailed

[18]Including negated forms such as *big but not clumsy*.

Type of pairs	Use
Adjective and noun in a modification relationship	Positive indicator for the task of forming semantic groups of adjectives and of nouns
Adverb and adjective in a modification relationship	Possible positive indicator for the task of forming semantic groups of adjectives
Verb and head of subject noun phrase or head of direct object noun phrase	Positive indicator for the task of forming semantic groups of nouns[19]
Two adjectives in a modification relationship or simultaneously modifying a noun	Negative indicator for the task of forming semantic groups of adjectives
Two nouns in a modification relationship or simultaneously modifying a third noun	Negative indicator for the task of forming semantic groups of nouns
Conjunction between adjectives and/or negated forms of adjectives; the conjunction can be *and*, *or*, *but*, *either ... or*, or *neither ... nor*	Indicator of semantic orientation
Intensifying adverb and adjective in a modification relationship	Indicator of the adjective being a scalar one

Table A.3: Pairs of words that are used by the system and need to be extracted from the text.

grammar of English such as [Quirk *et al.* 1985], [Baker 1995], [Greenbaum 1996], or [Bache and Davidsen-Nielsen 1997] that lists the possible ways such relationships can be expressed. I present below the major surface forms that correspond to the syntagmatic relationships of Table A.3.

Adjective-noun modification This modification relationship shows the greatest variety of surface forms among the ones I am interested in. It can be realized in English with the following surface forms:

[19]While the extraction grammar contains rules for identifying verb and noun phrases and their heads, information about verb-noun relationships is not currently collected by the parser and is not used in the semantic classification of nouns.

- The adjective precedes the noun, possibly with other adjectives and/or nouns in between, as well as other intervening grammatical categories such as conjunctions, adverbs, intensifiers, commas, even entire noun phrases and relative clauses. For example, *vivid, copiously illustrated catalog.* This is the most common realization of this relationship.

- The adjective appears in a predicative position after a copular verb such as *be*, *seem*, or *grow*, and the noun is the head of the noun phrase that functions as the subject of the copular verb. For example, *Some questions remain unanswered.* In this case, the adjective does not belong in the same noun phrase with the noun it modifies; again various kinds of intervening modifiers may separate the verb and the adjective. Note that not all adjectives can appear in such a form, and sometimes the sense of the adjective is affected by its use in an attributive or predicative position [Elhadad 1991]. Compare *the late professor*, usually meaning *deceased*, and *the professor is late*, always meaning *tardy*. On the other hand, some adjectives can only appear in predicative position (e.g., the *a*- adjectives: *afraid*, *alone*, *asleep*, etc.).

- The adjective may be the complement of a verb that subcategorizes both for a direct object and an adjectival complement. In this case the modified noun appears as the head of the noun phrase that functions as the direct object of the verb. This category of verbs, which I will call *object-complement* verbs, includes for example *consider*, *make*, and *declare*; e.g., *The court declared the administration's policy unconstitutional.* Since many members of the class of object-complement verbs are resultative (e.g., *make* and *drive*), aspectual constraints apply to the adjectives that appear as complements of these verbs.

- The adjective may postmodify the noun. This may be either because the adjective-noun pair forms a frozen collocation, as in *Inspector General*, *president elect*, or *heir apparent*, or because the adjective subcategorizes for a prepositional phrase, e.g. *water suitable for drinking*; compare **suitable for drinking water.* The first type of this surface form is available to very few adjectives, and the set of nouns that can be modified in this way is also restricted. Even the second type is relatively rare, since movement of the prepositional phrase after the noun (e.g., *suitable water for drinking*) is common.

- The adjective may appear in an appositive phrase, following the noun it modifies, e.g., *provide an excuse, moral or economic, for government intervention.* The adjectival phrase is delimited by commas or dashes, and frequently contains a conjunction of two or more adjectives. Unlike NP-NP apposition, explicit indicators of apposition such as *namely* or *especially* are usually absent.

Adverb-adjective modification Adverbs precede the adjective they modify, e.g., *highly speculative*, with the exception of *indeed* which may appear either before or after the item it modifies (e.g., *indeed happy* – *happy indeed*), and *enough* which always postmodifies the adjective (e.g., *safe enough*). The adverb and the adjective are part of the same adjectival phrase and are typically close in the text. The same surface forms also apply to the more specialized case of an intensifying adverb modifying an adjective. In this case, words that are frequently called *qualifiers*, such as *more*, *so*, *too*, and *very*, may appear in the intensifying adverb's position.

Verb-noun relationships These relationships are observed between heads of phrases (a noun phrase and a verb phrase) that may be quite far apart in the clause structure. Interpretation is straightforward once the phrase boundaries have been identified and the role of each phrase has been determined. The verb phrase is typically finite, although it can also be nonfinite (e.g., a complement of another verb). If the voice of the verb phrase is active, the subject noun phrase almost always precedes the verb phrase and the direct object noun phrase immediately follows the verb phrase, with the exception of bitransitive verbs where the indirect object may appear between the verb and the direct object. A complement, which can be an adjectival or noun phrase, follows the verb for copular verbs and the direct object for object-complement verbs.

If the verb phrase is in passive voice, the (semantic and deep-syntax) subject is usually omitted although it may appear as the complement of *by* in a prepositional phrase after the verb phrase; with monotransitive or object-complement verbs, the deep-syntax direct object becomes the surface subject. Copular verbs do not allow the passive voice. With object-complement verbs, the semantic object becomes the surface subject and the behavior of the verb is similar to the one exhibited by an active copular verb; for example, *A weakening dollar drove interest rates higher* → *Interest rates were driven higher (by a weakening dollar).* Bitransitive verbs in the passive voice may raise either the direct or indirect object to the subject position, although the indirect object is more common; e.g. *John gave Sue some flowers* → *Sue was given some flowers (by John)* or *Some flowers were given Sue (by John).*[20]

[20] Usually, this form of the passive is avoided in favor of the passive of the alternate form of bitransitive verbs where the indirect object is expressed with a prepositional phrase. In this case, the example would

Adjective-adjective and noun-noun incompatibility relationships In both of these cases the two semantically incompatible words occur in the same adjectival or noun phrase. While the incompatibility stems either from one of them modifying the other or from both of them modifying a third noun (or, very rarely, adjective), we cannot easily tell which of the two cases holds. Fortunately, we do not need to do so for our purposes (see Section 2.2.3). In Section 2.2.3 I observed that a sequence of two or more nouns or adjectives occurring next to each other is the most common surface manifestation of this incompatibility relationship. If more than two words appear in the sequence, all of them are incompatible with each other. The sequence must run uninterrupted, since conjunctions and commas between the words may negate the incompatibility by introducing coordination or apposition.

Conjunction between adjectives The conjoined adjectives appear in the same adjectival phrase, and they may be premodified by adverbs or qualifiers. Conjunction patterns can be combined, giving sometimes highly ambiguous constructs. For example, *ominous and threatening or favorable* accepts two structural analyses: [[*ominous and threatening*] *or favorable*] and [[*ominous and threatening*] *or* [*ominous and favorable*]], the second being rejected on semantic grounds. Scope ambiguities generally exist, however, even with a single conjuction, in terms of modification scope. They are further exacerbated when the negative particle *not* or some other expression of negation is present.

Although we can find rules like the above that map the syntagmatic relationships to

---

become *Some flowers were given <u>to</u> Sue (by John).*

surface syntactic forms, two problems persist. First, the complexity of the resulting forms may be high, with the two words that are linked together being quite far apart in the sentence. This problem is most important for relationships that involve words from different major clause constituents. In that case I need to be able to recognize the various valid forms that an adjective, noun, and verb phrase can take, and to identify the head of a phrase. Second, even when a full syntactic analysis is available, *scope ambiguities* remain and cannot be resolved without access to lexical semantic knowledge. Conjunctions offer perhaps the prototypical example of such ambiguities, as discussed above. However, even for a sequence of two adjectives and a noun, we cannot tell if the first adjective modifies the noun as in *big yellow house* or the second adjective as in *light blue paint* without access to the semantics of the words involved.

I address the first problem of syntactic complexity by observing that while the complexity of the surface manifestations of the relationships I am interested in can be very high, most occurrences of the relationships nevertheless tend to use relatively simple forms. For example, noun phrases can become very complex, containing adjectival, noun, and adverb premodification, premodification by genitives, apposition of a noun or adjectival phrase, coordination, and postmodification by an adjective, an adverbial adjunct, a prepositional phrase, a relative clause, or a nonfinite *-ing* or *-ed* clause. Yet, Quirk *et al.* performed an analysis of a sample of 16,961 noun phrases from the Survey of English Usage corpus, and found that 67.8% of them were proper nouns, pronouns, or simple nouns without any modifiers [Quirk *et al.* 1985, pp. 1350–1352]. In addition, 19% of the total number of noun phrases in the sample contained either a single adjectival premodifier or a single simple prepositional phrase postmodifier. Even for scientific writing, which as a text category exhibits the highest complexity, they found that only 16.3% of the noun phrases contained multiple or complex modifiers.

Since the present system uses the corpus as a source of examples of language use, it is not imperative that all occurrences of valid examples be retrieved.[21] It is also beneficial to the accuracy of the system if we extract examples with high confidence, instead of collecting examples that are likely to be mistakenly analyzed. This goal of simplicity needs to be balanced with the goal of collecting as much information as possible. Therefore, I have identified a reduced set of syntactic constructs that are frequent in the corpus and are not so hard to retrieve. I proceed to describe a grammar specifying this set in the next subsection. I then return to the problem of structural and scope ambiguities in Section A.4.3, where I describe how these ambiguities can be resolved with the use of appropriate heuristics, in a way that does not guarantee absolute correctness but produces the correct interpretation most of the time.

A.4.2 A grammar for the surface forms

I have separated the description of the syntactic structures that the proposed system will recognize in two separate levels. The first level deals with relationships between major clause constituents, namely adjectival, noun, and verb phrases; prepositional phrases are in

[21] Although retrieving more examples improves the overall efficiency of the learning process.

general ignored except for the special cases mentioned below. The second level describes the allowed forms for each of the major phrases.

At the highest level, the grammar specifies *sentence fragments*. A sentence fragment is defined as one of the following:

1. an adjectival phrase (AP)
2. a noun phrase (NP)
3. an optional noun phrase plus a verb phrase (VP) with the features +COPULAR, +ACTIVE plus an adjectival or noun phrase
4. an optional noun phrase plus a verb phrase with the features +OBJECT-COMPLEMENT, +ACTIVE plus a noun phrase, plus another adjectival or noun phrase.
5. an optional noun phrase plus a verb phrase with the features +OBJECT-COMPLEMENT, −ACTIVE plus an adjectival or noun phrase, optionally followed by a by-prepositional-phrase (BPP)
6. an optional noun phrase plus a verb phrase with the feature +ACTIVE, optionally followed by a noun phrase
7. an optional noun phrase plus a verb phrase with the feature −ACTIVE, optionally followed by a by-prepositional-phrase
8. an optional noun phrase plus a verb phrase with the features +BITRANSITIVE, +ACTIVE followed by two noun phrases

9. an optional noun phrase plus a verb phrase with the features +BITRANSITIVE, −ACTIVE, plus a noun phrase, optionally followed by a by-prepositional-phrase

10. an optional noun phrase plus a verb phrase with the features +BITRANSITIVE, −ACTIVE, plus a to-prepositional-phrase (TPP), optionally followed by a by-prepositional-phrase

At this level, the grammar captures the most common manifestations of the relationships between the adjectives, nouns, and verbs that appear as heads of the corresponding phrases. The structure of the five types of phrases (AP, NP, VP, BPP, and TPP) is specified separately below. I selected the defining rules for these phrases so that almost all local constructs are covered, but any constructs that could lead to long-distance dependencies, such as relative clauses and (embedded) prepositional phrases, are disallowed.

A by-prepositional-phrase or to-prepositional-phrase is simply the corresponding preposition (*by* or *to*) followed by a noun phrase. A noun phrase is defined as one of the following:

- A sequence of one or more proper nouns.
- A personal or reflexive pronoun. If it is a personal pronoun, it can be in either the subjective or the objective case.
- A compound indefinite pronoun (e.g., *anyone*, *nothing*).
- An optional predeterminer or quantifier (e.g., *both*), followed by an optional determiner and/or an optional adjectival phrase (AP), followed by a noun sequence. A

determiner can be a possessive pronoun (e.g., *her*), a sequence of one or more cardinal numbers (e.g., *two*, *35*, *three hundred thousand*), a deictic pronoun/determiner (e.g., *this*, *those*), an indefinite pronoun/determiner (e.g., *some*), or an article (e.g., *the*). An adjectival phrase consists of a sequence of qualifiers (e.g., *so*, *too*), adverbs, and adjectives, with at least one adjective. A noun sequence is a sequence of nouns and possessive markers (*'* or *'s*) that contains at least one noun, does not contain two possessive markers in sequence, and does not start with a possessive marker.

- A predeterminer or quantifier, or a sequence of one or more cardinal numbers, or a deictic pronoun, or an indefinite pronoun. This is a special case of the previous alternative, where there is no noun sequence and no adjectival phrase. Such a noun phrase structure can occur because of ellipsis of the head noun (e.g., *Both went to the funeral*, *That coat will be five hundred*) or because some of the indefinite or demonstrative determiners can also function as pronouns (e.g, *This is not acceptable*).

The noun phrase may also include at the end an appositive adjectival phrase, which must be delimited with matching commas or dashes. In addition, coordinators (whether central coordinators such as *and* or indefinite dual pronouns/determiners such as *neither* that function as coordinators) and commas can also appear anywhere inside a noun or adjectival phrase, with the constraint that they do not appear at the start or the end of the phrase or after another word or symbol of the same category.

A verb phrase is defined as an optional auxiliary (such as *may*, *should*, or *would*) or an optional finite form of *do*[22], plus one of the following:

- A single verb (finite or non-finite), e.g., *find*, *finding*.
- A form of *have* plus a past participle, e.g., *had found*, *having found*.
- A form of *be* plus a present participle or a past participle, e.g., *are finding*, *was found*, *been found*.
- A form of *have* plus *been*, an optional *being*, and a past participle, e.g., *had been found*, *having been being found*.[23]

The optional auxiliary or *do*-form cannot be present if a non-infinitive verb form follows it. Commas and adverbs (including qualifiers) can also appear at any place within a verb phrase, with the constraint that commas do not appear at the start or the end of the phrase or after another comma. The inclusion of auxiliary *do*-forms allows handling of negation[24] (e.g., *did not mean*) and emphatic constructs (e.g., *did mean*).

Since the grammar does not recognize noun phrases that contain embedded prepositional phrases or clauses as single phrases, special care must be taken to avoid locating only part of a noun phrase that enters in relationships with other major phrases and identifying the wrong head from it for inclusion in relationships with the heads of the other phrases. For

[22] *Do* has only finite forms as an auxiliary; cf. [Quirk *et al.* 1985, pp. 132–133].

[23] The last case (nonfinite passive perfective progressive) is quite complicated, and rare; no occurrence of this verb phrase structure appears in the 1987 Wall Street Journal corpus.

[24] Since *not* is marked by the tagger as an adverb.

example, the sentence *Industrial production in Italy declined in January* would be analyzed by the grammar as it has been described so far in the following manner:

[Industrial/JJ production/NN]$_{\mathrm{NP}}$ in/IN [Italy/NP]$_{\mathrm{NP}}$ [declined/VBD]$_{\mathrm{VP}}$ in/IN [January/NP]$_{\mathrm{NP}}$

where the major phrases recognized by the grammar are indicated with brackets. Rule 6 for sentence fragments covers the fragment *Italy declined* and, if applied, would produce an incorrect subject-verb association between *Italy* and *declined* instead of the correct one between *production* and *declined.* A similar problem exists with relative clauses. The sentence *Appropriations assigned the tax agencies were judged insufficient* would lead to an incorrect subject-complement association between *agencies* and *insufficient* instead of the correct one between *appropriations* and *insufficient.*[25]

If the grammar can cover only part of a noun phrase, then this part will almost always be a right substring of the full noun phrase for noun phrases in subject position and a left substring of the full noun phrase for noun phrases in object or complement position. This is due to the fact that many constructs that may postmodify noun phrases are not covered by the rules for noun phrases, while almost all premodifying construncts are covered. The problem of identifying the wrong head for a noun phrase is associated with the grammar's failure to recognize that a noun phrase is part of a larger noun phrase or of the preceding clause. Since recognizing a right substring of the full noun phrase is much more problematic

[25]Such constructs are much rarer that the prepositional phrase postmodification described in the previous example. Some speakers even find the present example sentence ungrammatical, since they are uncomfortable with the second passive form of bitransitive verbs as it appears in sentences like *The book was given Mary.*

than recognizing a left substring for the purpose of correctly identifying the head of the phrase, the situation presents significant difficulties for the grammar practically always only when the noun phrase is in subject position (rules 3–10 for sentence fragments). To correct this problem, I include an additional constraint for these rules, namely that they can be applied only when the left context indicates that the recognized noun phrase is not a proper right substring of a larger noun phrase or of the preceding clause. More specifically, these rules for sentence fragments apply only at the start of a sentence or when the immediately previous word is a subordinating or coordinating conjunction, a *wh*-word, a noun, a pronoun, a cardinal number, a determiner (including indefinite and demonstrative determiners/pronouns), or a punctuation symbol. In other words, the rules are suspended after a preposition or verb phrase which would indicate that the noun phrase is part of a prepositional phrase or of a (possibly reduced) relative clause or separate clause respectively.

The fact that sometimes the subject noun phrase may not be recognized by the grammar (i.e., that a partial noun phrase is identified and then rejected according to the constraints given above) is the reason that I have designated subject noun phrases as optional in the rules for segment fragments above (rules 3–10). In this way, even if the grammar does not recognize the subject noun phrase, it can retrieve other relationships that do not involve the (surface) subject, such as object-complement (rule 4) or verb-object relationships (rules 6, 8, and 9). Even if no such relationships exist (as is the case for rules 3, 5, 7, and 10), making the subject optional enables partial matches and allows subsequent matches to be correct instead of being potentially misled by residual phrases that belong to a previous sentence fragment.

The grammar rules are listed in concise form in Figure A.3, using the nonterminals described above and the part of speech categories of Table A.2 as terminals. In some cases, specific lexical items are referenced in the grammar when there is no tag exactly corresponding to them (e.g., *–/:*, *having/VBG*, *did/VBD*). The grammar also uses features for the VP category that are either lexical features inherited from the head verb in much the same way as in Lexical Functional Grammar [Kaplan and Bresnan 1982] (COPULAR, OBJECT-COMPLEMENT, BITRANSITIVE), or are computed during the construction of the verb phrase (ACTIVE, FINITE, PERFECTIVE, PROGRESSIVE, TENSE).

The grammar contains several additional features that are computed during the recognition of a sentence fragment. These features, which are not used to decide the grammaticality of a fragment but are output for further processing by subsequent stages, are:

- Aspect, for adjectives. It can have one of the values *dynamic*, *static*, or *unknown*. An adjective is marked as *dynamic* (*static*) if it appears as the complement of a copular or object-complement verb phrase that has (doesn't have) the PROGRESSIVE feature. If an occurrence of an adjective is not in the complement of such a verb, the value *unknown* is assigned to the ASPECT feature. Note that these values are not the result of global decisions for the adjective, but rather local indicators of the overall aspectual character of the adjective.
- Degree, for adjectives and adverbs. It can have one of the values *absolute*, *comparative*, or *superlative*.
- Gender, for the head of a noun phrase and by extension for the adjectives that modify

SF → *AP* |
NP |
[* *NP*] *VP*(+COPULAR, +ACTIVE) (*AP* | *NP*) |
[* *NP*] *VP*(+OBJECT-COMPLEMENT, +ACTIVE) *NP* (*AP* | *NP*) |
[* *NP*] *VP*(+OBJECT-COMPLEMENT, −ACTIVE) (*AP* | *NP*) [*BPP*] |
[* *NP*] *VP*(+ACTIVE) [*NP*] |
[* *NP*] *VP*(−ACTIVE) [*BPP*] |
[* *NP*] *VP*(+BITRANSITIVE, +ACTIVE) *NP* *NP* |
[* *NP*] *VP*(+BITRANSITIVE, −ACTIVE) *NP* [*BPP*] |
[* *NP*] *VP*(+BITRANSITIVE, −ACTIVE) *TPP* [*BPP*]

The * symbol indicates required left context that must be present but is not matched as part of the text covered by the rule. This context can be either the start of the sentence or one of the following terminal symbols: ABN AT CD DT DTS DTI DTX NN NNS NP PN PPS PPSS PPO PP$ PPL PPLS WDT WPS WP$ WRB CC CS ( ) , : .

AP → A sequence of QL, RB, CC, comma (,), and JJ, with the constraints that it does not start or end with CC or comma, does not contain two successive CCs or commas, and contains at least one JJ.

NP → *NP* (, *AP* ,) |
NP (−/: *AP* −/:) |
PPS | PPSS | PPO | PPL | PPLS |
PN |
[ABN | DTX] [*DET*] [*AP*] *NS* |
ABN | DTX | (CD+) | DT | DTS | DTI

In addition, CC and comma (,) may appear repeatedly anywhere in the *NP* except the start or the end, with the additional constraint that no two successive coordinators or commas may appear.

DET → PP$ | (CD+) | DT | DTS | DTI | AT

NS → A sequence of NN, NNS, and $, with the constraints that it does not start with $, does not contain two successive $'s, and contains at least one NN or NNS.

Figure A.3: Rules of the grammar used for the extraction of syntagmatic relationship realizations. Nonterminals are in italics, while terminals in roman typeface. Terminals refer to the part of speech tags of Table A.2, or to specific combinations of a lexical item and tag; in the latter case, a slash (/) separates the word and the tag. Parentheses are used for grouping, or for attaching features to the preceding constituent. Brackets indicate optional constituents. The | symbol separates alternatives, while the + symbol indicates one or more repetitions of the item at its left. Continued on the next page.

$BPP \rightarrow$ by/IN NP

$TPP \rightarrow$ to/IN NP

$VP \rightarrow$ VB | VBZ | VBD | VBG | VBN | BE | BEM | BER |
BEZ | BED | BEDZ | BEN | HV | HVZ | HVD | DO |
(HV | HVZ | HVD | having/VBG) (VBN | BEN | HVD) |
(BE | BEM | BER | BEZ | BED | BEDZ | BEN) (VBG | VBN | BEN | HVD) |
(HV | HVZ | HVD | having/VBG) BEN [being/VBG] (VBN | BEN | HVD) |
(MD | DF) (VB | HV | DO) |
(MD | DF) HV (VBN | BEN | HVD) |
(MD | DF) BE (VBG | VBN | BEN | HVD) |
(MD | DF) HV BEN [being/VBG] (VBN | BEN | HVD)

In addition, QL, RB, and comma (,) may appear repeatedly anywhere in the VP except the start or the end, with the additional constraint that no two successive commas may appear.

$DF \rightarrow$ DO | does/VBZ | did/VBD

Figure A.3: Rules of the grammar used for the extraction of syntagmatic relationship realizations. Continued from the previous page.

this head word. It can have one of the values *masculine*, *feminine*, *non-person*, or *unknown*. The current implementation assigns to all words the value *unknown* except for personal pronouns (e.g., *she*, *him*, *it*).

- Modification, for adjectives. It indicates the type of modification relationship between the adjective and the noun phrase it modifies, according to the rules of Section A.4.1. It can take the values *attributive*, *predicative*, *complement* (for the complement of an object-complement verb), *appositive*, or *unknown*. The last value is assigned to an adjective when it cannot be reliably attached to some specific noun phrase. Note that postpositive modification of nouns by adjectives is not implemented in the grammar, partly because it is not so frequent but mainly because its implementation would considerable complicate the grammar, as the boundary between successive noun phrases would become harder to detect.

- Negated, for adjectives. It is a binary feature that indicates whether the adjective is in the scope of a negating construct such as the particle *not* or the adverb *hardly*.

- Number, for a verb phrase, a noun phrase, and by extension for the adjectives that modify the head of this noun phrase. It can take the values *singular*, *plural*, *ambiguous*, or *conflicting*. Initially, values for this feature are assigned individually to each verb phrase, noun phrase head, and noun phrase determiner, according to the assigned tags for the first two cases and to either the tags or precompiled lexical information for the last case. Some determiners (e.g., *some*, *the*) or tags (e.g., *VBD* for past tense verbs or *NP* for proper nouns) can be either singular or plural; in these cases, the value *ambiguous* is assigned to the NUMBER feature. Then a bottom-up unification process is employed, checking for number agreement between the verb phrase and its subject noun phrase, as well as between the head and determiner of all noun phrases, and unifying the corresponding NUMBER features. If at the end of this process both values are plausible for a phrase, the value *ambiguous* is reported, while a unification failure will produce the value *conflicting*.[26]

Table A.4 lists all the features used in the current implementation of the grammar.

There are three important points that should be noted about the grammar I have de-

[26] Note that failure of the unification does not mean rejection of the fragment, as it would for a more traditional parsing approach. Rather, the philosophy of this grammar is to recognize its inherent limitations and simply mark the offending case as potentially problematic. Later stages of the system can then independently decide if they prefer to trust the grammar's recognition mechanisms or the text writer's language competence.

Feature	Values	Applicable grammar categories	Method of determination
COPULAR, OBJECT-COMPLEMENT, BITRANSITIVE	+, −	Verb	Lexical
		Verb phrase	Inherited from head verb
ACTIVE, PERFECTIVE, FINITE, PROGRESSIVE	+, −	Verb, Verb phrase	Synthesized from tags and VP structure
TENSE	Present, Past, Unknown	Verb, Verb phrase	Synthesized from tags and VP structure
ASPECT	Dynamic, Static, Unknown	Adjective	Synthesized from the PROGRESSIVE feature of the corresponding linking verb
DEGREE	Absolute, Comparative, Superlative	Adjective, Adverb	Synthesized using morphology
GENDER	Masculine, Feminine, Non-person, Unknown	Noun phrase head	Lexical
		Noun phrase	Inherited from head of NP
		Adjective	Inherited from head of NP
MODIFICATION	Attributive, Predicative, Complement, Appositive, Unknown	Adjective	Synthesized according to clause and NP structure
NEGATED	+, −	Adjective	Synthesized from AP structure
NUMBER	Singular Plural Ambiguous	Determiner	Lexical or synthesized from tags
		Noun, Verb, Verb phrase	Synthesized from tags
	Singular, Plural, Ambiguous, Conflicting	Noun phrase	Synthesized by unifying the NUMBER feature of head, determiner (if present), and corresponding verb phrase (if the NP is in subject position)
		Adjective	Inherited from modified head of NP

Table A.4: Features used in the extraction grammar.

scribed in this section. First, given a string of words, several different sequences of alternate rules for the sentence fragment category may match it, producing different parse trees. To resolve ambiguity, I specify that in such cases the rule that covers the most words should be tried first, so that the sentence *Mary is smart* is covered by one rule (rule 3) and not by the two rules that cover the fragments *Mary is* and *smart* (rules 6 and 1 respectively). Since subsequent processing, and in particular the extraction of pairs of related words, is local to each sentence fragment, assigning a maximal sequence of words to a sentence fragment ensures that as many relationship instances as possible will be retrieved.

The reason that the grammar contains rules that cover a proper substring of what other rules cover is robustness and extended coverage. Given that the grammar does not cover all possible ways that the intended relationships between nouns, adjectives, verbs, and adverbs are expressed in English[27], a parser employing this grammar may fail to recognize an appropriate sentence fragment. But even when the entire clause cannot be covered by one rule, it is likely that at least parts of it will be correctly interpreted. For example, the sentence *BankEast's generous offer appears to be attractive to the typical shareholder* contains a subject-complement relationship instance (*offer–attractive*) which is not covered by the grammar as the latter does not recognize the deletion of the subject in the subordinate infinitive clause. Yet, the modular approach adopted in the grammar enables it to identify local relationships at lower syntactic levels, such as the modification relationships between *generous* and *offer* and between *typical* and *shareholder*.

The second point that should be made explicit about the grammar is that the lan-

[27]Much more detailed grammars fail to do so.

guage it is generating contains fragments that are not valid English phrases (or parts of such phrases). For example, the utterance *Him did not be well good so* is transformed to *Him/PPO did/VBD not/RB be/BE well/QL good/JJ so/RB* by the part of speech tagger and thus would be accepted by the grammar according to rule 3 for sentence fragments. However, the grammar was not designed as a model of linguistic competence, and is not intended to be used to validate the input text. Rather, it is expected that the input will be syntactically correct and will satisfy semantic and lexical constraints, as it is part of a corpus of actual written English prose.[28] Therefore, the over-acceptance of the grammar is not a drawback; indeed, in several cases this strategy allows the simplification of the grammar itself, instead of incorporating various cases testing for constructs that should not (and will not) appear. This approach also offers the advantage of enabling the extraction of information even from grammatically questionable or unacceptable utterances, as long as parts of these utterances can be matched to one or more of the grammar's rules.

Finally, the use of lexical information in the form of inherited lexical features such as BITRANSITIVE and COPULAR should be discussed. It was noted in Section 1.5 that the system should not depend on precompiled lexical resources. Designing a grammar that would intrinsically depend on such features would invalidate a major part of the appeal of the system. On the other hand, some kind of syntactic lexical knowledge is useful for guiding the grammar rules. The solution I adopted was to use very limited lexical knowledge at the syntactic level only, employing only properties that are true for small and relatively

---

[28] This is even more the case for carefully written and edited text, such as newspaper articles. Actual spoken text may contain grammatical violations, such as reformulations and repairs [Levelt 1983; Bear *et al.* 1992; Nakatani and Hirshberg 1993; Heeman and Allen 1994; Heeman and Allen 1997]. But in most cases, text produced by humans has a lower error rate than the automatic methods that check its grammaticality, unless the scope of these methods is very limited.

appear	come	go	leave†	remain	show†	stay
be	emerge	grow	lie	report†	smell	taste
become	feel	hold†	look	rest†	sound	turn
call†	get	keep	prove	seem	stand†	

Figure A.4: Copular verbs known to the extraction system. Verbs marked with † are not copular verbs in the full sense of the term, i.e., the adjectival or noun phrase that follows does not have full complement status as it can be omitted without a change in the meaning of the verb. However, this adjectival or noun phrase is frequently present (e.g., *leave furious*, *stand quiet*), and the relationship between the subject and this phrase is a copular one.

stable sets of words. These properties correspond to partial syntactic subcategorization information for verbs, as manifested with the features COPULAR, OBJECT-COMPLEMENT, and BITRANSITIVE, and number information for indefinite determiners/pronouns and articles.

Since the sets of copular, object-complement, or bitransitive verbs are small and relatively stable over time and domain, we can use for each set a simple list of all its members. Similarly, the small number and closed-class status of indefinite determiners/pronouns and articles allows listing them along with the value of the associated NUMBER feature. In this way, access to a lexicon continues to be unnecessary. Note that, consistently with this philosophy, the grammar does not test a feature TRANSITIVE to see if a verb should take an object, although such a test would be useful for checking if a noun phrase should follow the verb phrase in rule 6 for sentence fragments (page 516), and if rule 7 for sentence fragments can apply at all. The reason is that we cannot prepare a small list of all the transitive (or intransitive) verbs in advance, so if this test were adopted we would have to access a dictionary for this information.

Figures A.4, A.5, and A.6 contain all the lexical knowledge that has been encoded in

believe	deem	force	keep	picture	rate	suppose
call	demonstrate	get	leave	prefer	reckon	think
catch†	depict	grow	like	presume	render	turn
certify	display	have‡	love	proclaim	report	want
color†	drive	hold	make	profess	reveal	wish
consider	feel	imagine	order	pronounce	send	
declare	find	judge	paint†	prove	show	

Figure A.5: Object-complement verbs known to the extraction system. Verbs marked with † do not fully belong in the object-complement category as the adjectival or noun phrase that follows the direct object is not obligatory and therefore its grammatical status is between a complement and an adjunct. However, such a phrase frequently occurs with these verbs (e.g., *catch them unprepared*, *paint the house white*), and whenever it occurs, it is in a copular relationship with the object. *Have* is another special case: it fails some of the tests for an object-complement verb (for example, it doesn't allow the passive voice) but can appear followed by a noun and an adjectival phrase in an existential sense (e.g., *have two students ill*).

allow	do	give	offer	read	show
ask	envy	grant	order	refuse	spare
assign	excuse	hand	owe	reserve	teach
bring	find	leave	pay	save	tell
charge	fine	lend	promise	send	throw
deny	forgive	make	provide	serve	wish

Figure A.6: Bitransitive verbs known to the extraction system.

the system for verbs.[29] I have omitted from the verb lists (in both the figures and the actual implementation of the grammar) a small number of phrasal verbs (e.g., the copular *turn out*), since the grammar rules for verb phrases do not recognize prepositions as part of the verb phrase structure (cf. Figure A.3). I have also omitted lexical information about some verbs which exhibit copular or object-complement subcategorization patterns only in rare and highly specialized contexts. Examples of such omitted cases are *blush scarlet*, which

[29] I extracted these lists from an analysis of the subcategorization patterns of the most frequent verbs in the 1987 Wall Street Journal corpus, and I augmented them with the lists of verbs in [Quirk *et al.* 1985, pp. 1171–1174, 1196–1198, and 1208–1211].

Grammatical category	Value	Words
Determiners	Singular	*a, an, every, nary*
		Any other word tagged DT
	Plural	*all, both, many*
		Any other word tagged DTS
	Ambiguous	*no, the*
		Any other word tagged ABN, AT, DTI, or DTX
Possessive pronouns	Ambiguous	All words tagged PP$
Cardinal numbers	Singular	*one, 1*
	Plural	Any other word tagged CD

Table A.5: Values of the NUMBER feature for determiners and words that function as determiners. In addition to lexical information pertaining to specific words, the table also lists the rules that the system uses for the remaining determiners according to their assigned part of speech tag.

accepts as an adjectival complement only very few adjectives of color (compare with **blush tall* and **blush green*), and *sweep the floor clean*, which obeys similar semantic constraints. Since the use of these verbs with these syntactic patterns is severely constrained, removing them not only reduces the amount of precompiled lexical knowledge that is stored in the system, but also avoids introducing bias in the syntagmatic relationships that are extracted; the adjectival complement of *blush* is more constrained semantically by *blush* than by the subject of *blush*, and therefore *blush* does not function as a semantically neutral linking verb.

Table A.5 shows the values of the NUMBER feature for determiners and other words that have a similar syntactic function and can specify the number of a noun phrase (e.g., possessive pronouns and cardinal numbers). Explicit lexical information is used only for a small subset of the articles, indefinite determiners/pronouns, and other words in this category. The rules used by the grammar to determine the NUMBER feature of the remaining determiners and similarly functioning words according to their part of speech tag are also listed in the table.

A.4.3 Heuristic analysis of the syntactic patterns

The extraction task does not end with the identification of the patterns covered by the grammar; once these patterns have been retrieved from the text, I need to further analyze them to produce the pairs of Table A.3 (Section A.4.1). The syntactic patterns described in the previous subsection capture most of the occurrences of the relationships I am looking for. Yet, they still contain cases of ambiguity of two types: Scope ambiguity for the constructs involving conjunctions and the negative particle *not*, and attachment ambiguity for constructs involving multiple candidates for a modifier to act upon. These types of ambiguity cannot be resolved without access to lexical semantic information about the words participating in the construct; there is nothing that distinguishes at the surface level the structure of *big yellow house* and *light blue paint*. On the other hand, we cannot resort to lexical semantics for solving this problem, since this would defy the very purpose of the system I describe in the dissertation. The solution I adopted is to use heuristic rules that decide the scope or attachment. These rules are based on what has been observed in the corpus among similar constructs that have been analyzed by hand. Like all heuristic methods, they are not guaranteed to produce the correct answer, but since they prefer the decision that has been observed to be correct most of the time, it is likely that the answer will also be correct most of the time when they are applied to new text.[30]

Consequently, the grammar of the previous subsection is augmented with the following heuristic disambiguation rules:

- Assume narrow scope for the conjunctions *and*, *or*, and *but* when multiple occurrences of these conjunctions appear in the same adjectival phrase. In other words, no deleted conjunctions that need to be re-introduced in the phrase are supposed to exist. Sequences of multiple conjoined terms are analyzed from left to right and converted to pairs of simple conjunctions consisting of the two words that appear immediately to the left and right of each conjunction. In this way, the phrase *good and efficient or bad* will produce the simple conjunctions *good and efficient* and *efficient or bad*. The method retrieves only valid pairs of conjoined words and has the added advantage of simplicity, since there may be several positions that would otherwise need to be considered if a deleted conjunction were to be re-inserted. Note this disambiguation rule does not apply if several adjectival phrases are connected with the same coordinator (or equivalently, in cases of asyndetic coordination where all but the last occurrence of a coordinator have been replaced by commas). In these cases, the full conjunction involving all participating adjectival phrases (i.e., all pairwise combinations of the adjectival phrases) is extracted.

- Assume wide scope for the negative particle *not*. This interprets phrases like *not*

[30] As is common with all statistical methods, I make the broad assumption that new text is produced by the same stochastic process as the old text I used for training. For problems like attachment whose solution does not vary with the text's genre and type, this assumption is more than justified.

consistent and credible as [[*not consistent*] *and* [*not credible*]] instead of limiting the scope of the negation to the first conjunct only.

- Attach an adjective inside a noun phrase to the last noun of the noun phrase (the head), but stop looking for the head if a possessive marker (*'* or *'s*) is encountered. This will correctly interpret phrases such as *tasty ice cream* as [*tasty* [*ice cream*]] and not as [[*tasty ice*] *cream*], but will fail in cases where the adjective must be attached to another adjective as in [[*light blue*] *paint*], or to a noun modifying the head noun, as in [[*financial services*] *company*]. However, occurrences of the first type are more frequent than the other two. The exception for possessive markers allows the correct handling of phrases like *the poor fisherman's wife*, where *poor* will be attached to *fisherman* and not to *wife*.

- Attach an adverb inside an adjectival phrase to the first adjective or adverb that follows it; if none exists, to the immediately previous adjective. According to this rule, the adjectival phrase *once prohibitively expensive* (from the sentence *Consumers are finding once prohibitively expensive traditional outfits affordable*) will lead to the correct attachments of *once* to *prohibitively*, and of *prohibitively* to *expensive*. The second case of the rule allows the proper handling of postmodifying adverbs such as *enough* and *indeed*.

- Attach a noun[31] inside a noun phrase to the noun immediately following it, as long as no comma separates the next noun from this one. If a comma does appear as a separator, then do not consider this noun as a modifier and do not attach it to

[31] Except the last noun, which is assumed to be the head.

any other noun. The above choice of left-branching within sequences of nouns is supported by empirical evidence[32] and will correctly interpret *state college student* as [[*state college*] *student*] (i.e., a student who attends a state college) and not as [*state* [*college student*]] (i.e., a college student who is somehow associated with the state). Nouns separated by commas from the rest of the noun phrase could be modifying the head of the noun phrase; but in most cases, they are a manifestation of (possibly asyndetic) coordination within the noun phrase (e.g., *the army, navy, and air force*) or part of an adverbial adjunct or disjunct at the clause level (e.g., *for example, government employees are* ...). In both cases, it would be wrong to consider *army* a modifier of *force*, or *example* a modifier of *government* or *employees*. Since noun-noun modification relationship instances are an indicator of semantic unrelatedness where accurate extraction is of paramount importance (see Section 2.2.3), it is preferable to avoid extracting instances of the relationship when the probability of their being valid indicators of incompatibility is low.

In addition to the above rules that disambiguate scope and attachments, disambiguation rules are also needed when there multiple ways to interpret the roles of the constituent phrases in a syntactic rule of the grammar. The grammar of the previous subsection has only one rule which can be the source of such *syntactic* ambiguity: rule 9 for sentence fragments, which specifies that a sentence fragment can be a noun phrase followed by a bitransitive verb in the passive voice which is in turn directly followed by another noun

[32]Lauer [1995, p. 50] analyzed 308 triples of consecutive nouns observed in Grolier's encyclopedia. Of these, 279 sequences were part of the same phrase, and 58% of the latter exhibited left-grouping while only 29% right-grouping (there was no preference for either left or right grouping in the remaining 13% of the 279 noun sequences, such as in *precision navigation systems*).

phrase. There are two possible passive constructs for a bitransitive verb, one where the apparent subject is the deep syntax indirect object and the apparent object is the deep syntax direct object (e.g., *Mary was given the book*), and another where the roles of the deep syntax direct and indirect objects above are interchanged (e.g., *The book was given Mary*). The structure of both passive forms is identical, which makes the interpretation of phrases matched by rule 9 ambiguous. However, the first passive form given above (and the associated interpretation) is far more common.[33] Consequently, I adopted the disambiguation rule that the surface subject is always the deep syntax indirect object while the surface object is the deep syntax direct object.

A similar ambiguity problem for bitransitive verbs in the active voice is implicitly avoided by appropriately constraining the grammar rules. This problem is related to the occasional omission of the indirect object, as in the sentence *The colonel gave orders to evacuate the village*. In this case, where only one noun phrase directly follows the bitransitive verb, rule 8 for sentence fragments does not apply, since it requires two noun phrases to follow. Consequently, the rule that is selected is the general transitive rule (rule 6) which correctly interprets the noun phrase as a direct object.

Finally, there are cases where a sentence fragment containing a verb phrase can be matched to more than one rule, with different interpretations for the verb phrase (i.e., copular, object-complement, bitransitive, monotransitive, or intransitive[34]). Normally, we expect the lexical information encoded in the features COPULAR, OBJECT-COMPLEMENT,

---

[33] See footnotes 20 and 25 on pages 512 and 520 respectively.

[34] Obviously, not all of these possibilities are applicable to any given sentence fragment.

and BITRANSITIVE to filter out all but one possibility. If more than one choices survive (for example, *get* is considered both a copular and an object-complement verb, and *find* is considered both an object-complement and a bitransitive verb (cf. Figures A.4, A.5, and A.6); both can also be used as regular monotransitive verbs), an explicit disambiguation decision needs to be made only when the verb phrase and at least one noun phrase following it have been matched. If only the verb phrase has been matched, no relationship instance will be extracted and consequently we do not need to decide the type of the verb. Similarly, adjectival phrases following the verb have always a unique interpretation (as complements). For the remaining case of noun phrases, which can be either an object or a complement, I use the heuristic rule that prefers the object over the complement interpretation. In this way, interpretation of the verb as bitransitive or monotransitive has the highest priority, followed by the object-complement and then the copular interpretations.

Note that not all potentially ambiguous constructs need to be disambiguated. Such "don't care" cases arise either because I am not interested in relationships involving the words in the pattern, or because the attachment decision will not make any difference for the extracted relationship instance. An example of the first case is attachment of adverbs inside a verb phrase; since I am not collecting information about verbs or adverbs, the exact attachment is immaterial.[35] Sequences of adjectives exemplify the second case: whether the correct attachment is $[[X\ Y]\ Z]$ or $[[X\ Z]$ and $[Y\ Z]]$, the same information, namely that the three adjectives X, Y, and Z are semantically unrelated, will be extracted (cf. Sections 1.7.5

[35]Naturally, if the system is extended to cover relationships with the primary focus on verbs or adverbs (e.g., for forming semantic groups of verbs) and this modification relationship is actually informative, a rule for this case will need to be added.

and 2.2.3).

A.4.4 Finite-state implementation of the grammar

I have implemented a parser that recognizes the constructs described by the grammar of Section A.4.2. The parser operates on two levels. First, it greedily processes the input (a sentence where the words have been separated and augmented with part of speech information as described in Sections A.3.3–A.3.6) to identify maximal sequences of tokens that can form one of the five types of phrases *NP*, *AP*, *VP*, *BPP*, or *TPP*. This is accomplished by examining the part of speech tag of each word (or punctuation symbol) against precompiled sets of tags that can either start or terminate a particular kind of phrase; in the latter case, the terminator is not part of the phrase. For example, the following grammatical categories are always terminating a phrase: phrase or sentence final punctuation, the existential *there*, *wh*-words, and subordinating conjunctions. Other categories may also terminate a particular type of phrase; for example, a determiner, adjective, or noun always terminate a verb phrase. The rules for termination of the phrase may also depend on features of the current word or previous words in the phrase that are either lexical or synthesized from the part of speech tags (see Table A.4); for example, if we are currently processing a verb phrase starting with *is*, a present or past participle (with tag *VBG*, *VBN*, or *BEN*) will be included in the phrase but an infinitive or finite form of a verb (i.e., a word that has been assigned any other verb tag) will terminate the verb phrase. In general, the rules of Section A.4.2 (Figure A.3) determine the allowed structure for each type of phrase.

Once the parser has identified the extent of a phrase, it automatically analyzes it, using the disambiguation rules of Section A.4.3 if necessary. This allows the immediate extraction of information that is local to a phrase, without waiting for determining the role of the phrase in the overall sentence structure. As discussed earlier, the latter may not even be possible.

The parser keeps track of the required arguments of previously analyzed phrases. For example, upon seeing a verb phrase with the +ACTIVE feature[36], the parser will expect to see one noun phrase that will fulfill the syntactic object role. If the expected phrase(s) do in fact occur after the current phrase, then the parser will match such sequences of phrases to the longest corresponding rule for sentence fragments, and extract relationships that cross phrase boundaries, such as a subject-verb relationship, or an adjective-noun relationship where the adjective appears in predicative position and modifies the head of the subject noun phrase. Since all such relationships involve only the heads of the participating phrases, information about the head is saved when the local analysis for each remembered phrase is completed.

The algorithm that the parser follows at this second level is a version of multiple pattern matching which matches a set of patterns to a (potentially long) search string. The algorithm updates the current position by one symbol a time and keeps track of all patterns that start at the current position or earlier and are still viable matches after the current position is considered. In my version, each recognized phrase corresponds to one symbol and the multi-phrase rules of the grammar correspond to patterns; words that are not considered

[36]The active or passive status of a verb phrase is determined during the recognition of the phrase according to the auxiliaries participating in it and the tag assigned to the head verb by the tagger.

part of a phrase at the first level correspond to a special symbol that doesn't match any of the patterns. Heads of phrases that can be part of a larger sentence fragment are remembered until a pattern matches or until no patterns that could possibly use this particular phrase remain. This, in particular, means that an intervening unrecognizable construct would make the parser forget previously recognized phrases for the purpose of matching. Given the complexity of natural language sentences, this policy seems reasonable; it keeps the the accuracy of the matching process high and its complexity low, at the expense of missing some partial matches. In fact, as discussed in Section A.4.2, special constraints are also placed on a noun phrase for it to be a candidate subject (and thus the starting phrase of many patterns, cf. Figure A.3). The purpose of these constraints is to limit matches to cases where we can be highly confident of their quality and thus able to base predictions about linguistic properties (e.g., semantic relatedness or unrelatedness) on them.

An important difference with the standard pattern matching algorithms is that matched strings are not allowed to overlap, i.e., each phrase is removed from the list of candidates for matching as soon as it becomes part of a successful match. This effectively assigns at most one syntactic role to each phrase, in accordance with theoretical linguistic principles such as the θ-Criterion of Government-Binding Theory and the Principle of Function-Argument Biuniqueness of Lexical-Functional Grammar [Sells 1987, pp. 37 and 158].

The parser essentially simulates a finite state automaton, or more precisely a high level finite automaton with separate embedded finite automata for each of its states. The use of flat structures by the grammar of Section A.4.2 and the corresponding treatment of prepositions and *wh*-words as noun phrase terminators further simplifies the automata and improves the efficiency of the parser. The use of features does not increase the complexity of the matching process, since the features take values from finite sets and therefore tests on them could be simply replaced with additional, specialized states of the underlying automaton.[37] The parser takes only about 1.63 seconds to process 10,000 words on a Sun SparcStation 10, or about 3.3 times *less* than the time needed to preprocess the corpus and assign part of speech tags to it (Section A.3). Most of this time is actually spent on disk reading and writing operations, since 10,000 words in tagged format correspond on average to a 103 Kbyte text file. The speed of this approach, the simplicity of the grammar, the increased accuracy for the constructs that are retrieved, and the relative rarity of the constructs that are not, justify the decision to limit the definitions of the various recognized phrases to a subset of their possible forms. Other researchers have also successfully used similar approaches in the past. The focus of this line of work has been on the extraction of noun phrases, given the significance of those phrases for indexing, terminology, and information retrieval [Abney 1991; Bourigault 1992; Hobbs *et al.* 1993; Kupiec 1993; Voutilainen 1993; Daille 1994; Ramshaw and Marcus 1995; Evans and Zhai 1996; Federici *et al.* 1996; Aït-Mokhtar and Chanod 1997; Neumann *et al.* 1997]. A simplified definition of noun phrases is typically adopted, so that a noun phrase contains only an optional determiner, an optional sequence of adjectives and a sequence of nouns. Sometimes a single simple prepositional phrase is allowed to postmodify the head noun, but often this is limited to prepositions like *of* in English or *de* in French for which the attachment to the noun can be predicted with high accuracy (e.g., [Kupiec 1993; Daille 1994]).

[37]In fact, the grammar uses only features related to verb phrases (COPULAR, OBJECT-COMPLEMENT, BITRANSITIVE, ACTIVE, PERFECTIVE, PROGRESSIVE, FINITE) in tests, and these features are all binary.

It should be noted that the efficient retrieval of phrases and relationships from the corpus is made possible because of the structure of the grammar of Section A.4.2 and the local character of the disambiguation rules of Section A.4.3. While I have used phrase structure rules to formally describe the grammar, no construct of the grammar contains direct or indirect recursive definitions, except for the apposition of adjectival phrases to noun phrases. In that case, the grammar allows for the recursive inclusion of multiple appositive modifiers; but, in practice, it is very rare for a noun phrase to contain more than one such modifier. The only other kind of repetition allowed by the grammar is simple juxtaposition of words from a specified set of grammatical categories; for example, an uninterrupted sequence of cardinal numbers ($CD+$) in one of the alternates for a noun phrase, or a sequence of adjectives, adverbs, qualifiers, conjunctions, and commas in the definition of adjectival phrases. Tests for determining the applicability of each rule are local, despite the use of features and matching delimiters (for apposition, cf. Figure A.3), since in both cases only a finite set of values for the features and delimiters is allowed. The tests themselves are efficiently computable, since they consist of copying features or unifying atomic values of features. Furthermore, the rules of the grammar (combined with the disambiguation heuristics) exhibit very limited overlap. In fact, the only overlap that exists between rules is when a rule for sentence fragments or noun or verb phrases is a prefix of another rule for the same category; Table A.6 lists all prefix relationships for each of the rules for sentence fragments in the grammar, while the overlap between rules for noun phrases (specifically, between the members of two pairs of alternate right sides for the nonterminal NP in Figure A.3) and verb phrases is again of the (proper) prefix type. But in all these cases, the matching rules have the same structure up to the point where

Rule	Rules that the rule on the left is a prefix of
2	3, 4, 5, 6, 7, 8, 9, 10
3†	4, 6, 8
4†	8
5†	9
6	3, 4, 8
7	5, 9, 10
8	4
9	5
10	None

Table A.6: Prefix relationships among rules for the sentence fragment category (cf. Section A.4.2). The cases marked with † apply only when the rule on the left matches a noun phrase (not an adjectival phrase) as complement.

some of them are discarded. This point can be after one more token (phrase) in addition to those which will be actually matched is considered, as we may need to check if some of the currently viable rules can match a longer sequence of tokens. As a result, at most one token of lookahead is needed, and backtracking never occurs.

Although the use of tests on features may at first indicate that the grammar is context-sensitive, or the use of phrase structure rules that it is at least context-free, the form of the grammar rules and the constraints discussed above show that the grammar is in fact regular and can therefore be implemented by finite state automata without loss of accuracy. As discussed above, one token of lookahead is necessary to ensure that the longest possible match will be found, while an additional token of left context needs to be remembered for the constraints on the subject noun phrase that were described in Section A.4.2.

The grammar and associated disambiguation procedures that have been discussed above have been implemented, and tested on subsets of the Wall Street Journal corpus.[38] The grammar covers most of the syntactic constructs related to adjectives, and many of those related to nouns. It has been used to extract relationship instances for the experiments described in Chapters 4–7 and 9. Example output from the current implementation of the parser is shown in Figure A.7 for one sample sentence.[39] This output was produced with the *verbose* option which identifies each extracted relationship or word feature in a way suitable for human inspection. During actual use of the parser, the output is placed in ten separate files, each containing different types of relationship instances or feature combinations extracted from the text.

A.5 Postprocessing the extracted pairs

The output of the finite-state parser consists of pairs of two words or one word and an associated feature. The words in these pairs are further processed before being passed to the similarity or decision computing modules. The postprocessing stage consists of the following steps:

[38] In line with the arguments presented in Section 1.11, the grammar identifies the extent and features of verb phrases, but does not match verbs to the nouns they are syntagmatically related with. The latter type of information could be added in a future extension if semantic classification of verbs were desired.

[39] This sentence was constructed by combining fragments of several sentences from the Wall Street Journal corpus. This was necessary so that several of the features of the parser be demonstrated within the limited space of the figure.

Input sentence: Many industry experts are critical of FDA's longstanding policy of requiring lengthy and costly tests before drug approval; nevertheless, most consider radical changes unlikely.

Tagged sentence: Many/JJ_1 industry/NN_2 experts/NNS_3 are/BER_4 critical/JJ_5 of/IN_6 FDA/NP_7 's/$\$_8$ longstanding/JJ_9 policy/NN_{10} of/IN_{11} requiring/VBG_{12} lengthy/JJ_{13} and/CC_{14} costly/JJ_{15} tests/NNS_{16} before/IN_{17} drug/NN_{18} approval/NN_{19} ;/$:_{20}$ nevertheless/RB_{21} ,/$,_{22}$ most/QL_{23} consider/VB_{24} radical/JJ_{25} changes/NNS_{26} unlikely/JJ_{27} ./$._{28}$

Parser output:

NP (1–3)
  AP (1–1)
    many [ASPECT=*unknown*, DEGREE=*absolute*, MODIFICATION=*attributive*, NEGATED=−]
  Adjective-noun pair: **many experts** [GENDER=*unknown*, NUMBER=*plural*]

AP (5–5)
  critical [ASPECT=*static*, DEGREE=*absolute*, MODIFICATION=*predicative*, NEGATED=−]
  Adjective-noun pair: **critical experts** [GENDER=*unknown*, NUMBER=*plural*]

NP (7–8)

NP (9–10)
  AP (9–9)
    longstanding [ASPECT=*unknown*, DEGREE=*absolute*, MODIFICATION=*attributive*, NEGATED=−]
  Adjective-noun pair: **longstanding policy** [GENDER=*unknown*, NUMBER=*singular*]

NP (13–16)
  AP (13–15)
    lengthy [ASPECT=*unknown*, DEGREE=*absolute*, MODIFICATION=*attributive*, NEGATED=−]
    costly [ASPECT=*unknown*, DEGREE=*absolute*, MODIFICATION=*attributive*, NEGATED=−]

Figure A.7: Example output of the parser. Subscripts in the tagged sentence indicate token positions and are used for identification purposes in the parser's output. The current implementation only extracts relationships and features relevant to adjectives and nouns. Continued on the next page.

Parser output (continued):

Conjunction: **lengthy costly** [CONJUNCTION=*and*, MODIFICATION=*attributive*, NUMBER=*plural*]
Adjective-noun pair: **lengthy tests** [GENDER=*unknown*, NUMBER=*plural*]
Adjective-noun pair: **costly tests** [GENDER=*unknown*, NUMBER=*plural*]

NP (18–19)

NP (25–26)
AP (25–25)
radical [ASPECT=*unknown*, DEGREE=*absolute*, MODIFICATION=*attributive*, NEGATED=−]
Adjective-noun pair: **radical changes** [GENDER=*unknown*, NUMBER=*plural*]

AP (27–27)
unlikely [ASPECT=*static*, DEGREE=*absolute*, MODIFICATION=*complement*, NEGATED=−]
Adjective-noun pair: **unlikely changes** [GENDER=*unknown*, NUMBER=*plural*]

Figure A.7: Example output of the parser. Continued from the previous page.

- Capital letters in words are converted to their lower case counterparts, with the exception of the initial letter of proper nouns and the subjective personal pronoun *I*.

- Inflected forms of open class words are replaced by their citation forms. In particular, this means that adjectives and adverbs in the comparative or superlative degree are replaced with the absolute degree form of the word and plural forms of nouns are replaced with the singular form. As the current implementation of the data extraction subsystem does not collect information about verbs, no conversions of verb forms are performed. Such conversions would consist of the replacement of participles and finite forms of verbs inflected for person or tense with the infinitive form of the verb, and they would have to be added to the current system if semantic classes or other semantic features of verbs were to be computed.

- All words are checked against a master list of valid words. Words that do not appear in the list, and cannot be derived as a compound of one or more words in the list, are discarded. Extracted word pairs that contain one or more words that fail this test are also discarded.

The rationale for the first two steps is to allow the combination of evidence offered by pairs that are only superficially different. The initial letter of a word or the entire word may be capitalized in some contexts, with sentence-initial position being a prime example, but the semantics of the word does not change from this alteration. Similarly, the changes performed in the second step essentially strip from the word one or more of the following features: DEGREE, NUMBER, PERSON, FINITE, ACTIVE, PERFECTIVE, PROGRESSIVE, and TENSE. While these features affect the semantics of the word in the technical sense (i.e., the extensional denotation of the word changes), they do not affect the *semantic constraints* that the use of the word imposes on other words in the same sentence.[40] As a special case, all inflected forms of the word would belong to the same semantic group with the same orientation and relative order in comparison to other words from the group. Therefore, applying these two steps does not result in loss of information, and yet it increases both the expected and actual counts of the words and word pairs and consequently the accuracy of any statistical method that operates on these counts.

The third step attempts to remove from the set of collected pairs any that contain invalid

[40]With the possible exception of the PROGRESSIVE feature, which also conveys aspectual information [Vendler 1967]. Further investigation is needed to see if removing this feature is as neutral as is the case for the other features listed. But I have left this investigation for the future, as I do not currently use the PROGRESSIVE or ASPECT features in any of the subsequent modules of the system.

words, that is typographical errors which unfortunately exist in any large corpus. Such errors can produce wrong estimates for the frequencies of each pair, but most importantly introduce "unique" words appearing only with one of the words that are analyzed, skewing the comparisons of word pair distributions.

Implementing the first postprocessing step is relatively easy: capital letters in words are converted to their lowercase equivalents, except for the personal pronoun *I/PPSS*[41] and the first letter of words that have been identified as proper nouns by the tagger and assigned the tag *NP*. The accuracy of this conversion procedure depends on the accuracy of the method that identifies proper nouns in the text, especially in contexts where a word could be either a proper noun or some other common word which simply happens to appear with an initial capital letter. The problem is in general a hard one [Palmer and Day 1997; Wacholder *et al.* 1997], and many systems resort to using precompiled lists of names, locations, and other proper nouns (e.g., [Hayes 1994], and treating such lists as a supplementary knowledge source, [Wacholder *et al.* 1997] and [Hatzivassiloglou and McKeown 1998]). In order to avoid this dependence on precompiled, and thus relatively rigid and definitely incomplete, lists, I use a heuristic method that was already incorporated in the part of speech tagger. For words that can be proper nouns or belong to some other class, the method checks the number of times a word appears capitalized in the current discourse unit versus the number of times it appears with a lower case initial letter in the same unit. When the ratio of these two numbers is below a threshold the word is considered not to be a proper noun. By

[41] The part of speech tagger marks *I* as plural, essentially distinguishing only between third person singular and any other number and person combination. This distinction is based on subject-verb agreement constraints.

default, the current paragraph is used as the discourse unit; when newspaper text is used as the input corpus, I modify the method to widen the scope of the algorithm to the entire newspaper article (document).

A rule-based approach is employed for implementing the second conversion step. The system applies to each open class word that is part of an extracted pair a series of morphology rules that detect plural or more generally inflected forms and specify the necessary changes so that the base form is produced. The rules that will be applied and the kind of conversion that is sought are determined from the part of speech that the word has been assigned. The most specific rules are tried first, so that exceptions to more general rules can be easily represented. For example, for the conversion of plural nouns to the singular form, irregular plurals such as *feet* → *foot* are checked first; the most general rule, that a final *-s* should be simply removed, is tried last. Table A.7 lists the rules used for the conversion of plural nouns to the singular form, while Table A.8 lists the rules used for the removal of degree-indicating suffixes from adjectives.

Tables A.7 and A.8 exhibit one important difference: While the rule-based approach is sufficient to uniquely identify the singular form of a noun[42], the situation is more complicated in the case of adjectives.[43] Because of the vowel *e* in the suffixes *-er* and *-est*, several changes in the final part of the stem of the base form may occur during the forward application of the morphology rules that produce a comparative or superlative form. This results in ambiguity when we try to apply these rules in the reverse direction, going from the inflected

[42]Taking into account some exceptions that are not very numerous and can be explicitly listed, as is done in Table A.7.

[43]And also of adverbs, and, I expect, of verbs.

Irregular plurals for specific words
children† → *child*, *dice* → *die*, *feet* → *foot*, *knives*† → *knife*, *leaves*† → *leaf*, *men*† → *man*, *mice* → *mouse*, *oxen* → *ox*, *teeth* → *tooth*, *wives*† → *wife*, etc.

Words borrowed from foreign languages			
Original language	Plural	Singular	Examples
French	~*eaux*	~*eau*	*bureaux* → *bureau*, *chateaux* → *chateau*, *tableaux* → *tableau*
Greek	~*a*	~*on*	*criteria* → *criterion*, *phenomena* → *phenomenon*
	~*es*	~*is*	*crises* → *crisis*, *oases* → *oasis*, *theses*† → *thesis*
Hebrew	~*im*	~	*cherubim* → *cherub*, *kibbutzim* → *kibbutz*
Latin	~*a*	~*um*	*curricula* → *curriculum*, *data* → *datum*, *spectra* → *spectrum*
	~*ae*	~*a*	*alumnae* → *alumna*, *formulae* → *formula*, *personae* → *persona*
	~*i*	~*us*	*nuclei* → *nucleus*, *phagi*† → *phagus*, *stimuli* → *stimulus*
	~*ices*	~*ex*	*indices* → *index*
		~*ix*	*appendices* → *appendix*
Italian	~*i*	~*o*	*mafiosi* → *mafioso*, *paparazzi* → *paparazzo*

Exceptions to the regular morphology rules	
Exception	Examples
Plural word with no corresponding singular form, or word with identical plural and singular forms; no conversion is performed	*calamari*[44], *clergy*, *faculty*, *kiwi*, *mathematics*, *memorabilia*[45], *people*, *physics*†, *series*, *sheep*
Singular word that appears plural; no conversion is performed	*bias*, *chaos*, *ethos*, *gas*, *omen*[46]
Compound receiving a plural suffix in the middle of the word	*passers-by* → *passer-by*
Plural in $\left\{\begin{array}{r}\sim ches\\ \sim ies\\ \sim oes\\ \sim sses\\ \sim uses\end{array}\right\}$ but the singular keeps the final -*e*	*cliches* → *cliche*, *niches* → *niche* *calories* → *calorie*, *zombies* → *zombie* *foes* → *foe*, *shoes*† → *shoe* *impasses* → *impasse*, *posses* → *posse* *fuses* → *fuse*, *houses*† → *house*
Plural in ~*es* and the rules below predict a final -*e* that the singular form of the word doesn't have	*gases* → *gas*, *lenses* → *lens*, *summonses* → *summons*
Plural in ~*is*, singular in ~*i*	*martinis* → *martini*, *taxis* → *taxi*
Plural in ~*us*, singular in ~*u*	*bayous* → *bayou*, *menus* → *menu*

Table A.7: Reverse morphology rules for plural-to-singular noun conversion. Rules are tried from most to least specific (top to bottom). Words marked with † are also used as suffixes, so that *men*† → *man* also implies that *policewomen* → *policewoman*, and *theses*† → *thesis* also implies *hypotheses* → *hypothesis*. Continued on the next page.

[44] The Italian singular, *calamaro*, does not appear in English.

[45] The Latin singular, *memorabilis*, does not appear in English.

[46] *Omen* is listed as an exception (along with other words such as *abdomen* and *regimen*) because otherwise it would be matched to the rule *men* → *man*, since *men* is considered a plural suffix.

Rules corresponding to regular plural formation		
Plural	Singular	Examples
$\sim Xhes$	$\sim Xh$, if X is c or s	*approaches* → *approach*, *torches* → *torch*, *flashes* → *flash*, *wishes* → *wish*
	$\sim Xhe$, otherwise	*catastrophes* → *catastrophe*, *lathes* → *lathe*, *scythes* → *scythe*
$\sim Xies$	$\sim Xy$	*agencies* → *agency*, *babies* → *baby*, *penalties* → *penalty*
$\sim Xses$	$\sim Xs$, if X is s or u	*classes* → *class*, *guesses* → *guess*, *buses* → *bus*, *octopuses* → *octopus*
	$\sim Xse$, otherwise	*bruises* → *bruise*, *nurses* → *nurse*, *responses* → *response*
$\sim Xes$	$\sim X$, if X is o or x	*heroes* → *hero*, *mosquitoes* → *mosquito*, *boxes* → *box*, *faxes* → *fax*
	$\sim Xe$, otherwise[47]	*apples* → *apple*, *gloves* → *glove*, *measures* → *measure*
$\sim is$, $\sim js$, $\sim qs$, $\sim ss$, $\sim us$, $\sim xs$, or $\sim zs$	No conversion is performed[48]	*arthritis*, *basis*, *bourgeois*, *chassis*, *iris*, *precis*, *tennis*,[49] *bliss*, *boss*, *cross*, *darkness*, *dress*, *kiss*, *loss*, *progress*, *bonus*, *calculus*, *corpus*, *focus*, *hiatus*, *surplus*, *virus*[50]
$\sim s$	$\sim$	*areas* → *area*, *ports* → *port*, *relics* → *relic*, *ships* → *ship*

Table A.7: Reverse morphology rules for plural-to-singular noun conversion. The symbol X in the rules stands for any single letter. Continued from the previous page.

form to the citation form. In many cases, it cannot be decided which was the citation form, since without a lexicon several base forms look plausible. For example, *bigger* could have been produced from *big*, *bigg*, or *bigge*. Some of the alternate right sides of the rules in Table A.8 are rarely used and could have been replaced with specific lexical exceptions; this is the case for the rules $\sim ie(r/st) \rightarrow \sim ie$ and $\sim VCCe(r/st) \rightarrow \sim VCCe$. However, this is not possible for other sources of ambiguity, namely selecting $\sim VC$ or $\sim VCC$ as the replacement for $\sim VCCe(r/st)$, and even more importantly, selecting $\sim e$ or $\sim$ as the replacement for

[47] Note that the suffixes *-hes*, *-ies*, and *-ses* have already been covered by previous rules.

[48] No words ending in *-js*, *-qs*, *-xs*, or *-zs* appear in the 1987 Wall Street Journal corpus. I assume that if any such words are ever encountered, they should not be converted as they are probably loans from foreign languages (or typographical mistakes).

[49] Most singular words ending in *-is* are loans from Greek or French.

[50] Most singular words ending in *-us* are loans from Greek or Latin (primarily the latter).

Irregular comparatives/superlatives for specific words
better / best → *good*, *farther / farthest / furthest* → *far*[51], *worse / worst* → *bad*

Words that appear inflected but really are a base form
• *bitter*, *cavalier*, *clever*, *dapper*, *elder*, *former*, *foster*, *further*, *eager*, *inner*, *kosher*, *lavender*, *latter*, *lesser*, *limber*, *meager*, *nether*, *other*, *outer*, *premier*, *proper*†, *sheer*, *silver*, *sinister*, *slender*, *sober*, *somber*, *tender*, *upper*[52], *utter*, *volunteer*, *wicker*
• *east*†, *honest*†, *modest*†, *manifest*, *west*†

Rules corresponding to regular comparative/superlative formation		
Inflected form	Base form	Examples
$\sim ie(r/st)$	$\sim y$	*angrier* → *angry*, *earlier* → *early*, *funnier* → *funny*, *rosier* → *rosy*
	$\sim ie$	*eerier* → *eerie*
$\sim ee(r/st)$	$\sim ee$	*freer* → *free*, *weer* → *wee*
$\sim VCCe(r/st)$	$\sim VC$	*bigger* → *big*, *hotter* → *hot*, *slimmer* → *slim*, *thinner* → *thin*
	$\sim VCC$	*crosser* → *cross*, *duller* → *dull*, *odder* → *odd*, *stiffer* → *stiff*
	$\sim VCCe$	Although adjectives in $\sim VCCe$ exist in English (e.g., *bizarre*, *brunette*, *passé*), they normally do not appear in forms inflected for degree.
$\sim e(r/st)$	$\sim e$	*braver* → *brave*, *cuter* → *cute*, *ruder* → *rude*, *sparser* → *sparse*
	$\sim$	*bolder* → *bold*, *darker* → *dark*, *louder* → *loud*, *warmer* → *warm*

Table A.8: Reverse morphology rules for removing degree inflections from adjectives. While the examples use the suffix *-er* of the comparative form, exactly the same rules apply for the suffix *-est* of the superlative form. Words marked with † are also used as suffixes, so that listing *proper* and *honest* as absolute degree adjectives implies that *improper* and *dishonest* are not inflected either. More specific rules (towards the top of the table) are applied before less specific rules (towards the bottom of the table). The symbols V and C in the general rules stand for any vowel and any consonant respectively. Repeated occurrences of these symbols in the same rule refer to the same letter. In particular, the sequence CC corresponds to a double consonant (e.g., *dd* or *gg*).

$\sim e(r/st)$. The great majority of English adjectives that have single-word comparative and superlative forms fall in this last category and both candidate absolute degree forms look equally plausible without access to a dictionary.

[51] *Further* is ignored on purpose, since it is most commonly used as a synonym of *more* or *additional*, and not as a comparative form of *far*; e.g., *Any further questions?* *Lesser* and *least* are also ignored with a similar rationale. These forms are frequently used not as intensified forms of *little*, but rather as general diminishers (e.g., *lesser men* means less important or able men, not men of smaller size).

[52] *Lower* is a somewhat problematic case, as it can be used as an independent adjective (e.g., *the lower*

I solve this problem by observing that while this ambiguity exists and cannot be resolved locally at the (reverse) morphology rules level, it goes away when the problem is considered at a more general level: it is highly unlikely that more than one of the candidate base forms will actually exist in the language. So the final disambiguation is delayed until all base forms of adjectives have been observed in the corpus; then we can select the unique base form for each ambiguous inflected form by counting the occurrences of each candidate base form in the corpus and selecting the one with non-zero frequency.[53]

Turning to the problem of removing the degree-indicating inflection from adverbs, I observe that there are four ways according to which the comparative and superlative may be formed for adverbs. Most adverbs, namely those derived from adjectives with the addition of the suffix *-ly*, have only periphrastic comparative and superlative forms, i.e., multiword forms constructed with the addition of the modifying adverbs *more* or *most*; for example, *more vividly* or *most appropriately*. Some adverbs, however, form single-word comparative and superlative forms according to one of the following patterns:

- Irregularly (e.g, *badly* → *worse*/*worst*).
- By adding the suffixes *-er* or *-est* to the base form, when there is an adjective identical in form to the adverb. For example, *fast* → *faster*/*fastest*, *slow* → *slower*/*slowest*.
- By adding the suffixes *-er* or *-est* to the base form of the adjective from which the ad-

chamber of the legislature) in addition to being the comparative form of *low*. I have elected to convert occurrences of *lower* to *low* since the second case seems to be more frequent in the corpus.

[53] For robustness, the form with the highest count is selected. Spurious occurrences of a non-existent form may be retrieved from the corpus (e.g., *bigg* can appear in the corpus because of a typographical error), but the counts of such spurious forms will be much smaller than the count of the correct form.

verb is itself derived. For example, *easily* → *easier*/*easiest*, *poorly* → *poorer*/*poorest*.[54] The acceptability of such inflected forms depends on the particular adverb and also varies from speaker to speaker.

No conversion is necessary (or possible) for adverbs that have only periphrastic comparative and superlative forms, and the set of adverbs that belong in the first two of the three groups with single-word comparative and superlative forms is small and stable. This leaves the last case (adverbs forming the comparative and superlative from the corresponding adjective), which is a somewhat open class, although only few of the adverbs inflected in this way will be clearly acceptable. I decided to incorporate in the system a list of adverbs that have this last type of comparative and superlative forms, along with their corresponding base forms. This simplifies the rules for removing the degree inflection from adverbs, since in all cases we can consult fixed lists that specify the absolute degree form. The decision is also not likely to result in significant loss of information, since few adverbs not belonging to the list are likely to appear with single-word inflected forms; if such forms do appear, they are likely to be of questionable acceptability. The alternative would be to consider all adverbs ending in *-er* and *-est*, retrieve the base forms as if they were adjectives (see Table A.8 and the discussion of related statistical techniques above) and then apply forward morphology rules adding the suffix *-ly* to the retrieved adjectives to produce the corresponding absolute

[54] A related phenomenon is the "adverbization" of adjectives, which has become quite common, especially in spoken American Engish and despite being criticized. In many contexts, the adjective replaces the corresponding adverb, often producing idiomatic expressions where the original and syntactically more appropriate adverb is not as current as the adjective; for example, *take it easy*. *Good* has practically displaced *well* in many contexts where an adverb is expected, e.g., as a response to the colloquial greeting *How are you (doing)?* Other adjectives, such as *clear*, have acquired almost full adverb status. It is interesting to note that *clear* is listed as an adverb in American dictionaries, but not in some British ones such as COBUILD [Sinclair 1987b]; Quirk *et al.* [1985, p. 465] specifically use *clear* as an example of an adjective that is used incorrectly in a context that requires an adverb.

Irregular comparatives/superlatives for specific adverbs	*better/best* → *well*, *farther/farthest/furthest* → *far*[55], *worse/worst* → *badly*
Adverbs identical to an adjective with which they share the comparative and superlative forms	*earlier* → *early*, *faster* → *fast*, *harder* → *hard*†, *later* → *late*†, *longer* → *long*, *lower* → *low*[56], *sooner* → *soon*[57]
Adverbs sharing the comparative and superlative forms with the adjective they are derived from	*closer* → *closely*, *easier* → *easily*, *harsher* → *harshly*, *heavier* → *heavily*, *milder* → *mildly*, *poorer* → *poorly*, *rougher* → *roughly*, *smarter* → *smartly*, *smoother* → *smoothly*, *thinner* → *thinly*
Adverbs having two base forms, one identical to the adjective they are derived from and another regularly inflected with the suffix *-ly*. The comparative and superlative forms are shared with the corresponding adjective.	*clearer* → *clearly*[58], *deeper* → *deeply*, *hotter* → *hotly*, *higher* → *highly*, *slower* → *slowly*, *tighter* → *tightly*, *quicker* → *quickly*, *sharper* → *sharply*

Table A.9: Reverse morphology rules for removing degree inflections from adverbs. While the rules use the suffix *-er* of the comparative form, exactly the same rules apply for the suffix *-est* of the superlative form (except for the first category of irregular adverbs). The rules are expressed in the form of lists of adverbs to which the specified conversion should be applied. If none of the above rules applies, no conversion of the adverb is performed. This in particular includes adverbs that end in *-er* or *-est* and therefore appear inflected for degree but are really absolute degree forms; e.g., *however*, *west*. Adverbs marked with † also have a ∼*ly* form, but with very different meaning.

degree adverbs. As discussed above, the anticipated benefits from these transformations do not appear to warrant the effort needed for their implementation.

Table A.9 lists the rules that the system applies to determine if any of the three cases of

[55] *Further* is not converted to the base adverb *far* for the same reasons that apply when it is used as an adjective; see footnote 51 on page 552. Also, the following irregular adverbs are not converted to the absolute degree form, since they are principally used as generic qualifiers (diminishers or intensifiers): *less*, *least*, *more*, and *most*.

[56] The form *lowly* exists, but is an adjective, not an adverb. It also has different (although somewhat related) meaning from the adverb or adjective *low*.

[57] There is no adjective *soon*; nevertheless, I have listed *soon* in this category as it forms the comparative and superlative exactly as if there were a regularly inflected adjective *soon*. *Soon* is the only adverb exhibiting this behavior.

[58] The alternative *clear* appears as an adverb primarily in American English. See footnote 54 on the previous page.

single-word inflected adverb forms match a particular adverb, in the form of categories of adverbs to which the specified conversion should be applied. Note that there is a potential overlap between the categories given in Table A.9, corresponding to the case where an adjective can produce both an identical adverb and an adverb with the suffix *-ly* added, e.g., *deep* and *deeply*. Frequently, such pairs of adverbs are identical in meaning (e.g., *slow* and *slowly*), but sometimes there are subtle differences, as in the case of *deep* and *deeply*. This ambiguity is automatically handled by the reverse morphology subsystem, which selects the alternative that is most generally used in the language. In all these cases, the ∼*ly* form is far more frequent in the 1987 Wall Street Journal corpus than the corresponding uninflected adverb form.

Finally, for the third step that filters misspelled words, I have implemented a module that uses a finite-state automaton to properly recognize numbers and analyze hyphenated compounds into their constituent simpler words (i.e., words containing only alphabetic characters). These simpler words are checked with the UNIX *spell* program, which in turn consults a sizable word list (about 25,000 words in most installations) plus a set of simple inflection and affixation rules. A subsequent program identifies the words in the originally extracted pairs that match any word not recognized by the *spell* program, and discards any pair containing one or more such words. Matching for this purpose is defined as follows: Two simple words match only if they are identical, while a hyphenated compound matches all its constituent simple words.

I performed an analysis of the effects of this step for the set of pairs of an adjective and a noun in a modification relationship extracted from the 1987 Wall Street Journal corpus according to the methods described in Section A.4. After applying the first two postprocessing steps discussed in the present section (conversion to lower case and to citation forms), this set contains 935,371 occurences of 276,239 distinct pairs; the pairs themselves are formed from 40,553 distinct words and hyphenated compounds that occur 1,870,742 times within the collection of extracted pairs. These words are analyzed by the matching algorithm into 25,260 simple words, of which 4,207 (16.65%) are considered mistakes by *spell*. This, in turn, leads to the rejection of 5,268 distinct words and compounds, which occur 17,533 times in this collection of pairs and appear in 11,462 distinct (17,369 total) pairs. In addition, 210 distinct words (with 218 occurrences in the set) appear only in pairs that contain an unrecognized word, and therefore are also removed. A further 16,987 occurrences of valid words are removed because they belong in pairs where the other member is not recognized. This leaves a total of 35,075 distinct words occurring 1,836,004 times in the filtered set (out of 1,852,991 times in the original set), and appearing in 264,777 distinct (918,002 total) pairs in the filtered set.

The results of this distributional analysis are presented in Table A.10. Note that 12.99% of the word *types* are not recognized by the spelling checking algorithm, and the removal of words that only appear in pairs that contain unrecognized words causes a total of 13.51% of the original word types to be discarded. However, these discarded words account only for 1.86% of the word *tokens* and total pairs in the original set, and only for 4.15% of the number of distinct pairs. Actually, only 0.94% of the total number of words (tokens) occuring in the original set are removed as mistakes; approximately the same number of word tokens is removed because they are not useful any more (i.e., their corresponding word in the pair

		Types		Tokens	
Pairs	In the original collection	276,239		935,371	
	Containing one or two unrecognized words	11,462	(4.15%)	17,369	(1.86%)
	Survived in the filtered collection	264,777	(95.85%)	918,002	(98.14%)
Words	In the original collection	40,553		1,870,742	
	Removed as mistakes	5,268	(12.99%)	17,533	(0.94%)
	Recognized but removed because they appeared with an unrecognized word	210	(0.52%)	17,205[59]	(0.92%)
	Survived in the filtered collection	35,075	(86.49%)	1,836,004	(98.14%)

Table A.10: Counts of words and pairs removed by the spelling checking module. The initial collection of pairs contained all adjective-noun pairs retrieved from the 1987 Wall Street Journal corpus, after previous postprocessing steps (conversion to lower case and morphological reduction) had been applied.

has been removed as a mistake). The much smaller percentage of unrecognized words when tokens are considered instead of types indicates, in agreement with intuitive expectations, that unrecognized words are more rare than recognized words. Each unrecognized word appears in this set 3.33 times on average, in contrast to 52.52 occurrences on average for each recognized word.

Figure A.8 presents a small representative sample of the 5,268 word types that were unrecognized by the spelling cheking algorithm. This sample was obtained by selecting every 200th word in an alphabetical list of the rejected types. As the figure indicates, discarding words that are not recognized by the spelling checking module is actually a double-edged sword. It removes typographical errors but may also inadvertently remove

[59] Of the 17,205 word tokens in this category, only 218 correspond to the 210 types removed because they never appeared in a pair with another recognized word. The remaining 16,987 tokens are instances of valid, recognized types; these instances had nevertheless to be removed because they occurred in pairs where the other member of the pair was not recognized.

Amerongen	ESOP	Milan	shee
Asian-made	Florida-owned	neanderthal	stepchildren
birddog	Gorham	offeror	Tibetan
Cajun-style	homogenous	penseroso	unusal
Cinderella	Iraqi-sought	pre-death	woodfiber
crackup	Kuwaiti	Pritzker-controlled	
discernable	magazinecover	risotto	

Figure A.8: Sample of word types rejected by the spell-checking module.

valid words that are too rare or specialized to be included in *spell*'s wordlist.[60] Indeed, of the 26 types in the sample, five are typographical mistakes with missing or replaced characters (e.g., *homogenous* instead of *homogeneous*, *discernable* instead of *discernible*, and *unusal* instead of *unusual*), and four are compounds where a space or hyphen is required as separator but not present (e.g., *crackup* instead of *crack up* or *crack-up* and *magazinecover* instead of *magazine cover*). The spell checking procedure also correctly rejected three proper nouns (e.g., *Amerongen*, *Pritzker*) and one abbreviation (*ESOP*, for *E*mployee *S*tock *O*wnership *P*lan) that were misclassified as common nouns by the tagger, as well as a foreign word in quoted material (*penseroso*, from Italian). But in addition to these correct decisions, eight valid words are rejected because *spell* recognizes them only with the initial letter capitalized (e.g., *Asian-made*, *Florida-owned*, and *neanderthal*); two compounds (*pre-death* and *stepchildren*) are rejected because *spell* expects (or does not expect) a dash as a separator within these particular compounds; and two words are not in *spell*'s vocabulary at all (*Kuwaiti* and *risotto*).

Consequently, a significant percentage of the word types rejected by the spell-checking

[60] Or any static list for that matter.

module should have been kept; this percentage is 46.15% in this sample. Note that this doesn't mean that the error rate of the module is the same when word *tokens* are considered, since, as noted above, the average frequency of rejected words is much lower than the average frequency of the words that pass the spell-checking test. Nevertheless, the effect of this filtering step may not always be a positive one. An answer to the question of the effectiveness of this step is given in Sections 5.3.1–5.3.2, where results of a comparative quantitative analysis of the performace of the word grouping subsystem with and without the spell-checking module were reported. This analysis shows that, on average, spell-checking has a slight positive effect for the word grouping task.

A.6 Extracting additional indicators from the corpus

In Sections 6.2, 7.3, and 8.2 I presented several indicators for the linguistic properties of gradability, semantic orientation, and semantic markedness that relate to the further processing and analysis of the semantic groups formulated by the first stage of the system. These indicators are summarized below:

- Appearance of adjectives in the comparative or superlative form and modification of adjectives by grading adverbs, qualifiers, and other constructs that indicate degree (see Figure 6.1 on page 248). These indicators are used to determine the gradability of adjectives (Sections 6.2–6.3).

- Conjunctions between adjectives, which are used for predicting whether the conjoined adjectives have the same or different semantic orientation (Sections 7.3 and 7.5).

- Value of the NUMBER and MODIFICATION features (see Section A.4.2) for conjoined adjectives, and explicit modification of conjunctions by *both*. These are used to stratify the information from conjunctions according to the syntactic context for the purpose of distinguishing between orientation values (Sections 7.3–7.5).

- Number and text frequency of words derived from adjectives (several variants according to how "derived" is defined), length of adjectives in characters, number of syllables in adjectives, and number of homographs of an adjective. All these measures are used to determine whether an adjective is semantically marked or not (Sections 8.2–8.4 and 8.7–8.8).

- Existence of a morphological relationship between two adjectives (more specifically, derivation of an adjective from another via the addition of a negative prefix), which is used to detect a difference in the semantic orientation of the two related adjectives (Sections 7.3 and 7.5).[61]

- Text frequency of adjectives, which is used for determining semantic markedness values (Sections 8.2, 8.4, and 8.9) and assigning orientation labels to groups of adjectives separated according to orientation (Section 8.10).

---

[61] Although this indicator is also relevant to semantic markedness and thus to the direct determination of semantic orientation values, it is subsumed in that case by the more general morphological complexity criteria (i.e., length in characters and number of syllables). Therefore, I do not explicitly use this indicator for the latter task.

Of these indicators, text frequency and the number of homographs of given words can be measured easily from the tagged corpus, while the length of a word in characters can be computed immediately. Conjunctions of adjectival phrases and constructs modifying adjectives are extracted by the methods of Section A.4; the retrieved modifiers of adjectives can be compared to a predetermined list of grading modifiers (Figure 6.1 on page 248) for the special case of gradability indicators. Occurrences of *both* as a premodifier of adjectival conjunctions are identified by the parser. The parser also computes features for the extracted adjectives, including the features NUMBER and MODIFICATION mentioned above. Finally, the degree of an adjective is extracted from the text and encoded in the DEGREE feature as a by-product of the conversion to the absolute degree form that takes place according to the rules of Table A.8 on page 552 (Section A.5).

This leaves us with three indicators which cannot be computed from the information collected in the parsing stage or with other simple methods. These indicators are the number of syllables of an adjective, the existence of a morphological relationship between two adjectives, and the various numbers and frequencies of words derived from an adjective.

Finding the number of syllables in a word is a hard task. I decided to postpone the implementation of a fully automatic method for this problem until I investigate further the usefulness of this indicator for the determination of semantic markedness. In order to perform this investigation, I used stored values of this variable extracted from an extensive machine-readable annotated word list, the MRC Psycholinguistic Database [Coltheart 1981; Wilson 1987]. This database contains a variety of measurements for 150,837 entries, or 115,331 distinct citation forms if we count words that differ in part of speech but are identical in form as instances of the same citation form. The exploratory data analysis for the usefulness of indicators of semantic markedness (see Section 8.4) has shown that predictions of markedness made according to the number of syllables test can be replaced with predictions made according to the simpler test based on word length in characters, with only a small change in accuracy. Furthermore, the test based on the number of syllables is not particularly useful for the overall determination of semantic markedness, to the point that the difference between predictions based on it and random assignments of words to the marked and unmarked categories is not statistically significant, even at the 5% level, when there is no morphological relationship between the two paired words that we analyze. The number of syllables in a word was not selected as an informative variable in the overall composite predictor of semantic markedness (see Section 8.7), and is not used by the system when orientation labels are assigned to groups of adjectives of opposite polarity. Consequently, there is no need for an automatic method that independently computes this information.[62]

The second remaining indicator involves the problem of determining whether two adjectives are morphologically related, in the sense that one of them is derived from the other with the addition of a negative prefix. I solve this problem by using a derivational component and a list of negative prefixes for English adjectives. Several of these prefixes can apply in principle to any adjective (e.g., *dis-*, *un-*), while others are restricted to particular

[62]Since the MRC database is quite extensive, an independent method for determining the number of syllables of a word would only serve the point of being consistent with the philosophy stated throughout the dissertation, namely that the system should avoid dependence on precompiled lexical knowledge sources (Section 1.5). Yet, many of the arguments against precompiled knowledge do not apply in this case, as the number of syllables of a word is not a subjective measure and does not change from domain to domain. In any case, the goal of independence from lexical knowledge sources is not a constraint here as the number of syllables is not used in the actual system.

Prefix	Apparent form	Required right context	Examples
a-	*a-*	First letter consonant	*apolitical*, *asymmetric*, *atonal*, *atypical*
	an-	First letter vowel or *h*	*anerotic*, *anhydrous*
anti-	*ant-*	First letter vowel	*antarctic*
	anti-†	No restriction[63]	*anticlimactic*, *antisocial*, *antiviral*
contra-	*contra-*†	No restriction	*contrafactual*
counter-	*counter-*†	No restriction	*counterintuitive*, *counterproductive*
dis-	*dis-*	No restriction	*discontinuous*, *disobedient*, *disreputable*
in-	*il-*	First letter *l*	*illegal*, *illiterate*, *illogical*
	im-	First letter labial (*b*, *m*, or *p*)	*immature*, *impartial*, *implausible*, *impure*
	in-	First letter not *b*, *l*, *m*, *p*, or *r*	*inarticulate*, *incapable*, *ineffective*, *insensitive*
	ir-	First letter *r*	*irrational*, *irregular*, *irreversible*
mis-	*mis-*	No restriction	*misrepresentative*, *mistrustful*
non-	*non-*†	No restriction	*noncommittal*, *nonlinear*, *nonviolent*
un-	*un-*	No restriction	*unaware*, *unhappy*, *unlawful*, *unsafe*

Table A.11: Negative prefixes employed in the formation of adjectival antonyms. The prefixes *anti-* and *non-* frequently occur with an explicit dash separating them from the base adjective, especially if the resulting word is long; for example, *anti-inflationary*, *non-discriminatory*. To account for this, the derivation algorithm allows an optional dash to appear between the prefix and the base for the prefix forms marked with †. *Non-* is frequently employed to produce non-gradable adjectives which express contrast (e.g., *nonalcoholic*, *nonresidential*). The use of *non-* may result in a non-gradable adjective even if the original adjective is a scalar one (e.g., *non-empty*, *nonlethal*, *nontoxic*); compare for example *undemocratic* (which is gradable) with *nondemocratic* (which is not). The prefixes *dis-*, *in-* (in its various manifestations), *non-*, and *un-* are the most productive. The prefix *extra-* has been omitted from the table because its semantic function is ambiguous. Although it sometimes indicates negation (e.g., *extracorporeal*, *extraordinary*, *extraterrestrial*), it can also indicate "excessive degree" (e.g., *extra-lean*, *extra-efficient*) or "being out of" (e.g., *extracurricular*, *extramarital*, *extraparliamentary*).

morphological contexts, since they place constraints on the first letter of the base word (e.g., the various forms of *in-*: *il-*, *im-*, *in-*, and *ir-*). Table A.11 presents the negative pre-

[63] Although *anti-* can combine with a word starting with a vowel (e.g., *antierotic*), an explicit dash is usually used in such cases.

fixes that are typically used in English for deriving an antonymous adjective from another adjective. The derivation algorithm starts from the shorter of the two words, and goes through the sequence of negative prefixes of Table A.11, adding each prefix form to that word if the morphological restrictions associated with the prefix allow this operation. For several of the prefixes, marked with † in Table A.11 (e.g., *anti-* and *non-*), the algorithm allows an optional dash between the prefix and the shorter adjective. If one of these attempts to derive the second, longer adjective is successful, the algorithm stops and reports that the two adjectives are morphologically related. Otherwise, when the list of prefixes is exhausted, the algorithm reports the two adjectives as morphologically unrelated. This method is highly successful in detecting morphological relationships whenever they exist, primarily because the morphological processes that can give rise to such pairs of adjectival antonyms are limited in number and simple to simulate. Conversely, the method is also quite accurate and rarely reports false matches, as very few pairs of adjectives that differ only in the presence of an apparent negative prefix are not also associated semantically according to the transformation of meaning implied by the prefix. While it is possible for adjectives to start with a sequence of characters identical to one of the prefixes in Table A.11 without truly containing a negative prefix, this does not present a problem in practice as usually there will be no adjective that could be mistakenly considered as the corresponding base form. This is the case for adjectives such as *contradictory*, *counterfeit*, *disciplinary*, *illustrative*, *immune*, *incendiary*, *mischevious*, *nondescript*, and *unruly*; in all these cases, the corresponding base forms that could have been predicted from Table A.11 (i.e., **dictory*, **feit*, **ciplinary*, **lustrative*, **mune*, **cendiary*, **chevious*, **descript*, and **ruly*) do not exist in English.

Of course, occasionaly the analysis of a pair will be mistaken. For example, the derivational component would erroneously detect an antonymy relationship between the two synonyms *flammable* and *inflammable*, where the initial *in-* would be interpreted as a productive negative prefix. It is interesting to note that adjectives that satisfy this morphological relationship but not the associated semantic relationship of opposition are so rare that even native speakers misinterpret them as antonyms. Consider the following example[64]: *BR switched from highly flammable to inflammable but slow-smouldering material ... They rightly discarded the highly flammable materials and went for those which were non-flammable with a low combustion point.* The writer of the article appears to incorrectly assign negative meaning to the prefix *in-* in *inflammable*, viewing *inflammable* as a synonym of *non-flammable* and as an antonym of *flammable*.

The final indicators that need to be extracted from the corpus are the numbers of distinct derived words and compounds for each adjective and their frequencies. The problem here is not the computation of these numbers or frequencies, which can be accomplished by counting the occurrences of such words in the corpus, but the determination of the set of English words that are in fact derived from each adjective or are compounds that include a given adjective. I consider the problems of identifying derived words and compounds separately, given that there are some known restrictions for the derivation process (i.e., the set of productive affixes is more or less fixed), while words can generally combine freely to produce compounds.

[64] *Observer*, 9 July 1978. Cited in [Quirk *et al.* 1985, p. 1541].

To address the problem of finding the derived words, I use again a derivational component and encoded morphological knowledge, in the form of lists of prefixes and suffixes. However, identifying such derived words is much more complicated than the previously discussed problem of determining if two adjectives are morphologically related, for the following reasons:

- The set of possible prefixes and suffixes is much larger, since all possible grammatical categories and semantic relationships must be covered for the produced words.
- The derivation process may include multiple affixation steps, not just the addition of one prefix.
- Many suffixes (and, more rarely, some prefixes) involve possible morphological alterations to the base form; for example, *heavy* + *-ness* = *heaviness*, not **heavyness*.
- I need to answer many more questions about morphological relatedness. When I looked for morphological relationships among N adjectives, I had to consider $\frac{N \cdot (N-1)}{2}$ pairs of words. For 1,000 adjectives, this number is equal to 499,500. If the investigation of morphological relationships is performed within each constructed semantic group, the number of derivational tests between adjectives becomes even less, i.e.,

$$\sum_i \frac{n_i \cdot (n_i - 1)}{2}$$

  where i goes over all formed semantic groups and n_i is the number of elements in group i ($N = \sum_i n_i$). Assuming that the 1,000 adjectives were placed in 200 groups of 5 words each, the number of word pairs analyzed would be equal to 2,000. But for the current problem, I need to examine $N \cdot (V-1)$ pairs, where N is the total number of adjectives as before and V is the total size of the vocabulary of the language. Taking $V = 100,000$ and $N = 1{,}000$ as before, we come up with 99,999,000 questions instead of 499,500 or 2,000.

Because of these considerations, the approach of the exhaustive search where all possible affixation rules are applied to an adjective and all formed "words" are checked against the vocabulary is no longer practical.[65] Instead, I use a more sophisticated approach that reverses the direction of the search, starting from each word in the vocabulary in turn and applying the affixation rules *in reverse*, until no more such reverse rules can be applied. During the application of these reverse rules, candidate words are formed and checked against the list of adjectives of interest. In this way, each word in the vocabulary at large is considered only once, only valid rules are applied, and (mostly) valid resulting word forms are considered. Typically, very few rules will match any particular word, and for many words (i.e., the ones that do not contain affixes) no rules will be applicable. Naturally, a word may lead to more than one adjectives during the application of this backwards derivation procedure. For example, if one adjective is derived from another one, all word forms that are derived from the former will be also considered as derived from the latter (e.g., *ingenuous* $\rightarrow$ *disingenuous* $\rightarrow$ *disingenuously*). For all these adjectives, the count of distinct derived words would be increased by one and the total frequency of derived words

[65]This approach has another problem, namely that it is not immediately clear when the search should stop, since now multiple applications of affixation rules are allowed. We can, however, heuristically stop the search after a given number (e.g., five) of affixation rules have been applied or after a given word length has been exceeded. In particular, we can provide a stopping condition that guarantees correctness if we stop only when the length of the longest word in the vocabulary has been reached or exceeded.

would be incremented by the frequency of the word currently under consideration.

Sometimes more than one reverse affixation rules may apply. For example, upon seeing the word *periscope*, the rules indicating the removal of the prefix *per-*[66] or the prefix *peri-*[67] match and could be applied to produce shorter morphologically related words. In such cases, a depth-first search is performed. The derivational component attempts to apply the rule that results in the removal of the longest affix first, then processes the resulting word until no more affixes can be removed. Irrespective of whether this path is fruitful or not (i.e., matching adjectives are found), the algorithm will backtrack to previous choice points and apply the remaining matching rules in order to retrieve other possibly matching adjectives. In this way, *binational* will be correctly analyzed as $bi + national$ using the Latin prefix *bi-*, meaning *twice*. The alternative analysis, $bin + ational$, which employs the closely related Latin prefix *bin-* with meanining similar to *bi-*, is initially preferred since a longer prefix is removed, but ultimately rejected since no word **ational* exists. An exception to this rule is that if different sequences of rules result into the removal of the same affix (e.g., the sequence of prefixes *di-*[68] and *a-*[69] versus the single prefix *dia-*[70]) then only one sequence (with the minimum number of steps) is applied. This is justified because I am not interested in the way the word was formed from an adjective or in the meaning of the matched affixes, but on whether there is any sequence of derivational rules that can lead from each adjective to the currently considered word.

[66] From the identical Latin preposition, meaning *through* (cf. *perlocutionary*) or *for each* (cf. *per capita*).

[67] From the identical Greek preposition, meaning *around*, as in this example of *periscope* or in *periodontal*.

[68] From Greek; meaning *two*, as in *dilemma*, *di-* + *lemma* (proposition, assumption).

[69] A negative prefix from Greek; see Table A.11.

[70] A Greek prefix meaning *through*; e.g., *diameter*, *diaphanous*, *diaphragm*.

Note that the number of words where more than one affixation rule applies is small, and this ambiguity could have been resolved by selecting only the rule which leads to the removal of the longest affix. Such a decision is correct in most cases. However, we still need the depth-first search because *every* potential application of a rule is ambiguous. The ambiguity is present because we cannot know if the sequence of letters that appears as a prefix or suffix is indeed a productive affix or if the word just happens to start or end with these letters.[71] For example, when analyzing the words *perilously* or *perishable*, we may first consider the initial *peri-* as a prefix. When no words can be produced after such a reduction, and after the subsequent attempt to remove a prefix *per-* also fails, we have to return back and treat *peri-* as part of the base word.

Tables A.12 on pages 571–577 and A.14 on pages 579–585 list the 106 rules employed for the removal of prefixes and the 132 rules employed for the removal of suffixes respectively. Inflections (e.g., *-ed* for verbs and *-s* for nouns) are treated as suffixes for the purposes of this component. Note that many suffixes (and some prefixes that effect a conversion from one grammatical category to another) really apply only to specific grammatical categories. For example, *-ity* applies to adjectival bases to produce (usually abstract) nouns as in *mature* → *maturity* and *stable* → *stability*, and *-ee* applies to verbal bases to produce concrete nouns as in *employ* → *employee* and *trust* → *trustee*. Similarly, the prefix *en-* applies to nouns and produces verbs, e.g., *courage* → *encourage* and *title* → *entitle*.[72] However, the grammatical category of the word is not traced by the derivational component, and con-

[71] Recall the related discussion of *inflammable* and *flammable* above. Although *inflammable* and *flammable* are morphologically related, neither is derived from the other. *Flammable* is derived from the Latin noun *flamma* (flame); *inflammable* is derived from the corresponding Latin verb *inflammare* (to set on fire).

[72] This prefix is converted to *em-* when the base starts with *b* or *p*; for example, *emblazon* and *empower*.

Prefix to be matched and removed	Required right context (base start)	Examples
ab	None	*abnormal* → *normal*†, *abuse* → *use*†
ac	*c*	*acclimate* → *climate*†, *accompany* → *company*, *accredit* → *credit*†
ad	None	*adjoin* → *join*†, *adjudge* → *judge*†, *administer* → *minister*†, *admixture* → *mixture*, *adopt* → *opt*†, *adventure* → *venture*†
af	*f*	*affirm* → *firm*†, *affix* → *fix*†, *affront* → *front*†
al	*l*	*allay* → *lay*†, *allocate* → *locate*†, *allot* → *lot*†
ambi	None	*ambidextrous* → *dextrous*, *ambivalence* → *valence*
amphi	None	*amphitheater* → *theater*†

Table A.12: General rules for the removal of prefixes. The rules are listed in alphabetic order of the removed prefixes, with the exception that prefixes that are themselves prefixes of other prefixes are listed after the more specialized prefixes, e.g., *anti-* is listed before *an-*. In this way, related prefixes are grouped together and the order of the rules in the table corresponds to the order the rules are applied (e.g., an attempt to match and remove *intra-* will be made before an attempt for *in-*). All listed prefixes can be optionally followed by a dash separating the prefix from the rest of the word. If such a dash is found, it will be matched and removed when the prefix is removed.

In the examples marked with †, the word on the left side is not derived from the word on the right side through the addition of the prefix in a direct productive manner. Rather, the application of the affixation rule took place in a foreign language (usually Latin, French, or Greek), and both words were adopted into English through similar conversion procedures. For example, *normal* has been adopted into English from the Latin *normalis*, which itself is derived from *norma* (meaning "rule"). While *abnormal* looks like it could have been analyzed as *ab-* + *normal*, it has actually been adopted into English from the Latin *abnormis* (meaning "abnormal"), which is in turn derived from *ab-* and *norma*. Although the precise etymology of *abnormal* does not involve a direct derivation from *normal*, the use of the same Latin root for both words, the similarity in the transformations of the two words from their Latin original forms, and the persistence of the semantic relationship introduced in Latin by the prefix into the English versions of the words all justify considering the two words as morphologically related.

Consequently, I have included rules for relationships of this type. However, I do not include in the table rules for prefixes that are not productive in English, or for which the base form that would be extracted is not an English word or has lost in English the meaning it originally had in the foreign language (e.g., *apo-*, which appears in several English words such as *apologize*, *apostate*, and *apostrophe*, but is not productive in English and the words that contain it cannot be separated to *apo-* plus an English base word). Several prefixes from those considered were found to belong to this latter category; these prefixes are listed in Table A.13 on page 578.

The table is continued on the next page.

Prefix to be matched and removed	Required right context (base start)	Examples
ante	None	*antechamber* → *chamber*, *antechapel* → *chapel*, *antenatal* → *natal*, *anteroom* → *room*
anti	None	*anticlimactic* → *climactic*, *anti-establishment* → *establishment*, *anti-immigrant* → *immigrant*, *antithesis* → *thesis*†
an	None	*anhydrous* → *hydrous*†, *anencephalic* → *encephalic*, *anerotic* → *erotic*
ap	*p*	*append* → *pend*†, *appoint* → *point*†, *apportion* → *portion*, *appraise* → *praise*†, *approve* → *prove*†
arch	None	*archdiocese* → *diocese*, *archenemy* → *enemy*, *archrival* → *rival*
ar	*r*	*arrange* → *range*†, *arrest* → *rest*†, *arrhythmic* → *rhythmic*
at	*t*	*attest* → *test*†, *attribute* → *tribute*†, *attune* → *tune*
auto	None	*autobiography* → *biography*, *automobile* → *mobile*, *autopilot* → *pilot*[73]
a	None	*aback* → *back*, *aboard* → *board*, *adrift* → *drift*, *aflame* → *flame*, *along* → *long*†, *amoral* → *moral*, *anew* → *new*, *ashore* → *shore*, *asleep* → *sleep*, *astray* → *stray*†, *atypical* → *typical*†[74]
be	None	*become* → *come*†, *bedevil* → *devil*, *befriend* → *friend*, *beget* → *get*†, *behead* → *head*, *bemoan* → *moan*†, *besiege* → *siege*
bin	*a* or *o*	*binaural* → *aural*, *binocular* → *ocular*
bi	None	*bi-annual* → *annual*, *bicycle* → *cycle*, *bilateral* → *lateral*†, *biweekly* → *weekly*
circum	None	*circumlocution* → *locution*†, *circumnavigate* → *navigate*, *circumscribe* → *scribe*†
col	*l*	*collapse* → *lapse*†, *collateral* → *lateral*†, *collocate* → *locate*†
com	*b*, *f*, *m*, or *p*	*commingle* → *mingle*, *commotion* → *motion*†, *compatriot* → *patriot*†, *compose* → *pose*†, *compress* → *press*†, *compromise* → *promise*†
contra	None	*contraindicate* → *indicate*, *contra-revolutionary* → *revolutionary*

Table A.12: General rules for the removal of prefixes. Continued from the previous page and on the next page.

[73] I have included *auto-* in the table of productive prefixes since it can be used in its original sense, meaning *self*, as in the examples given. Note, however, that *auto-* is also freely used in compounds as an abbreviated form of *automobile*, e.g., *auto-accident*, *auto-enthusiast*, *auto-industry*. When *auto-* is used in the latter sense, the compound is usually hyphenated.

[74] The prefix *a-* has several functions. It can be a negative prefix (e.g., *amoral*, *atypical*); it can be used to produce *a-* adjectives and adverbs from verbs and nouns (e.g., *adrift*, *ashore*, *asleep*); and it can be used to produce poetic or intensified forms of adverbs from adverbs or adjectives (e.g., *aback*, *afar*, *anew*).

Prefix to be matched and removed	Required right context (base start)	Examples
con	*c*, *d*, *f*, *g*, *j*, *n*, *q*, *s*, *t*, or *v*	*concede* → *cede*†, *condescend* → *descend*†, *conform* → *form*†, *congenital* → *genital*†, *conjoin* → *join*†, *connotation* → *notation*, *conquest* → *quest*†, *consequence* → *sequence*†, *consign* → *sign*†, *contemporary* → *temporary*, *converge* → *verge*†
cor	*r*	*correlate* → *relate*, *correspond* → *respond*†
counter	None	*counterattack* → *attack*, *counterpressure* → *pressure*, *counterproductive* → *productive*
co	None	*coauthor* → *author*, *co-chair* → *chair*, *coexist* → *exist*, *cooperate* → *operate*†, *coworker* → *worker*
cross	None	*cross-cultural* → *cultural*, *crossbreed* → *breed*, *crossfire* → *fire*, *crossword* → *word*
demi	None	*demigod* → *god*, *demilance* → *lance*, *demimunicipal* → *municipal*
de	None	*deactivate* → *activate*, *decipher* → *cipher*†, *deform* → *form*†, *degrade* → *grade*†, *deplane* → *plane*, *devoid* → *void*†
dia	None	*dialysis* → *lysis*†, *diameter* → *meter*†
dis	None	*disallow* → *allow*, *disband* → *band*, *disgrace* → *grace*†, *dishonest* → *honest*†, *disinfect* → *infect*, *disobey* → *obey*, *displace* → *place*†, *distrust* → *trust*, *disuse* → *use*
di	None	*dibasic* → *basic*, *dilemma* → *lemma*†, *dioxide* → *oxide*, *disodium* → *sodium*, *dispirit* → *spirit*, *diverge* → *verge*†[75]
dys	None	*dysfunctional* → *functional*, *dyspeptic* → *peptic*
em	*b* or *p*	*embalm* → *balm*†, *embattle* → *battle*†, *embody* → *body*, *empanel* → *panel*, *empower* → *power*
en	None	*enact* → *act*, *encircle* → *circle*, *enforce* → *force*†, *enlarge* → *large*†, *enslave* → *slave*, *envision* → *vision*
epi	None	*epicenter* → *center*†, *epineural* → *neural*
ep	*a*, *e*, *i*, *o*, or *u*	*epaxial* → *axial*

Table A.12: General rules for the removal of prefixes. Continued from the previous page and on the next page.

[75]There are several prefixes that express number. Among them, the prefixes *di-* (for *two*) and *tri-* (for *three*) have been included in the table, as they are frequent and productive, especially in the sciences (with chemistry being a prime example). Several prefixes of Latin or Greek origin that express higher numbers have been omitted because they are not so common as *di-* and *tri-*. The list of omitted prefixes from this category includes *quadr(i)-* (meaning *four*); *pent(a)-*, *quinqu(e)-*, and *quint-* (meaning *five*); *hex(a)-* and *sex-* (meaning *six*); *sept-*, *septi-*, and *septem-* (meaning *seven*); *oct-*, *octa-*, and *octo-* (meaning *eight*); *dec(a)-* (meaning *ten*); and *dodec(a)-* (meaning *twelve*). I have also omitted many productive native English prefixes that express number, e.g., *one-*, *two-*, *three-*; *single-*, *double-*, *triple-*; *once-*, *twice-*, *thrice-*; etc. Since all the prefixes in the latter category are also valid English words, all words including these prefixes will be analyzed as compounds by the derivational morphology component.

Prefix to be matched and removed	Required right context (base start)	Examples
equi	None	*equidistant* → *distant*, *equivalence* → *valence*†, *equivocal* → *vocal*†
ere	None	*erelong* → *long*, *erenow* → *now*, *erewhile* → *while*[76]
es	*c*	*escape* → *cape*†
eu	None	*eupeptic* → *peptic*, *euplastic* → *plastic*
exo	None	*exoatmospheric* → *atmospheric*, *exoskeleton* → *skeleton*
extra	None	*extracorporeal* → *corporeal*, *extracurricular* → *curricular*, *extramarital* → *marital*, *extrasensory* → *sensory*, *extraordinary* → *ordinary*†, *extraterrestrial* → *terrestrial*
ex	None	*ex-governor* → *governor*, *exchange* → *change*†, *exclaim* → *claim*†, *explain* → *plain*†, *express* → *press*†
e	None	*elapse* → *lapse*†, *emerge* → *merge*†, *enumerate* → *numerate*†, *espouse* → *spouse*†, *evocative* → *vocative*†
fore	None	*forearm* → *arm*, *foreclose* → *close*†, *foreground* → *ground*, *forehead* → *head*†, *foretell* → *tell*, *foreword* → *word*†
for	None	*forbid* → *bid*†, *forever* → *ever*, *forget* → *get*†, *forgive* → *give*†, *for-profit* → *profit*, *forswear* → *swear*†
half	None	*half-brother* → *brother*, *half-full* → *full*, *halftime* → *time*, *half-hearted* → *hearted*[77]
hemi	None	*hemiprism* → *prism*, *hemisphere* → *sphere*†
hetero	None	*heterosexual* → *sexual*, *heterozygote* → *zygote*
homo	None	*homograph* → *graph*†, *homophobia* → *phobia*, *homosexual* → *sexual*
hyper	None	*hyperactive* → *active*, *hyperinflation* → *inflation*, *hypersensitive* → *sensitive*, *hyperventilate* → *ventilate*
ill	None	*ill-adapted* → *adapted*, *ill-planned* → *planned*, *ill-trained* → *trained*
il	*l*	*illegal* → *legal*†, *illegible* → *legible*, *illicit* → *licit*†, *illiterate* → *literate*†, *illogical* → *logical*
im	*b*, *m*, or *p*	*imbalance* → *balance*, *immature* → *mature*†, *immobilize* → *mobilize*†, *impassable* → *passable*, *impatient* → *patient*†, *imprison* → *prison*†

Table A.12: General rules for the removal of prefixes. Continued from the previous page and on the next page.

[76] A prefix generating words that are now considered archaic.

[77] Although *hearted* usually appears in compound words (e.g., *half-hearted*, *faint-hearted*, *stout-hearted*), it can more rarely appear on its own, meaning either “taken to heart” or “having the shape of a heart”.

Prefix to be matched and removed	Required right context (base start)	Examples
infra	None	*infrared* → *red*, *infrastructure* → *structure*
inter	None	*interact* → *act*, *interconnect* → *connect*, *interdependent* → *dependent*, *interlace* → *lace*, *international* → *national*, *interweave* → *weave*
intra	None	*intrafactional* → *factional*, *intramural* → *mural*, *intraparty* → *party*, *intravenous* → *venous*
in	None	*inaccurate* → *accurate*, *inbreed* → *breed*, *incapacitate* → *capacitate*, *inexact* → *exact*, *insane* → *sane*†, *inside* → *side*, *intake* → *take*, *into* → *to*
ir	*r*	*irradiate* → *radiate*†, *irrational* → *rational*†, *irrelevant* → *relevant*, *irreplaceable* → *replaceable*
kilo	None	*kilobyte* → *byte*, *kilometer* → *meter*†, *kilowatt* → *watt*
macro	None	*macrochange* → *change*, *macroeconomic* → *economic*
mal	None	*maladroit* → *adroit*, *malcontent* → *content*, *malfunction* → *function*, *malpractice* → *practice*
micro	None	*microbiology* → *biology*, *microcircuit* → *circuit*, *microfilm* → *film*, *micromanage* → *manage*, *microwave* → *wave*
mid	None	*mid-level* → *level*, *midnight* → *night*, *midsea* → *sea*, *mid-session* → *session*, *midwife* → *wife*†
mini	None	*minidress* → *dress*, *miniseries* → *series*, *minivan* → *van*
mis	None	*misbehave* → *behave*, *miscarry* → *carry*, *misconduct* → *conduct*, *misfire* → *fire*, *misinform* → *inform*, *mismatch* → *match*, *misspell* → *spell*, *mistrust* → *trust*
mono	None	*monochromatic* → *chromatic*†, *monograph* → *graph*†, *monoplane* → *plane*, *monosyllable* → *syllable*†
multi	None	*multibillion* → *billion*, *multiethnic* → *ethnic*, *multilayer* → *layer*, *multipurpose* → *purpose*, *multiuser* → *user*
neo	None	*neoclassic* → *classic*, *neoconservative* → *conservative*, *neonatal* → *natal*, *neoprohibitionist* → *prohibitionist*
non	None	*nonalcoholic* → *alcoholic*, *nonfatal* → *fatal*, *nonperson* → *person*, *nontoxic* → *toxic*
ob	None	*oblong* → *long*†, *obnoxious* → *noxious*†
off	None	*off-chance* → *chance*, *off-color* → *color*, *off-season* → *season*, *offset* → *set*, *offshore* → *shore*, *off-white* → *white*
omni	None	*omnicompetent* → *competent*, *omnipotent* → *potent*†, *omnipresent* → *present*

Table A.12: General rules for the removal of prefixes. Continued from the previous page and on the next page.

Prefix to be matched and removed	Required right context (base start)	Examples
op	*p*	*oppose* → *pose*†, *oppress* → *press*†
out	None	*outburst* → *burst*, *outcry* → *cry*, *outdo* → *do*, *outguess* → *guess*, *outlaw* → *law*†, *outpatient* → *patient*
over	None	*overactive* → *active*, *overbook* → *book*, *overdose* → *dose*, *overnight* → *night*, *overwrite* → *write*
paleo	None	*paleoantropology* → *anthropology*, *paleolithic* → *lithic*
pan	None	*pan-Arab* → *Arab*, *panchromatic* → *chromatic*, *pantheism* → *theism*
peri	None	*perimeter* → *meter*, *perioptic* → *optic*, *periscope* → *scope*
per	None	*perforce* → *force*, *perfume* → *fume*†, *perjury* → *jury*†, *permutation* → *mutation*†, *peroxide* → *oxide*, *persevere* → *severe*†
poly	None	*polyacetylene* → *acetylene*, *polygraph* → *graph*, *polysyllabic* → *syllabic*, *polyurethane* → *urethane*
post	None	*postdate* → *date*, *postmodern* → *modern*, *postscript* → *script*
pre	None	*precancerous* → *cancerous*, *precaution* → *caution*†, *predominant* → *dominant*†, *preexist* → *exist*, *prehistory* → *history*, *prepay* → *pay*, *preschool* → *school*
proto	None	*proto-fascist* → *fascist*, *protomartyr* → *martyr*, *prototype* → *type*†
pro	None	*proactive* → *active*, *proclaim* → *claim*†, *pro-government* → *government*, *prolong* → *long*†, *pronoun* → *noun*†, *proportion* → *portion*†
pseudo	None	*pseudo-Gothic* → *Gothic*, *pseudoscience* → *science*
re	None	*readmit* → *admit*, *recharge* → *charge*, *reentry* → *entry*, *relax* → *lax*†, *renew* → *new*†, *return* → *turn*†
self	None	*self-appointed* → *appointed*, *self-hatred* → *hatred*, *self-sufficient* → *sufficient*
semi	None	*semiautonomous* → *autonomous*, *semiconductor* → *conductor*, *semifinal* → *final*, *semiofficial* → *official*
sub	None	*subculture* → *culture*, *subgroup* → *group*, *sublet* → *let*, *suboptimal* → *optimal*, *subsonic* → *sonic*
suf	*f*	*suffix* → *fix*†, *suffumigate* → *fumigate*†
super	None	*superabundant* → *abundant*, *superhero* → *hero*, *superimpose* → *impose*

Table A.12: General rules for the removal of prefixes. Continued from the previous page and on the next page.

Prefix to be matched and removed	Required right context (base start)	Examples
sup	*p*	*support* → *port*†, *suppose* → *pose*†, *suppress* → *press*†
sur	None	*surcharge* → *charge*†, *surface* → *face*†, *surplus* → *plus*†, *surreal* → *real*†, *surtax* → *tax*†
tele	None	*telecommunicate* → *communicate*, *teleshopping* → *shopping*, *television* → *vision*
through	None	*throughout* → *out*, *throughput* → *put*
trans	None	*transact* → *act*†, *transcontinental* → *continental*, *transform* → *form*†, *translucent* → *lucent*†, *transplant* → *plant*†
tri	None	*triangle* → *angle*†, *tricalcium* → *calcium*, *tricolor* → *color*†, *tricycle* → *cycle*, *triglyceride* → *glyceride*, *trioxide* → *oxide*, *tristate* → *state*[78]
ultra	None	*ultracompetitive* → *competitive*, *ultrasensitive* → *sensitive*, *ultraviolet* → *violet*
under	None	*underachieve* → *archieve*, *underdog* → *dog*, *underlie* → *lie*†, *underplay* → *play*
uni	None	*unidirectional* → *directional*, *uniform* → *form*†, *unilateral* → *lateral*†, *unisex* → *sex*
un	None	*unable* → *able*, *undo* → *do*†, *unhappy* → *happy*, *unnatural* → *natural*, *unsteady* → *steady*
up	None	*upbeat* → *beat*, *update* → *date*, *upgrade* → *grade*, *uplift* → *lift*, *uptown* → *town*
vice	None	*vice-chairman* → *chairman*, *vice-president* → *president*

Table A.12: General rules for the removal of prefixes. Continued from the previous page.

sequently, the morphology rules are not constrained by it. There are several reasons for adopting this policy:

- The morphological component, which is already quite complex, would be even more complicated if such constraints were employed.

[78] See footnote 75 on page 573.

Prefix	Required right context (base start)	Examples
apo	None	*apologize*, *apostate*, *apostrophe*
dicho	None	*dichotomy*
ec	*c*, *l*, or *s*	*eccentric*, *eclipse*, *ecstasy*
eph	None	*ephemeral*
oc	*c*	*occlude*, *occult*, *occupy*
of	*f*	*offend*, *offer*
o	*m*	*omit*
suc	*c*	*succeed*, *succubus*
sug	*g*	*suggest*
sum	*m*	*summon*
sus	*c*, *p*, or *t*	*susceptible*, *suspect*, *sustain*

Table A.13: Non-productive prefixes, which were excluded from the reverse morphology rules listed in Table A.12. The prefixes listed here are (or were) productive in the foreign language from which the word (together with the prefix) was adopted into English. But the base word has not been independently adopted and does not occur by itself in English, or it has lost its original meaning. See the related discussion in the caption of Table A.12 on page 571.

- Many affixes usually apply to more than one grammatical categories. For example, the suffix *-age* applies to nouns or verbs to produce nouns; *footage* is an example derived from a nominal base, while *marriage* is an example derived from a verbal base. If the word's grammatical category were used during the reverse application of the affixation rules, then in many cases we would have to keep sets of potential parts of speech for the resulting word, and either try to disambiguate the part of speech from the word list or corpus, or pursue several parallel alternatives.

- Even though many affixation rules are restricted to specific grammatical classes, there are numerous exceptions. For example, the suffix *-ity* normally applies to adjectives, as discussed above; however, we also have the denominal *authority* (from the noun *author*), and *hospitality* (from the same Latin base *hospes* → *hospitalis* from which

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
atic	None	None	*erratic* → *err*, *problematic* → *problem*
	m	*a*	*dogmatic* → *dogma*, *traumatic* → *trauma*
ic	None	None	*allergic* → *allergy*†, *athletic* → *athlete*†, *magnetic* → *magnet*
eed	None	*ood*	*bleed* → *blood*, *feed* → *food*
ed	None	None	*played* → *play*, *wooded* → *wood*
oid	None	None	*humanoid* → *human*
fold	None	None*	*fivefold* → *five*, *severalfold* → *several*
hood	None	None	*likelihood* → *likely*†, *motherhood* → *mother*, *priesthood* → *priest*
ward	None	None	*heavenward* → *heaven*, *upward* → *up*
ance	None	None	*admittance* → *admit*†, *compliance* → *comply*†, *inheritance* → *inherit*
		ant	*arrogance* → *arrogant*, *clairvoyance* → *clairvoyant*
ence	None	None	*convergence* → *converge*†, *existence* → *exist*, *occurrence* → *occur*†
		ent	*affluence* → *affluent*, *innocence* → *innocent*, *silence* → *silent*

Table A.14: General rules for the removal of suffixes. Despite their apparent random order, the rules are actually listed in alphabetic order of the removed suffixes when the latter are read from right to left, i.e., in the way they are read by the matching algorithm. The exception to this order is that suffixes that are themselves suffixes of other suffixes are listed after the more specialized suffixes, e.g., *-ous* is listed after *-ious*. This has the advantage that related suffixes which may simultaneously match a given word (e.g., *-ical* and *-al*) are listed close to each other, with more specialized suffixes first.

When a rule is matched, the remaining part of the word after the suffix has been removed has the contents of the "Replacement" column (if any) added to its right. For example, *traumatic* matches the *-atic* suffix and satisfies the additional constraint that the apparent base ends in *-m*. Removing the suffix *-atic* leaves *traum*, to which the base correction *-a* is added, so that *trauma* is produced.

In the examples marked with †, the word on the right side is derived after the application of the suffixation rule and one or more of the base correction rules of Table A.16. For example, the removal of the suffix *-hood* from *likelihood* initially gives *likeli*, which is corrected to *likely* by the rule that specifies that a final *i* in the base was initially a *y*.

Some rules mark the replacement part with *. In these cases, the forward suffixation rule is known never to alter the base, so the base correction rules of Table A.16 are not tried.

The table is continued on the next page.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
ee	None	None	*absentee* → *absent*, *retiree* → *retire*†
age	None	None	*baggage* → *bag*†, *pilferage* → *pilfer*, *wreckage* → *wreck*
like	None	None*	*dreamlike* → *dream*, *ghostlike* → *ghost*
able	None	None	*admirable* → *admire*†, *fashionable* → *fashion*, *noticeable* → *notice*, *taxable* → *tax*
-style	None	None*	*colonial-style* → *colonial*, *gourmet-style* → *gourmet*
some	None	None	*lonesome* → *lone*, *worrisome* → *worry*†
ine	None	None	*crystalline* → *crystal*†, *elephantine* → *elephant*, *heroine* → *hero*
ware	None	None*	*houseware* → *house*, *software* → *soft*
more	None	None	*anymore* → *any*, *furthermore* → *further*
ure	None	None	*exposure* → *expose*†, *pressure* → *press*
ese	None	None	*Japanese* → *Japan*, *legalese* → *legal*
wise	None	None*	*clockwise* → *clock*, *lengthwise* → *length*, *pricewise* → *price*
ise	None	None	*expertise* → *expert*[79]
rate	None	*er*	*calibrate* → *caliber*, *integrate* → *integer*
ate	None	None	*activate* → *active*†, *affectionate* → *affection*, *electorate* → *elector*, *originate* → *origin*
ite	None	None	*composite* → *compose*†, *Reaganite* → *Reagan*, *socialite* → *social*
ette	None	None	*diskette* → *disk*, *kitchenette* → *kitchen*, *statuette* → *statue*†, *sufragette* → *suffrage*†
esque	None	None	*Kafkaesque* → *Kafka*, *picturesque* → *picture*†
ative	None	None	*exploitative* → *exploit*, *preservative* → *preserve*†
itive	None	*e*	*definitive* → *define*, *sensitive* → *sense*
ive	None	None	*active* → *act*, *relative* → *relate*†, *secretive* → *secret*, *vituperative* → *vituperate*†
ve	*ie* or *l*	*f*	*believe* → *belief*, *grieve* → *grief*, *halve* → *half*, *relieve* → *relief*
-size	None	None*	*book-size* → *book*, *life-size* → *life*, *same-size* → *same*

Table A.14: General rules for the removal of suffixes. Continued from the previous page and on the next page.

[79] Also often employed in British English in the place of the suffix *-ize* to form verbs from adjectives and nouns; e.g., *legal* → *legalise*, *scrutiny* → *scrutinise*.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
ize	None	None	*authorize* → *author*, *categorize* → *category*†, *liberalize* → *liberal*
e	*th*	None	*bathe* → *bath*, *breathe* → *breath*, *loathe* → *loath*
ling	None	None	*duckling* → *duck*, *foundling* → *found*, *weakling* → *weak*
ing	None	None	*calling* → *call*, *fishing* → *fish*, *planning* → *plan*†
ish	None	None	*bluish* → *blue*†, *devilish* → *devil*, *fortyish* → *forty*, *tallish* → *tall*
eadth	None	*oad*	*breadth* → *broad*
idth	None	*ide*	*width* → *wide*
ength	None	*ong*	*length* → *long*, *strength* → *strong*
epth	None	*eep*	*depth* → *deep*
th	None	None	*health* → *heal*, *growth* → *grow*, *sixth* → *six*
ical	None	None	*astronomical* → *astronomy*†, *farcical* → *farce*†, *puritanical* → *puritan*
tial	*n*	*ce*	*consequential* → *consequence*, *essential* → *essence*, *substantial* → *substance*
ial	None	None	*ceremonial* → *ceremony*†, *facial* → *face*†, *ministerial* → *minister*
al	None	None	*appraisal* → *appraise*†, *logical* → *logic*, *personal* → *person*
ful	None	None	*boastful* → *boast*, *graceful* → *grace*, *spoonful* → *spoon*
dom	None	None	*freedom* → *free*, *kingdom* → *king*, *wisdom* → *wise*†
ism	None	None	*alcoholism* → *alcohol*, *expressionism* → *expression*, *terrorism* → *terror*
orium	None	None	*auditorium* → *audit*, *crematorium* → *cremate*†
ician	None	None	*beautician* → *beauty*†, *dietician* → *diet*
arian	None	None	*antiquarian* → *antique*†, *libertarian* → *liberty*†, *sectarian* → *sect*
ian	None	None	*civilian* → *civil*, *clinician* → *clinic*, *mammalian* → *mammal*

Table A.14: General rules for the removal of suffixes. Continued from the previous page and on the next page.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
man	None	None*	*boatman*→*boat*, *freshman*→*fresh*, *salesman*→*sales*[80]
isan	None	None	*artisan* → *art*, *partisan* → *party*†
an	None	None	*Chicagoan* → *Chicago*, *epicurean* → *Epicure*
		a	*Columbian* → *Columbia*
men	None	None*	*chairmen* → *chair*, *linemen* → *line*, *seamen* → *sea*
en	None	None	*awaken* → *awake*†, *beaten* → *beat*, *oxen* → *ox*, *widen* → *wide*†
mation	None	*im*	*acclamation* →*acclaim*, *exclamation* →*exclaim*
aration	None	*air*	*reparation* →*repair*
eration	None	*air*	*desperation* →*despair*
ration	*t*	*er*	*administration* →*administer*, *filtration* →*filter*, *registration* →*register*
ation	None	None	*accreditation* →*accredit*, *commendation* →*commend*, *improvisation* → *improvise*†, *taxation* → *tax*
ulsion	None	*el*	*compulsion* → *compel*, *repulsion* → *repel*
ssion	*i*	*t*	*omission* → *omit*, *permission* → *permit*, *transmission* → *transmit*
sion	*r*	*t*	*aversion* → *avert*, *inversion* → *invert*
	None	*d*	*expansion* → *expand*, *suspension* → *suspend*
		de	*conclusion* → *conclude*, *evasion* → *evade*, *implosion* → *implode*
ition	*ol*	*ish*	*abolition* → *abolish*, *demolition* → *demolish*
	None	None	*competition* → *compete*†, *erudition* → *erudite*†, *partition* → *part*
mption	*e*	*eem*	*redemption* → *redeem*
ption	*i*	*be*	*description* → *describe*, *inscription* → *inscribe*, *prescription* → *prescribe*
	e	*ive*	*conception* → *conceive*, *reception* → *receive*
	None	*b*	*absorption* → *absorb*
		None	*resumption* → *resume*†

Table A.14: General rules for the removal of suffixes. Continued from the previous page and on the next page.

---

[80]Words ending in *-man* without an explicit dash are somewhat between compounds and derived words. While *man* appears freely as a base word, changes in pronunciation indicate that it may sometimes function as a suffix (e.g., /mən/ in *policeman* versus /mæn/ in *ape-man*) [Quirk *et al.* 1985, p. 1520]; compare also the intonation patterns in these two examples.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
tion	*duc*	*e*	*introduction* → *introduce*, *production* → *produce*, *reduction* → *reduce*
ion	None	None	*communication* → *communicate*†, *infection* → *infect*, *profession* → *profess*
person	None	None*	*congressperson* → *congress*, *salesperson* → *sales*[81]
n	*w*	None*	*drawn* → *draw*, *known* → *know*, *sewn* → *sew*
ship	None	None*	*citizenship* → *citizen*, *friendship* → *friend*, *membership* → *member*
eer	None	None	*mountaineer* → *mountain*, *racketeer* → *racket*
ster	None	None	*gangster* → *gang*, *hipster* → *hip*, *prankster* → *prank*
er	None	None	*biker* → *bike*†, *dinner* → *dine*†, *lower* → *low*, *owner* → *own*, *villager* → *village*†
or	None	None	*advisor* → *advise*†, *collector* → *collect*, *possessor* → *possess*
wards	None	None	*backwards* → *back*, *upwards* → *up*
es	None	None	*affixes*→*affix*, *entities*→*entity*†, *perplexes*→*perplex*
itis	None	None	*sinusitis* → *sinus*, *tonsillitis* → *tonsil*†
less	None	None	*ageless* → *age*, *groundless* → *ground*, *worthless* → *worth*
ness	None	None	*blackness* → *black*, *fairness* → *fair*, *steadiness* → *steady*†
ess	None	None	*goddess* → *god*†, *largess* → *large*†, *priestess* → *priest*
eous	None	None	*righteous* → *right*
ious	None	None	*ceremonious* → *ceremony*†, *laborious* → *labor*, *spacious* → *space*†
ous	None	None	*calamitous* → *calamity*†, *famous* → *fame*†, *scandalous* → *scandal*
	i	*on*	*ambitious* → *ambition*, *religious* → *religion*, *suspicious* → *suspicion*
	ci	*ty*	*ferocious* → *ferocity*, *pugnacious* → *pugnacity*, *voracious* → *voracity*
s	None	None	*beasts* → *beast*, *detracts* → *detract*, *needles* → *needle*
let	None	None	*booklet* → *book*, *leaflet* → *leaf*, *starlet* → *star*

Table A.14: General rules for the removal of suffixes. Continued from the previous page and on the next page.

[81] Similar comments to those made for -*man* (footnote 80 on the previous page) apply to the use of -*person* as a suffix.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
et	None	None	*cabinet* → *cabin*, *facet* → *face*†, *packet* → *pack*
ant	None	None	*assistant* → *assist*, *malignant* → *malign*, *occupant* → *occupy*†
ment	None	None	*commitment* → *commit*, *employment* → *employ*, *treatment* → *treat*
ent	None	None	*dependent* → *depend*, *different* → *differ*, *president* → *preside*†
est	None	None	*coldest* → *cold*, *largest* → *large*†, *tightest* → *tight*
ist	None	None	*cartoonist* → *cartoon*, *fundamentalist* → *fundamental*, *novelist* → *novel*, *racist* → *race*†
most	None	None	*innermost* → *inner*, *southernmost* → *southern*, *uttermost* → *utter*
st	None	None	*amongst* → *among*, *canst* → *can*, *midst* → *mid*, *whilst* → *while*†
out	None	None*	*blackout* → *black*, *fallout* → *fall*, *layout* → *lay*
ocracy	None	None	*meritocracy* → *merit*, *pedantocracy* → *pedant*
acy	None	*e*	*conspiracy* → *conspire*, *supremacy* → *supreme*
cy	None	*te*	*adequacy* → *adequate*, *candidacy* → *candidate*, *literacy* → *literate*
		t	*efficiency* → *efficient*, *infancy* → *infant*, *pregnancy* → *pregnant*, *secrecy* → *secret*
		None	*bankruptcy* → *bankrupt*, *normalcy* → *normal*
ify	None	None	*glorify* → *glory*†, *purify* → *pure*†, *simplify* → *simple*†
worthy	None	None*	*newsworthy* → *news*, *seaworthy* → *sea*, *trustworthy* → *trust*
ology	None	None	*iconology* → *icon*, *immunology* → *immune*†, *methodology* → *method*, *toxicology* → *toxic*
urgy	None	None	*metallurgy* → *metal*†
ky	*c*	None	*panicky* → *panic*
ally	*ic*	None	*automatically* → *automatic*, *chronically* → *chronic*, *pragmatically* → *pragmatic*
ly	None	None	*anxiously* → *anxious*, *deadly* → *dead*, *motherly* → *mother*, *normally* → *normal*, *precisely* → *precise*, *weekly* → *week*

Table A.14: General rules for the removal of suffixes. Continued from the previous page and on the next page.

Suffix to be matched and removed	Required base ending	Replacement (base correction)	Examples
ary	None	None	*complementary* → *complement*, *inflationary* → *inflation*, *planetary* → *planet*
ery	None	None	*brewery* → *brew*, *greenery* → *green*, *jewellery* → *jewel*†
atory	None	None	*accusatory* → *accuse*†, *conservatory* → *conserve*†, *inflammatory* → *inflame*†, *purgatory* → *purge*†
ory	None	None	*contradictory* → *contradict*, *inhibitory* → *inhibit*, *promissory* → *promise*†, *regulatory* → *regulate*†
ry	None	None	*artistry* → *artist*, *bravery* → *brave*, *mimicry* → *mimic*, *nursery* → *nurse*, *slavery* → *slave*
ility	None	*le*	*ability* → *able*, *feasibility* → *feasible*, *nobility* → *noble*
ity	None	None	*activity* → *active*†, *cordiality* → *cordial*, *mobility* → *mobile*†
ty	None	None	*cruelty* → *cruel*, *mayoralty* → *mayoral*, *safety* → *safe*
y	None	None	*cheery* → *cheer*, *fluffy* → *fluff*, *honesty* → *honest*, *icy* → *ice*†, *mastery* → *master*, *runny* → *run*†, *sandy* → *sand*

Table A.14: General rules for the removal of suffixes. Continued from the previous page.

hospital and *hospitable* are produced; but compare **hospitability*).

- Applying an affixation rule to a word with a grammatical category that could not have been produced from this specific rule is not a major problem, as most of the "words" produced in this way will not belong to the vocabulary of English. Consequently, the danger of accidentally matching an adjective in this way is small. Recall that the aim of the derivational component is not to produce a full morphological analysis of any given word; rather, it is to determine with high accuracy if a word is derived from another word.

Irregular adjectives		
Base form	Comparative form	Superlative form
bad	*worse*	*worst*
good	*better*	*best*
far	*farther*	*farthest*
	further	*furthest*

Regular morphology rules		
Base form	Inflected forms	Examples
$\sim Cy$	$\sim Cie(r/st)$	*hefty* → *heftier*, *lazy* → *lazier*, *merry* → *merrier*[82]
$\sim VC$	$\sim VCe(r/st)$	*cheap* → *cheaper*, *proud* → *prouder*, *stupid* → *stupider*
	$\sim VCCe(r/st)$	*big* → *bigger*, *mad* → *madder*, *trim* → *trimmer*[83]
$\sim e$	$\sim e(r/st)$	*free* → *freer*, *noble* → *nobler*, *rare* → *rarer*
$\sim$	$\sim e(r/st)$	*bold* → *bolder*, *clean* → *cleaner*, *poor* → *poorer*

Table A.15: Forward morphology rules for forming the comparative and superlative forms of adjectives. While the examples use the suffix *-er* of the comparative form, exactly the same rules apply for the suffix *-est* of the superlative form. More specific rules (towards the top of the table) are applied before less specific rules (towards the bottom of the table). The symbols V and C in the general rules stand for any vowel and any consonant respectively. Repeated occurrences of these symbols in the same rule refer to the same letter. In particular, the sequence CC corresponds to a double consonant (e.g., *dd* or *gg*). Some base forms are ambiguous in their production of inflected forms. Consequently, up to two inflected forms per degree can be produced for each adjective.

The full algorithm for the derivational component (listed in Figure A.9 on pages 587–588) proceeds as follows: First, the list of adjectives that we are interested in is read and stored in memory. The single-word comparative and superlative forms for these adjectives are computed and stored along with the adjectives, by following the rules of Table A.15. These rules are essentially the counterparts of the reverse rules of Table A.8 on page 552 (Section A.5), but since they are applied in the forward direction, they are unambiguous

[82] Compare with adjectives ending in a vowel plus *-y*, e.g., *coy* → *coyer*, *gray* → *grayer*.

[83] Usually, only one of the two alternatives suggested by this rule is valid. If the vowel before the final consonant is part of a diphthong or not stressed, the first alternative applies, otherwise the second alternative applies. *Cruel* is an exception, as it allows both forms *crueler* (preferred in American English) and *crueller*.

Input: A list of adjectives, L, and a list of all words in the vocabulary of the language with their frequencies, V.

Output: For each $a \in L$ and each $w \in V$, a determination of the morphological relationship (if any) between a and w.

For each $w \in V$
  Separate w into word segments at each occurrence of a dash in w.
  Apply the rules of Table A.12 recursively, to remove any prefixes explicitly separated with a dash.
  Apply the rules of Table A.14 recursively, to remove any suffixes explicitly separated with a dash. Do not apply any of the corrective rules associated with suffixation (Table A.16).
  Let w' be the result of the above operations.

  If w' contains no dashes then
    For each $a \in L$
      If a and w' are identical, then
        If $w = w'$ then
          Report that a and w are identical
        else
          Report that w is a word derived from a
      else if w' is the same as one of the stored inflected forms of a (comparative or superlative) then
        If $w = w'$ then
          Report that w is an inflected form of a
        else
          Report that w is a word derived from a
      else
        Recursively remove suffixes from w', applying the rules of Table A.14 and the specific and generic rules for correcting the retrieved base (Tables A.14 and A.16).

        For each candidate word w'' so formed
          If $a = w''$ then
            Report that w is a word derived from a
          else if a is a suffix of w'' then
            Recursively remove prefixes from w'', according to the rules of Table A.12.

Figure A.9: The algorithm applied by the derivational component to determine whether each word in the vocabulary is derived from an adjective in a supplied list of adjectives or is a compound involving such an adjective. Whenever the algorithm reports the relationship between the current word and adjective, the flow of control is transferred to the next adjective in the supplied list. Continued on the next page.

If this process eventually leads to a being recovered, then
Report that w is a word derived from a

(This point is reached if no identifiable derivational procedure (i.e., sequence of rules) has been found.)

If a is a prefix or suffix of w', then
Report w as a non-hyphenated compound of a
else if a is a substring of w', then
Report w as a word containing a as a substring
else
Report that w and a are morphologically unrelated

else *(w' still contains at least one dash)*
For each $a \in L$
If a is identical to any of the segments to which w was separated then
Report w as an hyphenated compound containing a
else
Let w_1 be the prefix of w' up to (but not including) the first dash.

Repeat the procedure listed for the case where w' contains no dash but using w_1 in the place of w'. If w would be considered an inflected form of a or a word derived from a, report instead that w is a hyphenated compound of a word derived from a.

If no match has been found up to this point, let w_2 be the suffix of w' from (but not including) the last dash, and repeat the operations listed above for w_1 using w_2 in the place of w_1.

Figure A.9: The algorithm applied by the derivational component to determine whether each word in the vocabulary is derived from an adjective in a supplied list of adjectives or is a compound involving such an adjective. Continued from the previous page.

and simpler.[84] Then each word is read in turn from the word list that represents the total vocabulary of the language; either the MRC database or a word list extracted from the corpus can be used for this purpose.

[84]The application of these rules will produce many forms that are not valid English words, because multiple rules may apply for a given adjective, because the base adjective may be too long to form single-word comparative and superlative forms, or because the adjective may be non-gradable. For example, *appropriate* will produce **appropriater* and **appropriatest*, while *big* will produce two pairs of inflected forms, **biger*/**bigest* and *bigger*/*biggest*. However, this is not a problem as the erroneous forms will not be found in the word list and consequently will not be matched to any words.

Each word in the word list is separated into simpler words containing only letters. This is accomplished by breaking the word at every occurrence of a dash ('-') (see also the related discussion on spelling checking in Section A.5, where the same technique is employed to identify simple words participating in compounds). Initial and final word segments obtained with this method from a word that contains dashes are compared against the lists of prefixes and suffixes (Tables A.12 and A.14), to identify affixes that either require or frequently employ a dash. Such affixes are removed from the word, and if more than one segment remains, the original word is considered a compound, otherwise it is considered a simple word. This affects which counts of types and frequencies will be incremented when successful matches with adjectives are found, as explained below.

Each of the retrieved word segments (possibly just one, if we had a simple word to start with) is compared with each adjective to determine if they are identical; if so, we have found a compound where the adjective participates, or, if no dashes appear in the original word, the entry for the adjective (or one of its homographs) in the word list. If the word and an adjective are not the same, then the word and the comparative and superlative forms of the adjective are compared. If again no match is found, the word and the adjective are passed to the affixation algorithm. The affixation algorithm attempts first to remove suffixes from the word, until the adjective is itself a suffix of the word (i.e., the word, after being stripped from suffixes, contains the adjective and possibly several prefixes). Suffixes are removed according to the rules of Table A.14, using the depth-first search method described earlier to exhaustively check alternatives. Several suffixes cause morphological changes to the base when they are applied, and theses changes must be accounted for when we are going in the reverse direction. The following morphological rules are applied to correct the base after the removal of each suffix, but only if no correction (i.e., replacement for the suffix) has been explicitly specified in the reverse suffixation rule[85]:

- If the derived base ends in *i*, the *i* is converted to *y*. For example, *happiness* is first matched to the suffix -*ness* producing *happi*; this is then corrected to *happy* by this rule.
- If the removed suffix starts with a vowel (*a*, *e*, *i*, *o*, *u*, or *y*) and the original base ends in *e*, the final *e* is normally dropped during suffixation and should be restored. Therefore, an *e* is added to the derived base whenever the suffix starts with a vowel. This rule will handle correctly cases such as *aspirant*, where the removal of the suffix -*ant* leads initially to *aspir* and the latter is transformed to the correct *aspire*.
- If the removed suffix starts with a vowel and the derived base ends in a sequence of two identical consonants preceded by a vowel, then the final consonant of the original base may have been duplicated during suffixation, in which case the second of the two identical consonants should be removed from the base. For example, *repellant* is initially matched to suffix -*ant*, producing *repell*; this is corrected to *repel* by this rule. Note that sometimes the original base ends in the double consonant, for example *depress* → *depressant*. Because of these cases, both alternatives (removing the final consonant and keeping it) are pursued in parallel, using the depth-first search method.

[85]Suffixation rules can also specify that these base-correcting rules should be blocked, even if no explicit morphological correction is mentioned in the suffixation rule. This is the case, for example, for the suffixes *fold* and *wise*. Suffixation rules that explicitly block the base correction rules are marked with * in the 'Replacement" column of Table A.14.

Another alternative, also pursued in parallel with the previous two, is that the original base ends in *e*, which is lost during suffixation and that an *e* should be restored after the double consonant. For example, *finesse* + *-ing* → *finessing*, so the form *fines*(*s*) that would have been produced by the previous two alternatives after removing the suffix *-ing* should be corrected to *finesse*. This alternative is actually a special case of the previous general rule covering the removal of a final *e* during suffixation, and so is not explicitly listed in this rule.

- An apparent loss of a final *e* in the base word can also occur when the suffix starts with a consonant. In these cases, the suffixation is not a productive process in English, but took place (according to different rules) in a foreign language and the two words were adopted independently in English. Nevertheless, the two words *are* morphologically and semantically related, and should be identified as such by the affixation analysis component. Examples of such pairs of related words are *resumption* → *resum* → *resume* and *whilst* → *whil* → *while*. I combine this case with the immediately previous rule, by dropping the restrictions on the initial character of the removed suffix.

- If the removed suffix starts with a vowel (usually *i*) and the derived base ends with a consonant, then the original base may have ended in *y*, which was converted to *i* before the suffix and then removed. For example, *symmetry* + *-ic* → *symmetriic* → *symmetric*, so a final *y* is added to the base *symmetr* when it is recovered from *symmetric*. Similarly, *library* + *-ian* → *librariian* → *librarian*.

Start of removed suffix	End of derived base	Correction applied to base	Examples
Not restricted	$\sim i$	Replace i by y	*happiness* → *happi* → *happy*, *compliance* → *compli* → *comply*, *worrisome* → *worri* → *worry*
Not restricted	Not restricted	Add e	*famous* → *fam* → *fame*, *president* → *presid* → *preside*, *whilst* → *whil* → *while*, *widen* → *wid* → *wide*
Vowel	Two identical consonants	Drop the second of the two identical consonants	*baggage* → *bagg* → *bag*, *crystalline* → *crystall* → *crystal*, *planning* → *plann* → *plan*, *runny* → *runn* → *run*
Vowel	Consonant	Add y	*calamitous* → *calamit* → *calamity*, *ceremonial* → *ceremon* → *ceremony*, *glorify* → *glor* → *glory*, *occupant* → *occup* → *occupy*
$l\sim$	$\sim l$	Add l	*dully* → *dul* → *dull*, *fully* → *ful* → *full*, *shrilly* → *shril* → *shrill*

Table A.16: General rules for the correction of the base word after a suffix is removed.

- If the removed suffix starts with l and the derived base also ends in l, then the base may have originally ended in *ll*, resulting in one of the l's being dropped during suffixation. Consequently, an alternative with an extra l at the end of the base word is pursued in parallel with the derived base. For example, adjectives in *-ll* lose the final *-l* when the adverbial suffix *-ly* is added; e.g., *full* + *-ly* produces *fully*, not **fullly*.

The above rules are summarized in Table A.16. All of them except the first introduce additional choice points for the derivational component. The resulting choices are examined in turn, according to the depth-first regime described above. The removal of suffixes is performed before the removal of prefixes because it is possible to check easily if a word contains just an adjective plus some prefixes. In that case, the two words should be identical in their last n characters, where n is the length of the adjective in characters. The corresponding test is not so easy when prefixes are removed first, because suffixes introduce morphological changes to the base, as described above. For example, if we start from *deregulation* and remove the prefix *de-*, we cannot easily determine that *regulation* matches *regulate* because of the loss of the final *e* of the base which occurs when the suffix *-ion* is added.

This procedure for the removal of a suffix is applied repeatedly, until no more suffixes can be removed from the derived word, or until a partial match with the adjective has been achieved.[86] After all possible suffixes have been removed, each derived word that is a successful partial match is processed for the removal of prefixes. The prefixes from Table A.12 are tried in turn, using again depth-first search to implicitly order the alternatives considered. The situation is considerably simpler for prefixes, since none of them cause morphological changes to the base. At the end of this phase, either a combination of prefixes and suffixes will have been removed from the word, leaving the target adjective, or the algorithm will determine that the word is not derivable from the adjective. If the word being examined is actually part of a more complex word containing a dash and a match is found, then the counts (types and frequencies) of compounds involving the matching adjective will be incremented. Otherwise, a match will cause the counts of derived words to be increased.

If no match between the word and the adjective can be found with the above method, the algorithm checks whether the word contains the adjective as a prefix, a suffix, or anywhere in the middle. Such relationships can indicate that the word is a compound involving the adjective; e.g., *greenback*, *greenhouse*, and *wintergreen*. Unfortunately, sometimes a sequence

[86] In the sense that the adjective is itself a suffix of the derived word, as explained above.

of characters identical to the adjective may appear within a longer word even though the latter is not a compound derived from the adjective. Naturally, the probability of this wrong association is higher when the adjective of interest is shorter. Also, since most compounds contain just two words, the probability of erroneously identifying an adjective within a longer word is higher when the adjective is a substring but not a prefix or suffix of the word. For example, the automatic morphological analysis component will retrieve *accidentally* and *metallic* (among many others) as words that contain *tall* as a substring. However, since the quality of the identified matches depends on the kind and position of the match, the morphology component keeps separate counts and total frequencies for each match type. For any pair consisting of an adjective and a general word from the vocabulary, one of the following relationships will be identified and reported by the morphological analysis module:

1. The adjective and the word are identical.
2. The word is an inflected form (comparative or superlative) of the adjective.
3. The word is a hyphenated compound containing the adjective.
4. The word is derived from the adjective with the addition of affixes separated from the adjective with hyphens.[87]
5. The word is derived from the adjective with the addition of one prefix according to the rules of Table A.12.
6. The word is derived from the adjective with the addition of one suffix according to the rules of Table A.14.

[87] Almost always this will involve a single prefix followed by a hyphen, for example *pre-dawn* or *re-elect*.

7. The word is derived from the adjective with the addition of one prefix and one suffix.
8. The word is derived from the adjective with the addition of multiple prefixes.
9. The word is derived from the adjective with the addition of multiple suffixes.
10. The word is derived from the adjective with the addition of a combination of multiple prefixes and suffixes.
11. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of one prefix.
12. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of one suffix.
13. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of one prefix and one suffix.
14. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of multiple prefixes.
15. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of multiple suffixes.
16. The word is a hyphenated compound containing another word which is derived from the adjective with the addition of a combination of multiple prefixes and multiple suffixes.
17. The adjective is a prefix or suffix of the word, but none of the rules above apply. The word is possibly a compound involving the adjective.

18. The adjective is a substring of the word and none of the rules above apply. The word is possibly a compound involving the adjective.

19. The adjective and the word are not related morphologically.

Except for the last case, which corresponds to no obvious relationship between the adjective and the general vocabulary word, the types of matches are listed in order of decreased expected accuracy (i.e., certainty that the reported relationship between the two words does indeed hold). To keep with the goal of separating decisions with different confidence levels as much as possible, the morphology module reports separately cases where multiple affixation rules have been applied, since as the number of such rules employed grows so does the probability of an erroneous derivational analysis. Prefixation and suffixation rules are also separated, as the former are generally easier to apply in reverse since the latter frequently effect morphological alterations on the base. The subsequent modules that use the output of the derivational morphology analysis module are free to use only some of the categories of reported relationships, or to combine several categories if the distinctions between them are not considered significant. In the current implementation of the module for the determination of semantic markedness (see Section 8.3), matches of type 3 are considered compounds, matches of types 2 and 4–16 are combined into a single category of "derived words", and matches of types 17 and 18 are ignored because of their lower expected accuracy.

I have run the derivational component described above on several sets of adjectives, using the MRC database as the reference vocabulary list. Using an extensive word list presents several advantages over extracting one from the corpus:

- The word list has already been checked by humans, for typographical mistakes and invalid words (e.g., nonce formations). In addition, other researchers who have access to it can verify its integrity and report any omissions or mistakes.
- The word list typically covers a wide variety of types of texts and domains. By contrast, most corpora are limited in usually one specific domain and a small number of text types (e.g., newspaper articles, or scientific abstracts).
- Even with a large corpus, we cannot expect to find all valid words of the language. For example, the 1987 Wall Street Journal corpus contains 21 million words (tokens) and 215,541 distinct word types. However, 36,555 of these word types are numbers or amounts (e.g., a dollar sign followed by a number), and 70,297 are proper nouns. Only 94,211 word types consist entirely of lower case letters or of two sequences of lower case letters separated by a dash[88] and a considerable number of these are typographical mistakes, nonce formations (e.g., *functionoid*), and foreign or dialectally restricted words. It is obvious that not all possible English words appear in the corpus. For example, the plural noun *liquidizers* appears in the corpus (once), but not the singular form *liquidizer* or the corresponding verb *liquidize*.[89]

---

[88] "Words" not in this set include, for example, *a-wham-bam-alamb-bam-a-lamb-bam-bam*, *people-are-sick-beneath-their-calm-small-town-veneer*, and *$1.10-to-$1.25-a-share*, all appearing just once.

[89] According to the COBUILD dictionary [Sinclair 1987b], *liquidize* means to process food in a *liquidizer* in order to make it liquid; a synonym of *puree*. Compare with the more general *liquefy*, which appears in the corpus 3 times as an infinitive or in the present tense and 25 times in the past tense or as a past participle.

Adjective	Compounds (type 3)	Derived words (types 2 and 4–16)	Superstrings (types 17 and 18)
clean	*clean-bowled, clean-cut, clean-limbed, clean-shaven, dry-clean, spring-clean*	*cleaned, cleaner, cleaners, cleanest, cleaning, cleanlier, cleanliest, cleanliness, cleanly, cleanness, cleans, cleansable, cleanse, cleansed, cleanser, cleansers, cleanses, cleansing, clean-up, clean-ups, dry-cleaned, dry-cleaner, dry-cleaners, dry-cleaning, dry-cleans, periclean, spring-cleaned, spring-cleaning, spring-cleans, unclean, uncleanly*	*heraclean, sophoclean, thericlean*
false	–	*falsehood, falsehoods, falsely, falseness*	*falsetto, falsettos*
happy	*happy-go-lucky, slap-happy, trigger-happy*	*enhappy, happier, happiest, unhappy*	*chappy*
large	*fool-large, large-handed, large-hearted, large-scale*	*alarge, elargement, enlarge, enlarged, enlargement, enlargements, enlarger, enlarges, largeish, largely, largen, largeness, larger, largess, largesse, largest, larget*	–
little	*do-little, little-ease, little-go, little-worth*	*belittle, belittled, belittles, littleness, littler, littlest*	*littlehampton*
smart	*nose-smart, smart-money*	*outsmart, outsmarted, outsmarting, outsmarts, smarted, smarten, smartened, smartening, smartens, smarter, smartest, smarting, smartish, smartly, smartness, smarts, smarty*	*smartweed*
young	–	*younger, youngest, youngish, youngling, youngly, youngness, youngster, youngsters*	–

Table A.17: Words identified by the morphological derivation analysis component as being either compounds involving the adjectives listed or words derived from these adjectives. All the retrieved compounds, derived words, and superstrings are shown for a sample of the adjectives analyzed for the purposes of the corpus-based validation of indicators of semantic markedness (Sections 8.3–8.4). The MRC Psycholinguistic Database [Coltheart 1981; Wilson 1987] was employed as the general vocabulary (word list).

Note, however, that we can easily extract the word list from the corpus, in one pass combined with a sorting step to remove duplicates and compute frequencies. We may want to do so to cover domain-specific terms that appear in the corpus but not in a general word list. Of course, we can combine both approaches by combining the static word list with the set of words appearing in the corpus; a spelling checking component (see Section A.5) or a frequency threshold may be used in this case to remove the most probably invalid words.

The derivational component operates reasonably fast, and produces highly accurate output. Table A.17 shows the sets of derived words identified from the MRC database for several adjectival bases. Since the morphology analysis module is not a central component of this dissertation, I have not formally evaluated its precision or coverage. In the future, I plan to provide quantitative measures for the accuracy of this module and include some estimate of the coverage of the rules, i.e., measure how many valid derived words are not detected by them. The latter is a much harder evaluation problem than estimating precision, since I must consider the words *not* reported by the derivation component, which are significantly more numerous than the words reported by it. However, a coverage estimate can be obtained by considering a reasonably large sample of the vocabulary.