Contextual Word Similarity and Estimation from Sparse Data

Ido Dagan
Department of Mathematics and Computer Science
Bar-Ilan University
Ramat Gan 52900, Israel
dagan@bimacs.cs.biu.ac.il

Shaul Marcus
Department of Computer Science
Technion - Israel Institute of Technology
Haifa 32000, Israel

Shaul Markovitch
Department of Computer Science
Technion - Israel Institute of Technology
Haifa 32000, Israel

February 9, 1995

Abstract

In recent years there is much interest in word cooccurrence relations, such as n-grams, verb-object combinations, or cooccurrence within a limited context. This paper discusses how to estimate the likelihood of cooccurrences that do not occur in the training data. We present a method that makes local analogies between each specific unobserved cooccurrence and other cooccurrences that contain similar words. These analogies are based on the assumption that similar word cooccurrences have similar values of mutual information. Accordingly, the word similarity metric captures similarities between vectors of mutual information values. Our evaluation suggests that this method performs better than existing, frequency based, smoothing methods, and may provide an alternative to class based models. A background survey is included, covering issues of lexical cooccurrence, data sparseness and smoothing, word similarity and clustering, and mutual information.

1 Introduction

Statistical data on word cooccurrence relations play a major role in corpus based approaches for natural language and speech processing. Different types of cooccurrence relations are in use, such as cooccurrence within a consecutive sequence of words (n-grams), within syntactic relations (verb-object, adjective-noun, etc.) or the cooccurrence of two words within a certain limited distance in the context (see Section 2.1). Statistical data about these various cooccurrence relations are employed for a variety of applications, such as speech recognition (Jelinek, 1990), language generation (Smadja and McKeown, 1990), lexicography (Church and Hanks, 1990), machine translation (Brown et al., 1992; Sadler, 1989), semantic clustering (Hindle, 1990), information retrieval (Maarek and Smadja, 1989) and various disambiguation tasks (Grishman et al., 1986; Hindle and Rooth, 1991; Dagan et al., 1991; Dagan and Itai, 1991; Gale et al., 1992b).

A major problem for the above applications is how to estimate the probability of specific word cooccurrences that were not observed in the training corpus. Due to data sparseness in unrestricted language, the aggregate probability of such cooccurrences is large and can easily get to 25% or more, even for a very large training corpus (Church and Mercer, 1993). Since applications often have to compare alternative hypothesized cooccurrences, it is important to distinguish between those unobserved cooccurrences that are likely to occur in a new piece of text and those that are not. These distinctions ought to be made using the data that do occur in the training corpus. Thus, beyond its own practical importance, the sparse data problem provides an informative touchstone for theories on generalization and analogy in linguistic data.

The literature suggests two major approaches for solving the sparse data problem: smoothing and class based methods. Smoothing methods estimate the probability of unobserved cooccurrences using frequency information (Good, 1953; Katz, 1987; Jelinek and Mercer, 1985; Church and Gale, 1991; Gupta et al., 1992) (see section 2.2). Church and Gale (1991) show, that for unobserved bigrams, the estimates of several smoothing methods closely agree (on average) with the probability which is expected using the individual frequencies of the two words and assuming that their occurrence is independent ((Church and Gale, 1991), figure 5). Relying on this result, we will use *frequency based estimation* (using word frequencies) as representative for smoothing estimates of unobserved cooccurrences, for comparison purposes. As will be shown later, the problem with smoothing estimates is that they ignore the expected degree of association between the specific words of the cooccurrence. For example, we would not like to estimate the same probability for two cooccurrences like 'eat bread' and 'eat cars', even if both 'bread' and 'cars' have the same frequency in the corpus.

Class based models (Brown et al., 1992; Pereira and Tishby, 1992; Pereira et al., 1993; Hirschman, 1986; Resnik, 1992; Brill et al., 1990) distinguish between unobserved cooccurrences using classes of "similar" words (see Section 2.3). The probabil-

ity of a specific cooccurrence is determined using generalized parameters about the probability of class cooccurrence. This approach, which follows long traditions in semantic classification, is very appealing, as it attempts to capture "typical" properties of classes of words. However, it is not clear to us that unrestricted language is indeed structured the way it is assumed by class based models. In particular, it is not clear that word cooccurrence patterns can be generalized to class cooccurrence parameters without losing too much information.

This paper suggests an alternative approach which avoids class based generalizations, skipping the intermediate level of word classes. Like some of the class based models, we use a similarity metric to measure the similarity between cooccurrence patterns of words. But then, rather than using this metric to construct a set of word classes, we use it to identify the most specific analogies that can be drawn for each specific estimation. Thus, to estimate the likelihood of an unobserved cooccurrence of words, we use data about other cooccurrences which were observed in the corpus, and contain words which are similar to the given ones. For example, to estimate the likelihood of the unobserved cooccurrence 'negative results', we use cooccurrences such as 'positive results' and 'negative numbers', that do occur in our corpus.

The analogies we make are based on the assumption that similar word cooccurrences have similar values of mutual information (see Section 2.4). Accordingly, our similarity metric was developed to capture similarities between vectors of mutual information values. In addition, we use an efficient search heuristic to identify the most similar words for a given word, thus making the method computationally affordable. Figure 1 illustrates a portion of the similarity network that is induced by the similarity metric (only some of the edges, with relatively high values, are shown). This network may be found useful for other purposes, independently of the estimation method.

The estimation method was implemented using the relation of cooccurrence of two words within a limited distance in a sentence. The proposed method, however, is general and is applicable for any type of lexical cooccurrence. The method was evaluated in two experiments. In the first one we achieved a complete scenario of the use of the estimation method, by implementing a variant of the disambiguation method in (Dagan et al., 1991), for word sense selection in machine translation. The estimation method was then successfully used to increase the coverage of the disambiguation method, with an increase of the overall precision compared to a naive, frequency based, method. In the second experiment we evaluated the estimation method on a data recovery task. The task simulates a typical scenario in disambiguation, and also relates to theoretical questions about redundancy and idiosyncrasy in cooccurrence data. In this evaluation, which involved 300 examples, the performance of the estimation method was by 27% better than frequency based estimation.

Having disambiguation tasks in mind, we developed an estimation method for local comparisons of alternative cooccurrences. The method provides a useful score for comparing the likelihood of unobserved cooccurrences, but cannot be interpreted

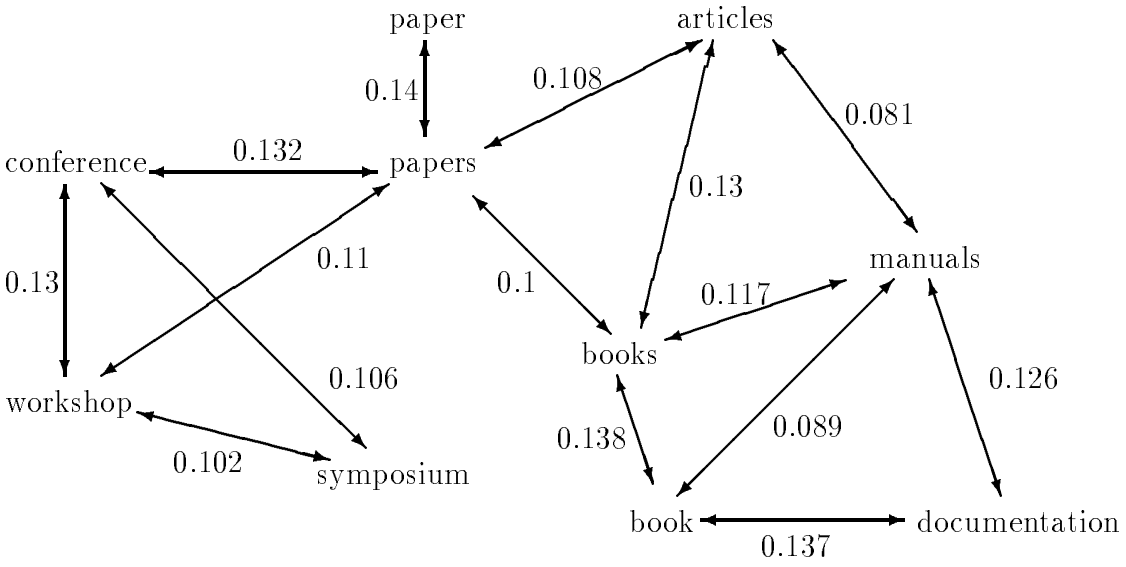

Figure 1: A portion of the similarity network.

directly as a probability measure. Other recent research, also following the similarity-based approach, provides methods for estimating word cooccurrence probabilities (Essen and Steinbiss, 1992; Dagan et al., 1994) (see Section 2.3).

In a general perspective, the similarity-based approach promotes an "unstructured" point of view on the way linguistic information should be represented. While traditional approaches, especially for semantic classification, have the view that information should be captured by the maximal possible generalizations, our method assumes that generalizations should be minimized. Information is thus kept at a maximal level of detail, and missing information is deduced by the most specific analogies, which are carried out whenever needed. Though the latter view seems hopeless for approaches relying on manual knowledge acquisition, it may turn very useful for automatic corpus-based approaches, and better reflect the nature of unrestricted language. From a machine learning point of view, our estimation method can be viewed as a case of *instance-based learning* (Aha et al., 1991) where the distance between instances (cooccurrence pairs) is measured using the similarity metric. Using the k most similar pairs is a manifestation of the k-nearest neighbour paradigm.

Section 2 describes the background for this work, surveying issues of lexical cooccurrence, data sparseness and smoothing, word similarity and clustering and mutual information. Section 3 presents our similarity and estimation methods, and their experimental evaluation on a data recovery task. In section 4 we describe our variant of the sense disambiguation method of Dagan et al. (1991), and its augmentation with similarity based estimation in order to obtain greater coverage. Section 5 draws some

conclusions and discusses issues for further research.

2 Background

2.1 Lexical cooccurrence: types and applications

Statistical information on lexical relations plays a major role in corpus based approaches for natural language and speech processing. Many such approaches rely on counting the frequency of joint cooccurrence of words sharing some relationship, in order to evaluate alternative interpretations of the input. The relationships used in different works can be classified to three major types:

- sequences of consecutive words (n-grams)
- cooccurrence of words within syntactic relations
- cooccurrence of words within a certain distance in the text

The n-gram model is used extensively in language modeling for automatic speech recognition systems (see (Jelinek et al., 1992) for a thorough presentation). In this model, the probability of an occurrence of a word in a sentence is approximated by its probability of occurrence within a short sequence of words. Typically sequences of two or three words (bigrams or trigrams) are being used, and their probabilities are estimated from a large corpus. These probabilites are combined to estimate the a priori probabilities of alternative interpretations of the acoustic signal, in order to select the most probable interpretation.

The information captured by n-grams is, to a large extent, only an indirect reflection of lexical relationships in the language. This is because the production of sequences of words is a consequence of more complex linguistic structures. However, n-grams were shown to have practical advantages, as it is easy to formulate probabilistic models for them, they are very easy to extract from a corpus, and, above all, they proved to provide useful estimations.

Lexical cooccurrence within syntactic relations, such as subject-verb, verb-object and adjective-noun, provide a more direct representation for linguistic information. Statistical data on such cooccurrence relations can be viewed as a statistical alternative to traditional notions of selectional constraints and semantic preferences (Wilks, 1975). As such, these relations were used successfully for various broad coverage disambiguation tasks: prepositional phrase attachment (Hindle and Rooth, 1991), pronoun reference resolution (Dagan and Itai, 1991) and word sense disambiguation

for machine translation (Dagan et al., 1991; Dagan, 1992). The latter work is described in more detail in section 4, as we use a variant of it to test the estimation method of the current paper.

The use of syntactically driven lexical relations relies on the availability of a robust syntactic parser. Although the accuracy of current parsers is not high, the above mentioned works have shown that this accuracy is sufficient for acquiring reliable statistical data, where some noise can be tolerated. Yet, the use of a robust parser may be considered as a practical disadvantage, as such parsers are not widely available, and are not sufficiently efficient or accurate for some applications.

A third type of lexical relationship is cooccurrence within a limited distance in a sentence. Smadja (1990) proposes to use this type of relation as an approximation for significant lexical relations. His proposal relies on an earlier observation that 98% of the occurrences of lexical relations relate words that are separated by at most five words within a single sentence (Martin et al., 1983). Smadja uses this fact to extract lexical collocations, and applies the extracted data to language generation and information retrieval. Our variant of the lexical disambiguation method in (Dagan et al., 1991) (see section 4) makes use of this type of data, as a practical approximation for the use of a parser in the original work. Our experiments, as well as those by Smadja, suggest that cooccurrence within a limited distance in a sentence provides a practical alternative for the use of a robust parser, for the purpose of extracting statistics on lexical cooccurrence relations.

Variants of the latter type of relationship were found useful in two other works. Brown et al. (Brown et al., 1991) use a part of speech tagger to identify relations such as "the first verb to the right" or "the first noun to the left", and then use these relations for sense disambiguation in machine translation. This provides a better approximation for syntactically motivated relations than just requiring a maximal distance between words, while relying on the availability of a part of speech tagger, which is simpler and more available than syntactic parsers.[1] Another variant appears in the work of Gale, Church and Yarowsky (1992a; 1992b) on word sense disambiguation. In this work cooccurrence within a maximal distance of 50 words in each direction was considered. With such a large distance they were capturing context words which mainly identify the topic of discourse. Word cooccurrence within the global context was also used for language modeling in speech recognition, by letting the occurrence of a word affect the probability of other words in the context (Lau et al., 1993).

In this paper we use the relationship of cooccurrence within a limited distance in the sentence (3 content words in each direction). However, the estimation method we present is general and applies to all types of lexical cooccurrence.

[1] In fact, Smadja (1990) also uses a part-of-speech tagger to filter out collocations that do not match some predefined syntactic patterns.

2.2 Sparse data and smoothing

Any approach that relies on statistical data about lexical cooccurrence faces the inherent problem of data sparseness. The distribution of lexical cooccurrences in unrestricted language is such that many cooccurrences have very small probabilities, and consequently most of them do not occur in the training corpus. Nevertheless, there are so many such infrequent cooccurrences, that their aggregate probability is large (e.g. 25%, (Church and Mercer, 1993)). This means that typically several cooccurrences per sentence in a new piece of text were not observed in the training corpus, thus making it very difficult to estimate their probability. As a consequence, any application that uses these estimates will either be limited in its coverage, or otherwise suffer a decrease in its precision due to inaccurate estimates.

The most common and robust approach that was proposed to address the sparse data problem is to smooth the observed frequencies. Church and Mercer (1993, page 11) give a short survey of such smoothing methods (Good, 1953; Katz, 1987; Jelinek and Mercer, 1985; Church and Gale, 1991), promoting their use as a solution for the sparse data problem. All these methods, as well as the methods in (Jelinek et al., 1992) and (Gupta et al., 1992), smooth the observed frequencies by making generalizations, which are based solely on the frequencies of the involved objects: either the frequency of cooccurrence (e.g. the frequency of a bigram or a trigram), or the frequency of the individual words which compose the cooccurrence. Thus, for example, all these methods would give exactly the same probability estimate for two bigrams (say $< w_1, w_2 >$ and $< w_1', w_2' >$) which were not observed in the corpus, and contain words with identical frequencies (i.e. w_i and w_i' have the same frequency, for $i = 1, 2$). The primary drawback of such estimates is that they ignore the identity of the specific words involved in a cooccurrence, while it is obvious that this identity has a major influence on the plausibility of that cooccurrence.[2] The next subsection discusses methods that do relate to the identity of specific words, trying to achieve more reasonable generalizations.

2.3 Word similarity and clustering

Traditionally, generalizations on plausible cooccurrence patterns of words are based on manually defined semantic classes (Allen, 1995, Chapter 10). Classes typically form a hierarchical structure, and selectional constraints are stipulated in terms of the classes in the hierarchy. In recent years, the difficulty to scale up semantic approaches for broad domains led to the development of automatic methods for identifying word similarities and word classes.

[2] The factors that determine the plausibility of the specific cooccurrence can be semantic, syntactic or others. We will not discuss these factors here, though the proposed method is expected to reflect them indirectly.

An early attempt to classify words to semantic classes was performed by members of the Linguistic String Project (Hirschman, 1986; Grishman et al., 1986). Their work was based on Harris' *distributional hypothesis*, which relates the meaning of words to their distribution relative to other words: "...the meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities." (Harris, 1968, p. 12). Following this idea, semantic classes were defined based on similar cooccurrence patterns of words within syntactic relations. This early work relied on small corpora and required a considerable amount of manual intervention.

Sadler (1989) proposes a proximity measure between words, which captures similarity in their cooccurrence patterns with other words. Word similarities are then used to draw analogies between a given combination of words that was not observed in the corpus, and another combination of words that do occur in the corpus, and is most similar to the given one. These analogies determine preferences between competing word combinations, for the purpose of disambiguation in machine translation. In contrast to most other works surveyed in this section, Sadler does not construct word classes for the purpose of generalization. Instead, in an ambiguous situation, the system first computes the degree of similarity between the examples stored in its database and each of the competing alternatives. Then, it prefers that alternative which achieves the highest degree of similarity. A potential disadvantage of Sadler's similarity method is that it ignores the frequencies of words and word cooccurrences. The evaluations reported in his book relied on a small corpus, and did not yield positive results.

Hindle (1990), also motivated by the distributional hypothesis, attributes the failure of previous implementations to the lack of a sufficiently robust syntactic parser that would identify relationships between words. He therefore uses a robust parser that extracts grammatical structures from unrestricted text (Hindle, 1983). His paper proposes a new similarity measure, that reflects the similarity between cooccurrence patterns of nouns in subject-verb and verb-object relations. The measure is based on the concept of *mutual information* (see subsection 2.4), which identifies the degree of association between each noun and verb. Hindle presents interesting examples of similarity between nouns that were identified by his method, and discusses various issues that should be addressed when adopting a similarity based approach.

Brown et al. (1992) develop a class based n-gram model, which uses probabilities of sequences of word classes instead of sequences of individual words (as in a basic n-gram model). Since the number of classes is much smaller than the number of words in the vocabulary, there are significantly fewer parameters to estimate, thus reducing the sparse data problem. According to their theory, the optimal model is achieved if the partition of the vocabulary to classes maximizes the average mutual information between adjacent occurrences of classes in the training corpus. In this model there is no direct measure for the degree of similarity between words, but rather a global

optimization of the partitioning. Since finding such a partition is computationally too expensive, they use a greedy heuristic that merges words into classes, selecting each time a merge operation with minimal loss in the average mutual information. This heuristic, which is still computationally very expensive, was used to partition a 260,000 word vocabulary into 1000 classes. Then, an interpolated tri-gram class-based model was constructed, and was further interpolated with a word-based n-gram model. As a result, they achieved a rather small reduction (3.3%) in the perplexity of the model relative to the original word-based model.

A drawback of the class based model is the loss of information caused by a priori generalization to word classes. This loss is the price paid for the reduction in the number of parameters. Though the classes are supposed to contain "similar" words, there is still a high variation within each class (the average size of a class in the experiment of Brown et al. is 260 words). However, the class based n-gram model distinguishes between members of the same class only by their frequency, ignoring any other information about these words. Thus it suffers to some extent from the same problem as the smoothing techniques mentioned in the previous section, though within the much smaller scale of a single class.

Pereira et al. (Pereira et al., 1993) try to reduce this problem by using "soft" rather than "hard" (boolean) classes. They have adopted techniques from mechanical statistics to construct word clusters that represent "typical" context (cooccurrence) distributions of the words they contain. Cluster membership is probabilistic, such that a word may belong to several classes with a certain membership probability in each of them. Using this technique, inferences about the cooccurrence probability of a certain word do take into account the specific identity of the word, since each word has different membership probabilities in the different clusters. In other words, each word is modeled as a mixture of clusters, each cluster having its own weight. However, information is still lost by the use of clusters in this model, since the method keeps track of cooccurrence distributions of clusters, but not of those of specific words. Thus, distinctions between words are being made only in terms of these general cluster distributions.

In comparison with the above approaches, our method (Dagan et al., 1993) estimates the likelihood of unobserved cooccurrences directly from the available cooccurrence data, relying on the most relevant evidence in each case. This way we skip the intermediate phase of clustering, and avoid the loss of information it necessarily introduces. From this perspective our approach is similar to Sadler's, and compete with the class based estimation methods of Brown et al. and Pereira et al.. The motivation of avoiding clustering appears also in a recent work by Schutze (Schütze, 1993) in which each word has its own individual representation in a multidimensional space. As claimed in his paper, "Any clustering into classes introduces artificial boundaries that cut off words from part of their semantic neighborhood. ... any class size is problematic, since words are either separated from close neighbors or lumped

together with distant terms."

Finally, two recent works also follow the similarity-based approach and apply it to estimate word cooccurrence probabilities. Essen and Steinbiss (1992) measure the similarity between two words using a *confusion probability*, the probability that one word can be substituted for another in an arbitrary context. Dagan, Pereira and Lee (1994) measure word similarity by the *relative entropy*, or *Kullback-Leibler (KL) distance*, between their cooccurrence distributions. Both methods estimate the probability of the cooccurrence of two words based on a weighted average of other word cooccurrence probabilities. The weights in the average are determined according to the similarity measure. These two probabilistic methods have produced encouraging empirical results for bigram language models.

2.4 Mutual Information as a measure for word association

The concept of mutual information, taken from information theory, was proposed as a measure of word association (Church and Hanks, 1990; Hindle, 1990; Jelinek et al., 1992). It reflects the strength of relationship between words by comparing their actual cooccurrence probability with the probability that would be expected by chance. More precisely, the mutual information of two events x and y is defined as follows (Fano, 1961):

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{1}$$

where $P(x)$ and $P(y)$ are the probabilities of the events, and $P(x,y)$ is the probability of the joint event. If there is a strong association between x and y then $P(x,y) \gg P(x)P(y)$ and as a result $I(x,y) \gg 0$. If there is a weak association between x and y then $P(x,y) \approx P(x)P(y)$ and $I(x,y) \approx 0$. If $P(x,y) \ll P(x)P(y)$ then $I(x,y) \ll 0$. In such cases x and y are said to be in complementary distribution.

Using Bayes theorem, mutual information can be presented also in the following way:

$$I(x,y) = \log_2 \frac{P(x|y)}{P(x)} = \log_2 \frac{P(y|x)}{P(y)} \tag{2}$$

Presented this way, mutual information represents the relative change in the probability of observing x when y is present (the amount of information that y provides about x).

Table 1, taken from (Hindle, 1990), demonstrates the utility of the mutual information measure. The table lists objects of the verb *drink*, along with the count and the mutual information of their cooccurrence with this verb. It can be noted that high frequency words (like 'it') yield low mutual information values, compared with other words with the same cooccurrence count.

OBJECT	COUNT	MUTUAL INFORMATION
brunch beer	2	12.34
tea	4	11.75
Pepsi	2	11.75
champagne	4	11.75
liquid	2	10.53
beer	5	10.20
wine	2	9.34
water	7	7.65
anything	3	5.15
much	3	1.25
it	3	1.25
SOME AMOUNT	2	1.22

Table 1: The objects of the verb *drink* (from (Hindle, 1990))

3 An Estimation Method Based on Word Similarity

In the previous section we have discussed the importance of lexical relationships, and focused on the sparse data problem. This section presents an estimation method that uses a measure of word similarity to reduce the sparse data problem. The goal of this method is to estimate the likelihood of cooccurrences that were not observed in the corpus. The basic idea is to estimate the likelihood of a given unobserved cooccurrence using the estimated probabilities of other cooccurrences that *were* observed in the corpus, and contain words which are "similar" to the words of the unobserved cooccurrence. The word similarity measure captures similarities of cooccurrence patterns of words. Though the method was developed and tested for estimating the likelihood of cooccurrence pairs (see below), its definition is general and can be used for other types of lexical relations, such as n-grams or cooccurrence within syntactic relations.

Subsection 3.1 defines the basic concept of a *cooccurrence pair*. Subsection 3.2 describes the way we estimate the likelihood of an unobserved pair, given similar cooccurrence pairs that were observed in the corpus. Subsection 3.3 presents the similarity metric, which is used to identify the most similar words of a given word. Subsection 3.4 describes the specific method we currently use to average the information associated with similar pairs. An efficient search heuristic for finding the most similar words for a given word is presented in subsection 3.5. Finally, in subsection 3.6 we describe an evaluation of the estimation method.

3.1 Definitions

We use the term *cooccurrence pair*, written as (x, y), to denote a cooccurrence of two words in a sentence within a distance of no more than d words. When computing the distance d, we ignore function words such as prepositions and determiners. In the experiments reported here $d = 3$.

A cooccurrence pair can be viewed as a generalization of a bigram, where a bigram is a cooccurrence pair with d=1 (without ignoring function words). As with bigrams, a cooccurrence pair is directional, i.e. $(w_1, w_2) \neq (w_2, w_1)$. This reflects the asymmetry in the linear order of linguistic relations, such as the fact that English verbs tend to precede their objects and follow their subjects.

Let N be the length of the corpus (in words). The total number of cooccurrence pairs is then Nd. Using the Maximum Likelihood Estimator (MLE), the probability of a pair, $P(x, y)$, is estimated by

$$\hat{P}(x, y) = \frac{f(x, y)}{Nd}$$

where $f(x, y)$ is the count of (x, y) in the corpus[3].

Since each occurrence of a word appears in the left position of d different cooccurrence pairs (and the same holds for the right position), the marginal probability for an occurrence of a word in the left (right) position of a cooccurrence pair, $P(w)$, is estimated by:

$$\hat{P}(w) = \frac{d \times f(w)}{Nd} = \frac{f(w)}{N}$$

The mutual information of a cooccurrence pair $I(x, y)$ is estimated by:

$$\hat{I}(x, y) = \log_2 \frac{\hat{P}(x, y)}{\hat{P}(x)\hat{P}(y)} = \log_2 \left(\frac{N}{d} \frac{f(x, y)}{f(x)f(y)}\right) \quad (3)$$

Due to the unreliability of measuring negative mutual information values between content words in corpora that are not extremely large, we have considered in this work any negative value to be 0. We also set $\hat{I}(x, y)$ to 0 if $f(x, y) = 0$. Thus, we assume in both cases that the association between the two words is as expected by chance.

[3]The estimations for pairs that were observed in the corpus may be improved using better estimation methods than MLE, such as in (Good, 1953) or (Church and Gale, 1991). For the sake of simplicity, we have used MLE for observed pairs, which still proved very useful for our purposes.

(w_1, w_2)	$\hat{I}(w_1, w_2)$	$f(w_1, w_2)$	$f(w_1)$	$f(w_2)$
(introduction, describes)	6.85	5	464	277
(book, describes)	6.27	13	1800	277
(section, describes)	6.12	6	923	277
Average:	6.41			

Table 2: The similarity based estimate as an average on similar pairs: $\check{I}(chapter, describes) = 6.41$

3.2 Estimation for an unobserved pair

Assume we have at our disposal a method for determining similarity between cooccurrence patterns of two words (as will be described in the next subsection). We say that two cooccurrence pairs, (w_1, w_2) and (w_1', w_2'), are *similar* if w_1' is similar to w_1 and w_2' is similar to w_2. A special (and stronger) case of similarity is when the pairs differ only in one of their words (e.g. (w_1, w_2') and (w_1, w_2)). This special case is less susceptible to noise, as we replace only one of the words in the pair. In our experiments, which involved rather noisy data, we have used only this restricted type of similarity. The method is presented, though, in terms of the general case.

What analogies can be drawn between two similar cooccurrence pairs, (w_1, w_2) and (w_1', w_2')? Their probabilities cannot be expected to be similar, since the probabilities of the words in each pair might be different. However, since we assume that w_1 and w_1' have similar cooccurrence patterns, and so do w_2 and w_2', it is reasonable to assume that the mutual information of the two pairs will be similar (recall that mutual information measures the degree of association between the words of the pair).

Consider for example the pair *(chapter, describes)*, which does not occur in our corpus (see section 3.6.1 for details on the corpus). This pair was found to be similar to the pairs *(introduction, describes)*, *(book, describes)* and *(section, describes)*, which do occur in the corpus. Since these pairs occur in the corpus, we estimate their mutual information values using equation 3, as shown in Table 2. We then take the average of these mutual information values as the *similarity based estimate* for $I(chapter, describes)$, denoted as $\check{I}(chapter, describes)$[4]. This represents the assumption that the word 'describes' is associated with the word 'chapter' to a similar extent as it is associated with the words 'introduction', 'book' and 'section'. Table 3 demonstrates how the analogy is carried out also for a pair of unassociated words, such as $(chapter, knows)$.

[4]We use $\check{I}$ for similarity based estimates, and reserve $\hat{I}$ for the traditional maximum likelihood estimate. The similarity based estimate will be used for cooccurrence pairs that do not occur in the corpus.

(w_1, w_2)	$\hat{I}(w_1, w_2)$	$f(w_1, w_2)$	$f(w_1)$	$f(w_2)$
(introduction, knows)	0	0	464	928
(book, knows)	0	0	1800	928
(section, knows)	0	0	923	928
Average:	0			

Table 3: The similarity based estimate for a pair of unassociated words: $\check{I}(chapter, knows) = 0$

In the general case, $\check{I}(w_1, w_2)$ is computed using the mutual information values of a set of most similar pairs, according to the following scheme:

$$\check{I}(w_1, w_2) = Avg\{\hat{I}(w_1', w_2') | (w_1', w_2') \in MostSim(w_1, w_2)\} \quad (4)$$

Avg is some averaging function and $MostSim$ is a set of most similar pairs, as determined using the similarity measure. Section 3.4 describes our current implementation of Avg and $MostSim$.

Having an estimate for the mutual information of a pair, we can estimate its expected frequency in a corpus of the given size using a variation of equation 3:

$$\check{f}(w_1, w_2) = \frac{d}{N} f(w_1) f(w_2) 2^{\check{I}(w_1, w_2)} \quad (5)$$

In our example, $f(chapter) = 395$, $N = 8,871,126$ and $d = 3$, getting a similarity based estimate of $\check{f}(chapter, describes) = 3.15$. This expected frequency of the pair is much higher than expected by the *frequency based estimate* (0.037), reflecting the plausibility of the specific combination of words[5]. On the other hand, the similarity based estimate for $\check{f}(chpater, knows)$ is 0.124, which is identical to the frequency based estimate, reflecting the fact that there is no expected association between the two words (notice that the frequency based estimate is higher for the second pair, due to the higher frequency of 'knows').

It should be pointed out that $\check{I}$ and $\check{f}$ provide useful scores for comparing alternative cooccurrences in disambiguation. However, they do not provide a *probabilistic* estimate, as they are not normalized such that the probabilities of all possible pairs would add up to one. Also, we have not incorporated the actual frequency of a pair into the estimate, which is irrelevant when comparing pairs of the same frequency

[5]The *frequency based estimate* for the expected frequency of a cooccurrence pair, assuming independent occurrence of the two words and using their individual frequencies, is $\frac{d}{N} f(w_1) f(w_2)$. As mentioned in the introduction, we use this estimate as representative for smoothing estimates of unobserved cooccurrences.

(0, in our case). Thus, $\check{f}(w_1, w_2)$ can be interpreted as the frequency that we would expect for this pair, based on similar pairs, without knowing its actual frequency in the corpus.

3.3 The Similarity Metric of Two Words

3.3.1 Defining the Metric

Assume that we need to determine the degree of similarity between two words, w_1 and w_2. Recall that if we decide that the two words are similar, then we may infer that they have similar mutual information with some other word, w. This inference would be reasonable if we find that on average w_1 and w_2 indeed have similar mutual information values with other words in the lexicon. The similarity metric therefore measures the degree of similarity between these mutual information values.

We first define the similarity between the mutual information values of w_1 and w_2 relative to a single other word, w. Since cooccurrence pairs are directional, we get two measures, defined by the position of w in the pair. The *left context similarity* of w_1 and w_2 relative to w, termed $sim_L(w_1, w_2, w)$, is defined as the ratio between the two mutual information values, having the larger value in the denominator:

$$sim_L(w_1, w_2, w) = \frac{\min(I(w, w_1), I(w, w_2))}{\max(I(w, w_1), I(w, w_2))} \tag{6}$$

This way we get a scale of similarity values between 0 and 1, in which higher values reflect higher similarity. If both mutual information values are 0, then $sim_L(w_1, w_2, w)$ is defined to be 0. The *right context similarity*, $sim_R(w_1, w_2, w)$, is defined equivalently, for $I(w_1, w)$ and $I(w_2, w)$[6].

Using definition 6 for each word w in the lexicon, we get $2 \cdot l$ similarity values for w_1 and w_2, where l is the size of the lexicon. The general similarity between w_1 and w_2, termed $sim(w_1, w_2)$, is defined as a weighted average of these $2 \cdot l$ values. It is necessary to use some weighting mechanism, since small values of mutual information tend to be less significant and more vulnerable to noisy data. We found that the maximal value involved in computing the similarity relative to a specific word provides a useful weight for this word in computing the average. Thus, the weight for a specific left context similarity value, $W_L(w_1, w_2, w)$, is defined as:

$$W_L(w_1, w_2, w) = \max(I(w, w_1), I(w, w_2)) \tag{7}$$

(notice that this is the same as the denominator in definition 6). This definition provides intuitively appropriate weights, since we would like to give more weight

[6] In the case of cooccurrence pairs, a word may be involved in two types of relations, being the left or right argument of the pair. The definitions can be easily adapted to cases in which there are more types of relations, such as provided by syntactic parsing.

similar words	SIM
aspects	1.00
topics	0.1
areas	0.088
expert	0.079
issues	0.076
approaches	0.072

Table 4: The similar words of *aspects*

to context words which have a large mutual information value with at least one of w_1 and w_2. The mutual information value with the other word may then be large, providing a strong "vote" for similarity, or may be small, providing a strong "vote" against similarity. The weight for a specific right context similarity value is defined equivalently. Using these weights, we get the following weighted average as the general definition of similarity:

$$sim(w_1, w_2) = \frac{\sum_{w \in lexicon} sim_L(w_1, w_2, w) \cdot W_L(w_1, w_2, w) + sim_R(w_1, w_2, w) \cdot W_R(w_1, w_2, w)}{\sum_{w \in lexicon} W_L(w_1, w_2, w) + W_R(w_1, w_2, w)} = \frac{\sum_{w \in lexicon} \min(I(w, w_1), I(w, w_2)) + \min(I(w_1, w), I(w_2, w))}{\sum_{w \in lexicon} \max(I(w, w_1), I(w, w_2)) + \max(I(w_1, w), I(w_2, w))} \quad (8)$$

The values produced by this metric have an intuitive interpretation, as denoting a "typical" ratio between the two mutual information values of each of the two words with another third word. Table 4 lists the six most similar words to the word 'aspects' according to this metric, based on our corpus. Out of the six words, five can be considered as similar to 'aspects' according to our own intuition. The only problematic word seems to be 'expert', which may have been found as similar to 'aspects' due to some noise in the data. The effect of such noise on our estimation method is reduced by considering sets of similar words, as was explained earlier.

3.3.2 Properties of similarity

It is interesting to point out the properties of our similarity metric, and compare them with intuitive notions of the "similarity" concept:

- **Reflexivity:** The metric is reflexive, as $sim(w, w) = 1$, which is the highest similarity score. This agrees with an intuition that an object is most similar to

itself.

- **Symmetry:** The metric is symmetric, as $sym(w_1, w_2) = sym(w_2, w_1)$. This agrees with an intuition that the degree of similarity between an object A and another object B is the same as between B and A.

- **Intransitivity:** The metric is not transitive, e.g. high values of $sym(w_1, w_2)$ and $sym(w_2, w_3)$ do not imply a high value of $sym(w_1, w_3)$. This agrees with an intuition that an object A can be similar to another object B in some aspects, while B is similar to a third object C in other aspects. In such a case, A will not necessarily be similar to C. Analogously, our metric may establish similarity between w_1 and w_2 based on one set of words which commonly cooccur with both, while finding another set of commonly cooccurring words for w_2 and w_3. In this case w_1 and w_3 will not necessarily be similar.

  The intransitivity of the metric expresses our intended deviation from the approach of clustering words to equivalence classes, as was stated in the introduction.

3.3.3 Comparison with other similarity metrics

Hindle (Hindle, 1990) defines similarity between nouns according to their cooccurrence with verbs in verb-object and subject-verb patterns, as identified by a syntactic parser. The formula used by Hindle to define the similarity of two nouns is very similar to the numerator of definition 8. Our evaluations have shown that omitting the denominator from the definition of the metric, as in Hindle's definition, has a negative effect on its accuracy. This is because the denominator serves as a normalization factor, which compensates for differences in word frequencies. Using Hindle's metric, frequent words tend to be defined as similar to many other words, since they cooccur with many other words, and thus there will be many nonzero terms in the sum of the enumerator. Our definition, on the other hand, avoids this bias, since for frequent words the denominator will also be large.

To demonstrate the effect of the normalization factor, we have computed the six most similar words to *aspects* using equation 8 without the denominator. The result is shown in table 5. As can be easily seen, the list of similar words found this way is much inferior (compare with table 4), and consists of irrelevant frequent words.

In their work on word clustering (see section 2.3), Pereira, Tishby and Lee (1993) use a measure of *dissimilarity* between words rather than similarity. The dissimilarity between two words is defined as the relative entropy (Kullback-Leibler distance) of the corresponding conditional distributions of other words given each of the two words:

$$D(w_1||w_2) = \sum_w P(w|w_1) \ln \frac{P(w|w_1)}{P(w|w_2)} \tag{9}$$

similar words	SIM
aspects	1437.44
systems	446.28
programming	432.91
such	426.03
its	403.79
system	400.87

Table 5: The similar words to *aspects*, omiting the denominator of equation 8.

Unlike our similarity metric, this distance metric is asymmetric, as $D(w_1||w_2) \neq D(w_2||w_1)$. A symmetric metric can be easily defined using the sum of these two distances[7]:

$$D(w_1||w_2) + D(w_2||w_1) = \sum_w [P(w|w_1) - P(w|w_2)] \ln \frac{P(w|w_1)}{P(w|w_2)} \tag{10}$$

This metric is pretty much analogous to our similarity metric, having logarithm of ratios instead of ratio of logarithms (the right multiplicand) and a somewhat different weighting factor (the left multiplicand)[8]. Definition 10 is thus expected to produce similar results to those of our metric, while having a better mathematical motivation. On the other hand, it raises smoothing problems when either $\hat{P}(w|w_1)$ or $\hat{P}(w|w_2)$ is zero. It is left for further research to apply this definition, or the original dissimilarity measure (equation 9), for the similarity based estimation method.

3.4 Averaging mutual information values of similar pairs

This section describes our current implementation for averaging the mutual information values of similar pairs (in equation 4). As explained before, this average provides the estimate for the mutual information value of a given unobserved pair. A similar pair is constructed by replacing one of the words of the given pair with a similar word. We will use the term *L-similar pair* to denote a pair which was constructed by replacing the left word of a given pair, and *R-similar pair* for a replacement of the right word of a given pair. Since we replace only one word at a time, we define the degree of similarity between two pairs as the degree of similarity between the original word of the given pair and the word that replaces it.

[7] We thank Fernando Pereira for pointing this out.

[8] It is left for the reader to work out the details of the analogy. Notice that $\frac{P(w|w_1)}{P(w|w_2)} = \frac{P(w,w_1)}{P(w_1)} \Big/ \frac{P(w,w_2)}{P(w_2)}$. This is the ratio of the terms which appear in the definitions of the two corresponding mutual information values, as the arguments of the logarithm.

Let (v, u) be a cooccurrence pair, $v_1, v_2 \ldots v_k$ the k most similar words to v, and $u_1, u_2 \ldots u_k$ the k most similar words to u. Let $I(v_1, u), I(v_2, u) \ldots I(v_k, u)$ be the mutual information values for the k most L-similar pairs to (v, u) and NZ_L be the number of non-zero ones. Analogously, NZ_R is the number of non-zero mutual information values among $I(v, u_1), I(v, u_2) \ldots I(v, u_k)$ (the k most R-similar pairs).

Using the k L-similar pairs, we take the average of their non zero mutual information values as an L-similar estimate for $I(v, u)$:

$$\check{I}_L(v, u) = \frac{\sum_{i=0}^{k} \hat{I}(v_i, u)}{NZ_L}$$

If $NZ_L = 0$, then $\check{I}_L(v, u)$ is defined as 0. The value of k is a parameter which we set experimentally for the specific corpus ($k = 6$ in our experiments). Analogously, for the most R-similar pairs we define:

$$\check{I}_R(v, u) = \frac{\sum_{i=0}^{k} \hat{I}(v, u_i)}{NZ_R}$$

Finally, we define the combined estimate for $I(v, u)$ as the maximum of the L-similar and R-similar estimates:

$$\check{I}(v, u) = \max(\check{I}_L(v, u), \check{I}_R(v, u)) \tag{11}$$

The definition of the specific averaging procedure was guided by practical motivations, and is the result of experimentation with several variations of this procedure. We were seeking estimates that set significant preferences among alternative pairs, using a small amount of supportive data. This is because in many cases the data is too sparse, such that many of the cooccurrence pairs similar to the given unobserved pair do not occur in the corpus as well. For this reason we average only non-zero mutual information values, and take the maximum of the L-similar and R-similar estimates. This way, words that were inappropriately included in the similarity list of one of the words of the pair, and therefore construct pairs with a zero mutual information value, do not affect the estimate. On the other hand, reasonable estimates can be achieved as long as the similarity list does contain some truly similar words. Further research and experimentation are still necessary to devise a better averaging procedure.

3.5 Heuristic search for most similar pairs

The estimation method requires that we know the k most similar words to a given word w. A naive method for finding these k words is to compute the similarity between w and each word in the lexicon. The complexity of computing the similarity between two words is $O(l)$, where l is the size of the lexicon, and therefore the complexity of

finding the k most similar words for a given word, using the naive method, is $O(l^2)$, and $O(l^3)$ to do this for all the words in the lexicon.

Obviously, the naive method is extremely expensive for a large scale lexicon (e.g. 100,000 words), even if we intend to compute all similarities off line and then store the results in the lexicon. To handle this problem we have developed a technique that approximates the original similarity method, and requires a considerably smaller amount of computation. In presenting this technique, we will use the term *neighbor* for words in a cooccurrence pair. In (w_1, w_2), w_1 is a *left neighbor* of w_2, and w_2 is a *right neighbor* of w_1.

The approximation technique is based upon the following observations. When computing $SIM(w_1, w_2)$, common neighbors with high mutual information values with both w_1 and w_2 make the largest contributions to the value of the similarity measure (equation 8). Also, high and reliable mutual information values are typically associated with relatively high frequencies of the involved cooccurrence pairs. These observations are captured by the following definition of *strong neighbors*:

Definition 1 Let t_I and t_f be some specific thresholds. Two words, x and y, are *strong neighbors* if $I(x, y) > t_I$ and $f(x, y) > t_f$.

Using this definition, instead of computing $SIM(w, w_i)$ for every word w_i in the lexicon, we would like to compute it only for a small set of words which share enough strong neighbors with w, and are thus likely to be similar to w. We will term such a word a *candidate* for being similar to w, according to the following definition:

Definition 2 Let t_N be some specific threshold. A *candidate* is a word which shares more than t_N strong neighbors with the given word w.

The values for the above three thresholds were tuned experimentally to $t_I = 5$,$t_f = 4$ and $t_N = 6$.

The candidates for being similar to a given word w are identified by the following search procedure:

1. Collect all the strong neighbors of the given word w. Let $N_L(w)$ be the set of all strong left neighbors of w and $N_R(w)$ the set of its strong right neighbors.

2. Collect the strong neighbors of all words in $N_L(w)$ and in $N_R(w)$ (strong neighbors of strong neighbors of w), as potential words for being candidates. More formally, we compute the set C as follows:

$$C = \{w_{rl} \in N_R(w_l) | w_l \in N_L(w)\} \bigcup \{w_{lr} \in N_L(w_r) | w_r \in N_R(w)\}$$

   Notice that for a word w_l in $N_L(w)$ we compute $N_R(w_l)$, such that both the words in $N_R(w_l)$ and w will have w_l as a *left* neighbor (similarly for right neighbors).

naive method		approximation technique	
similar words	*SIM*	*similar words*	*SIM*
aspects	1.000	aspects	1.000
topics	0.100	topics	0.100
areas	0.088	areas	0.088
expert	0.079	expert	0.079
issues	0.076	issues	0.076
approaches	0.072	concerning	0.069

Table 6: The most similar words of *aspects:* heuristic and exhaustive search produce nearly the same results.

3. Collect the candidates out of the set C, i.e. those words which share at least t_N strong neighbors with w.

The candidates are thus found among the strong neighbors of the strong neighbors of w. We then compute the similarity only between w and the candidates, and take the k most similar of them as an approximation for the k most similar words of w (as theoretically defined over the entire lexicon). Assuming that $|N_R(w)| \ll l$ and $|N_L(w)| \ll l$, the complexity of finding the k most similar words is reduced significantly. In practice we found the sizes of these sets never exceeds 1000, and are typically much smaller. For the example given in table 6 below, the naive method required 17 minutes of CPU time on a Sun 4 workstation, while the approximation required only 7 seconds (having 58 strong neighbors).

To illustrate the performance of the approximation technique, we show the 6 most similar words to the word *'aspects'* using this technique, and compare them to the most similar words computed by the naive exhaustive method. The results are shown in table 6. Table 7 lists the left and right common strong neighbors of the words *'aspects'* and *'topics'*, which led to the selection of *'topics'* as a candidate.

3.6 Evaluation: A data recovery task

3.6.1 The Corpus

The corpus used for our experiments consists of articles posted to the USENET news system. The articles were collected from news groups that discuss computer related topics. The length of the corpus is 8,871,125 words (tokens), and the lexicon size (distinct types, at the string level) is 95,559. The type of text in our corpus is quite

left strong neighbors	right strong neighbors
various	progress
list	programming
discussion	submissions
relevant	
many	
conference	

Table 7: The strong neighbors of *aspects* and *topics*

noisy, for several reasons[9]:

- Sentence Duplication: mainly due to the inclusion of the original articles when submitting a reply.
- Short and incomplete sentences: the average length of a sentences is 8 words.
- Irrelevant information, such as person and device names.

We have constructed a database containing the frequency counts of all cooccurrence pairs. The total number of distinct cooccurrence pairs in the corpus is 4,339,261, out of which 1,377,653 (31.75%) appear more than once. Only these non-singleton pairs were stored in the database, so that a pair which occurred only once was regarded in our experiments as not occurring at all.

3.6.2 The evaluation

The most important criterion for judging the similarity based estimation method is measuring its contribution to other natural languages processing tasks. Such an evaluation is described in the next section, where similarity based estimation is used to enhance an existing word sense disambiguation method. In this subsection we describe an additional evaluation, with a larger set of examples, which simulates to a large extent a typical scenario in disambiguation tasks. In this evaluation, the estimation method had to distinguish between members of two sets of cooccurrence pairs, one of them containing pairs with relatively high probability and the other pairs with low probability.

Ideally, this evaluation should be carried out using a large set of held out data, which will provide good estimates for the true probabilities of the pairs in the test sets. The estimation method should then use a much smaller training corpus, in which none

[9] A training corpus of higher quality is expected to yield higher performance.

of the example pairs occur, and then should try to recover the probabilities which are known to us from the held out data. However, such a setting requires that the held out corpus would be several times larger than the training corpus, while the latter should be large enough for robust application of the estimation method. This was not feasible with the size of our corpus, and the rather noisy data we had.

To avoid this problem, we obtained the set of pairs with high probability from the training corpus, selecting pairs that occur at least 5 times. We then deleted these pairs from the data base which is used by the estimation method, forcing the method to recover their probabilities using the other pairs of the corpus. The second set, of pairs with low probability, was obtained by constructing pairs that do not occur in the corpus. The two sets, each of them containing 150 pairs, were constructed randomly and were restricted to words with individual frequencies between 500 and 2500. We term these two sets as the *occurring* and *non-occurring* sets.

The task of the similarity based estimation method was to classify the 300 pairs into the two original sets, without access to the deleted frequency information. This task, in which the method has to recover the deleted data (to the extent which is necessary for the classification), is by no means trivial. Trying to use individual word frequencies will result in performance close to that of using random selection. This is because the individual frequencies of all participating words are within the same range of values.

To address the task, the following procedure was used: The expected frequency of each cooccurrence pair was estimated using the similarity-based estimation method. If the frequency was found to be above 2.5 (which was set arbitrarily as the average of 5 and 0), the pair was recovered as a member of the *occurring* set. Otherwise, it was recovered as a member of the *non-occurring* set.

Out of the 150 pairs of the *occurring* set, our method correctly identified 119 (79%). For the *non-occurring* set, it correctly identified 126 pairs (84%). Thus, the method achieved an overall accuracy of 81.6%. Optimal tuning of the threshold, to a value of 2, improves the overall accuracy to 85%, where about 90% of the members of the *occurring* set and 80% of those in the *non-occurring* set are identified correctly. This is contrasted with the optimal discrimination that could be achieved by *frequency based estimation* (see section 3.2), which is 58%.

Figures 2 and 3 illustrate the results of the experiment. Figure 2 shows the distributions of the estimated frequency of the pairs in the two sets, using similarity based and frequency based estimation. It clearly indicates that the similarity based method gives high estimates mainly to members of the *occurring* set and low estimates mainly to members of the *non-occurring* set. Frequency based estimation, on the other hand, makes a much poorer distinction between the two sets. Figure 3 plots the two types of estimation for pairs in the *occurring* set as a function of their true frequency in the corpus. It can be seen that while the frequency based estimates are almost

always low, the similarity based estimates are in most cases closer to the true value.

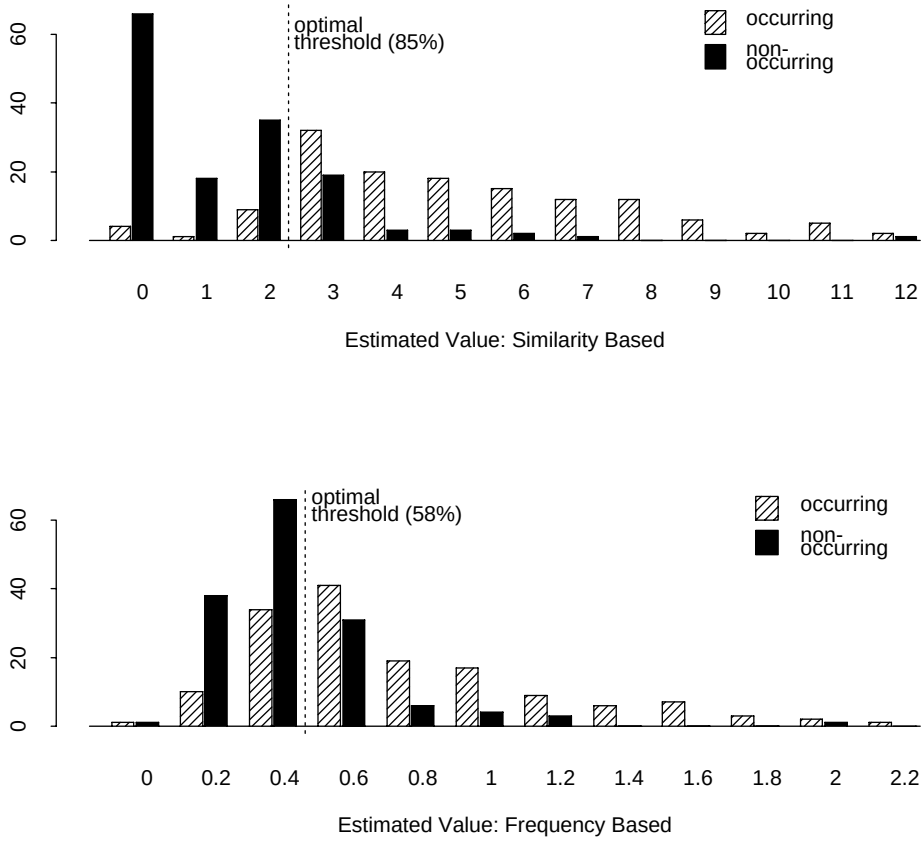

Figure 2: Frequency distributions of estimated frequency values for *occurring* and *non-occurring* sets.

To illustrate the experiment, consider the cooccurrence pair $(useful, features)$, which has the following mutual information and frequency values:

$$I(useful, features) = 4.29$$

$$f(useful, features) = 15$$

Table 8 summarizes the cooccurrence pairs most R-similar and L-similar to $(useful, features)$. According to equation 11, the estimate for the mutual information is:

$$\check{I}(useful, features) = \max(3.15, 2.11) = 3.15$$

and using equation 5 the estimated frequency is:

$$\check{f}(useful, features) = 6.81$$

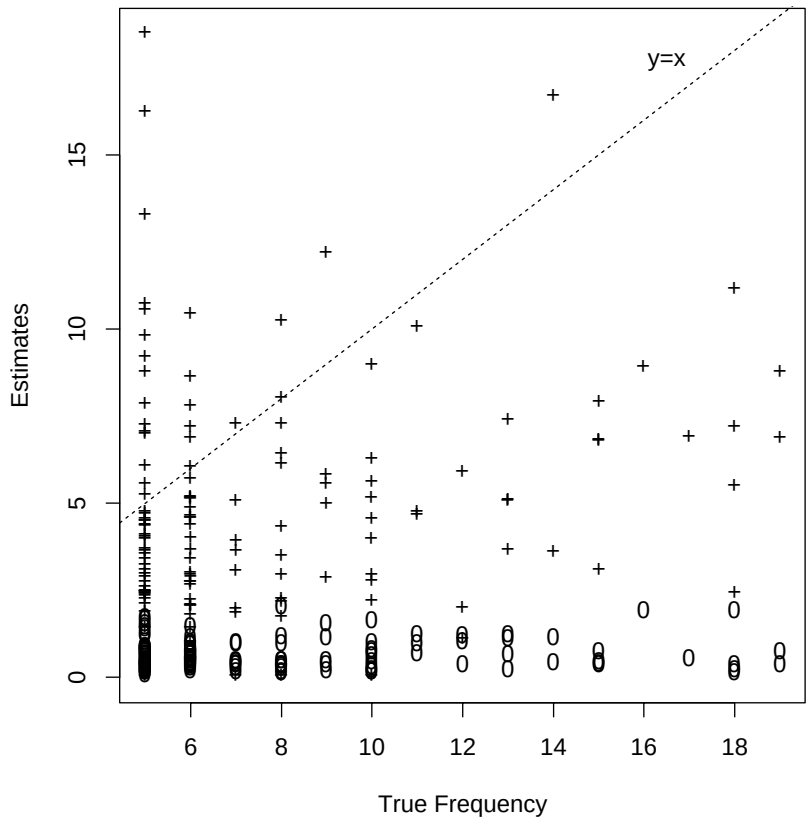

Figure 3: Similarity based estimation ('+') and frequency based estimation ('0') for the expected frequency of members of the *occurring* set, as a function of the true frequency.

L-similarity	I	R-similarity	I
(good,features)	1.96	(useful,feature)	4.58
(interested,features)	1.08	(useful,stuff)	3.98
(interesting,features)	2.66	(useful,things)	3.35
(important,features)	4.27	(useful,versions)	1.10
(looking,features)	0.58	(useful,functions)	2.75
$\check{I}(useful, features) =$	2.11	$\check{I}(useful, features) =$	3.15

Table 8: The estimation of *'useful features'*

L-similarity	I	R-similarity	I
(video,Mon)	0.00	(sound,tue)	0.00
(sounds,Mon)	0.00	(sound,wed)	0.00
(play,Mon)	0.00	(sound,thu)	0.00
(game,Mon)	0.00	(sound,fri)	0.00
(games,Mon)	0.00	(sound,sat)	0.00
$\check{I}(sound, Mon) =$	0.00	$\check{I}(sound, Mon) =$	0.00

Table 9: The estimation of *'sound Mon'*

It can be seen that although the estimate is not accurate, it managed to recover the plausibility of the pair, using evidence which largely agrees with our semantic intuition.

Table 9 summarizes the mutual information of the cooccurrence pairs most R-similar and L-similar to the randomly constructed pair $(sound, Mon)$, which does not occur in the corpus ('Mon' stands for 'Monday'). As can be seen from the table, our method estimated this pair to have zero mutual information.

4 Augmenting Sense Disambiguation with Similarity Based Estimation

The purpose of the evaluation described in this section is to test whether similarity based estimation can enhance the performance of a disambiguation technique. Typically in a disambiguation task, different cooccurrences correspond to alternative interpretations of the ambiguous construct. If none of the alternative cooccurrences was observed in the training corpus then it is difficult to decide which of them is

more plausible. For instance, experiments of the word sense disambiguation method in (Dagan et al., 1991), as well as experiments reported here (see section 4.3), showed that the disambiguation method was not applicable in more than 30% of the cases. The major part of these inapplicable cases were due to the sparse data problem, i.e., none of the alternative cooccurrences appeared in the corpus.

Our enhancement of the disambiguation method invokes the similarity based estimation method in such inapplicable cases. It then uses the provided estimates to make preferences between the alternative cooccurrences. It should be noted that for disambiguation purposes, the estimates for the probability of the alternative cooccurrences should reflect the relative order between their true probabilities. However, a consistent bias in the estimate is usually not harmful, as it still preserves the correct relative order between the alternatives.

To carry out the evaluation, we implemented a variant of the disambiguation method in (Dagan et al., 1991; Dagan, 1992), for sense disambiguation in machine translation. This method selects the translation of an ambiguous word by comparing the frequencies of alternative cooccurrences which correspond to alternative target words. We term this method as *TWS*, for *Target Word Selection*. In our implementation of the TWS method, cooccurrence pairs were used instead of lexical cooccurrence within syntactic relations (as in the original work), to save the need of parsing the corpus.

As noted in (Dagan et al., 1991), the TWS method resembles methods for language modeling in recognition tasks, such as OCR or speech recognition. It considers the words of the source language as a noisy source for target language words, where alternative target words correspond to alternative senses of source words. Then, statistics from a target language corpus are used to set preferences among the alternative translations.

Subsection 4.1 provides a brief description of the TWS method. Subsection 4.2 describes the variant of the method that we have implemented. Subsection 4.3 describes an experiment performed to test the "basic" TWS method and subsection 4.4 presents the augmentation of the *TWS* method with similarity-based estimation.

4.1 The *Target Word Selection* method

We illustrate the TWS method using the following ambiguous Hebrew sentence (transcripted to Latin letters) taken from (Dagan et al., 1991):

Nose ze mana' mi-shtei ha-mdinot mi-**laxtom** 'al **xoze** shalom. (12)

This sentence translates as: "This issue prevented the two countries from signing a peace treaty". The word 'laxtom' has several senses, and accordingly has four possible translations to English: 'sign', 'seal', 'close' and 'finish'. The word 'xoze' has two

possible translations: 'treaty' and 'contract'. Consequently, there are 8 alternative selections of the target English words ("sign treaty", "seal contract" etc.).

The TWS method selects a translation for each ambiguous source word using statistics from a *target* language corpus, about the alternative cooccurrence patterns that correspond to each selection. In example 12, the word 'treaty' is preferred as the translation of 'xoze', since the pattern *(adj-noun: peace treaty)* appeared 49 times in the training corpus, while the pattern *(adj-noun: peace contract)* did not appear at all. Similarly, the word 'sign' is preferred as the translation of 'laxtom', since the pattern *(verb-object: sign treaty)* appeared 79 times in the corpus, while all the other alternative patterns appeared at most twice[10].

When making a selection between alternative cooccurrence patterns, the TWS method selects one of them only if it is significantly more frequent than the others. The decision criterion considers the ratio between the counts of the two most frequent alternatives, n_1 and n_2. The statistical significance for this ratio is determined using a confidence interval for the ratio of two probabilities in a multinom (see (Dagan et al., 1991) for details). If the significance criterion is not met then no decision is made (an inapplicable case).

There are two reasons for inapplicable cases:

1. Both counters, n_1 and n_2, are very low as a result of data sparseness.
2. The values of both counters are significantly large, but they are very close to each other. As a result, the proportion $\frac{n_1}{n_2}$ is very small.

The second type of cases seems to be "a lost case" for the TWS method, since there is enough statistical evidence which indicates that there is no clear preference between the alternatives. The first type, however, which constitutes the majority of the inapplicable cases, may be resolved using a technique that overcomes the sparse data problem, such as similarity based estimation.

4.2 Using cooccurrence pairs for target word selection

The original TWS method uses statistics on word cooccurrence within syntactic relations, which are collected from an automatically parsed corpus. Our implementation approximates these statistics using cooccurrence pairs, thus saving the need to parse the corpus. We still assume parsing of the ambiguous source sentence (as in most ma-

[10]It should be emphasized that the TWS method uses only a *monolingual* target corpus, and not a bilingual corpus as in other methods (Brown et al., 1991; Gale et al., 1992b). The alternative cooccurrence patterns in the target language, which correspond to the alternative translations of the ambiguous source words, are constructed using a bilingual lexicon.

syntactic relationship	cooccurrence pair	
	first word	*second word*
adjective-noun	adjective	noun
subject-verb	subject	verb
verb-object	verb	object
noun-noun	preserving the word order in the sentence	
verb-verb	preserving the word order in the sentence	

Table 10: The required word order in cooccurrence pairs for approximating syntactic relations.

chine translation systems), which is necessary to identify the important cooccurrence relations for each ambiguous word.

Assume we need to approximate the frequency of a specific word cooccurrence within a syntactic relation R, denoted by $(R : w_1\ w_2)$ (such as *(verb-object: sign treaty)*). Occurrences of this pattern may correspond to two different *cooccurrence pairs*: (w_1, w_2) and (w_2, w_1) (recall that cooccurrence pairs are directional). To minimize the noise introduced by using cooccurrence pairs, we use only the word order in which the syntactic relation appears most frequently. For example, counting *adjective-noun* tuples in a parsed English corpus reveals that adjectives appear most frequently before their related nouns. Table 10 summarizes the word order in the cooccurrence pairs for approximating various types of syntactic relations.

The use of cooccurrence pairs for the above approximations introduces noise that leads to lower precision of the TWS method. On the other hand, it saves a considerable amount of parsing resources and, more important, makes the method feasible also when a robust parser of the target language is not available.

4.3 Experimenting with the TWS method

The test set used for the experiment consists of 78 Hebrew sentences that were taken out of a book about computers. The computer topic was chosen to get correspondence between the domain of the training corpus and that of the test sentences.

The preparation of examples was conducted by simulating a machine translation process by a professional human translator. The output of the process was a set of English sentences where each ambiguous Hebrew word is represented by the set of all its possible translations. The translations were taken from a Hebrew-English dictionary, following the same guidelines as in (Dagan et al., 1991). Each set of alternatives was marked with its syntactic roles relative to other words in the sentence.

		Word Frequency		
		correct	*incorrect*	**Total**
TWS method	*correct*	120	28	148
	incorrect	3	22	25
	Total	123	50	173

Table 11: Comparison of *TWS* and *Word Frequency* methods for the 173 applicable examples.

At the end of this process we got a set of 269 ambiguous Hebrew words. The average number of alternative translations per ambiguous word in this set is 5.8. For the purpose of evaluation, the translator has marked (a-priori) all the acceptable translations of each word (a selection of the TWS method will be regarded as correct if it is one of the "acceptable" translations). The average number of acceptable translations per ambiguous word in the set is 1.35.

Two measurements, *applicability* and *precision*, are used to evaluate the performance of the algorithm. The applicability denotes the proportion of cases for which the model performed a selection, i.e. those cases for which the significance criterion was satisfied. The precision denotes the proportion of cases for which the model performed a correct selection out of all the applicable cases. The performance of the TWS method is compared to that of a naive method that always selects the most frequent target word (the *Word Frequency* method).

The results of the experiment are summarized in tables 11 and 12[11]. The applicability of the Word Frequency method is 99.8% (the words of 3 examples were missing in the training corpus), while the applicability of the TWS method is 64.3%. The overall precision of the Word Frequency method is 66.9%. For the words that are covered by the TWS method, the word frequency method has a precision of 71.1%($\frac{123}{173}$), while the TWS method has a precision of 85.5%($\frac{148}{173}$)[12]. As can be seen in table 11, the TWS method agrees with almost all of the correct selections of the word frequency method, and corrects about half of its errors. These results give additional support for the usefulness of the TWS method, even for the noisy data provided by a low quality corpus, without any parsing or tagging.

According to the results reported above, the TWS method was inapplicable in 35.7%($\frac{96}{269}$) of the cases. Out of the 96 cases, in 26 cases the method was not applicable

[11]The parameters we used for the significance criterion of the TWS method were $\alpha = 0.3$ and $T = 0.38$. See (Dagan et al., 1991) for details.

[12]Dagan et al. (1991) reports applicability of 70% and precision of 92%. However, these results were achieved by the original TWS method, using a parsed corpus. In addition, the corpus used in that experiment was about 2.5 times larger and of much higher quality than the one used in the current experiment.

	Precision		Applicability	
TWS	85.5%	(148/173)	64.3%	(173/269)
Augmented TWS	83.6%	(179/214)	79.6%	(214/269)
Word Frequency	66.9%	(178/266)	98.8%	(266/269)

Table 12: Comparison between TWS, Augmented TWS and Word Frequency methods.

because the counters of the competing alternatives were relatively large but too close to each other. In those cases there is enough data in the corpus but the decision algorithm is not able to select a translation with high confidence. In 70 cases, the method was not applicable since all the alternative cooccurrence pairs did not occur in the corpus more than once (recall that we do not include in our data base pairs that occur only once). In the next subsection we show how the similarity based estimation method can augment the TWS method, and enable a selection in many of these cases.

4.4 Augmenting the TWS method with similarity based estimation

In the *Augmented TWS* method, the similarity based estimation method is invoked whenever none of the alternative pairs occur in the training corpus. After the expected frequencies of these pairs are estimated, the TWS method proceeds as if those frequencies were observed in the corpus itself. The only difference is that now we do not test for statistical significance, as we are using similarity based estimates of frequencies rather than statistical observations[13].

The performance of the *Augmented TWS* method compared to the *TWS* method is summarized in table 12. The augmented method resolved 41 additional examples out of the 70 examples that were handed to it, increasing the overall applicability by 15.3% (from 64.3% to 79.6%). The precision of the *Augmented TWS* method is 83.6% ($\frac{179}{214}$) vs. 85.5% of the TWS method alone. For the cases that were handed to it, the *Augmented TWS* method had an applicability of 58.6% ($\frac{41}{70}$) and a precision of 75.6%($\frac{31}{41}$). For these cases, the *Word Frequency* method had a precision of 61%($\frac{25}{41}$). It should be noted that using the Word Frequency method is equivalent to using *frequency based estimates* for unobserved pairs, as discussed in section 3.2. This is because higher word frequencies entail higher estimates for the corresponding pairs.

These results demonstrate how the similarity-based estimation method can aug-

[13]The resulting decision criteria is simply a threshold for $\ln(\frac{n_1}{n_2})$, which we have set to 0.22. n_1 and n_2 are the two highest estimates for alternative cooccurrence pairs.

ment an existing disambiguation method. The experiment shows that the applicability of the TWS method is significantly increased by the augmentation, with just a small reduction in its precision. The performance is improved also relative to a word-frequency based method, which, as discussed earlier, is representative for the expected results of frequency based smoothing methods when applied to unobserved cooccurrences. In this experiment, the similarity based estimation was tested on a rather small set of 70 examples. However, together with the experiment reported in subsection 3.6.2, using a data set of 300 examples, it demonstrates the utility of similarity based estimation.

5 Conclusions

Data sparseness is known to limit the applicability and/or precision of many corpus based techniques for natural language and speech processing. The goal of the research described in this paper was to develop a method which reduces the sparse data problem. The method follows the intuitively appealing idea that information about words, which is missing in the training data, can be inferred from information about similar words. Our evaluations show that this kind of analogy is useful for disambiguation, and suggest similarity based estimation as an enhancement for applications that use lexical cooccurrence data.

The similarity based approach suggests an alternative to the two other approaches for dealing with data sparseness:

- **An alternative to class based models**
  It has been traditionally assumed that semantic information about words should be generalized using word classes. In systems which rely on manual encoding of knowledge, this assumption seems necessary to cope with the high complexity of lexical relationships. However, it was never clearly shown that unrestricted language is indeed structured in accordance with this assumption. Moreover, the high variability in lexical cooccurrence data suggests that rather few generalizations can be performed on safe grounds.

  The similarity based approach differs from the class based approach both linguistically and computationally: From a linguistic point of view, the approach assumes minimal structure of lexical and semantic relationships. It avoids generalizations for classes of words, and demonstrates that these can be replaced by specific local analogies for each word. From a computational point of view, our approach does not require any form of clustering, but on the other hand raises other computational issues, some of them described in this paper. Future research is required for experimental comparison of the two approaches.

- **An alternative to frequency based estimates**

When estimating the probability of an unobserved word cooccurrence, frequency based estimates consider the frequencies of specific words. The similarity based method, like several class based methods, suggests to consider cooccurrence patterns of words when estimating their cooccurrence probability. The advantage of doing so is supported by our both evaluations, in which similarity based estimation performs better than frequency based estimation[14].

As a side effect of our evaluations, two additional results were achieved: first, we provide additional evidence, with a larger number of examples, for the effectiveness of the disambiguation method in (Dagan et al., 1991). Second, we have shown that both the similarity and estimation methods, as well as the disambiguation method, are effective when implemented using coccurrence pairs. This does not require any form of parsing or tagging of the corpus, thus making these methods very practical and easy to implement[15]. Yet, since the estimation method is applicable for any type of lexical relationship, its performance may be further improved if applied to a parsed or tagged corpus.

5.1 Further research

The similarity metric and the estimation method may be improved in several aspects. First, it is desirable to experiment with variations of the similarity metric (equation 6), such as discussed in section 3.3.3. Second, the procedure which averages the mutual information values of similar pairs (section 3.4) may be substituted with a dynamic weighting procedure, which considers a different number of similar pairs in each case, and assigns weights for these pairs according to the similarity values. Both the similarity metric and the averaging procedure were developed experimentally, based on the assumption that similar pairs have similar values of mutual information. It is desirable to have a probabilistic model that justifies this assumption (or an alternative one), and derives a similarity metric and a probabilistic estimation formula. As described in Section 2.3, two recent works (Essen and Steinbiss, 1992; Dagan et al., 1994) propose probabilistic models for estimating word cooccurrence probabilities, and report initial experimental results for word bigram language models. Further work is still needed to improve our understanding of the similarity-based approach and its potential for practical applications.

We encountered a certain problem in applying the similarity metric to cooccurrence pairs. Words which cooccur frequently as adjacent to each other, such as

[14]It should be noted that our method does take into account word frequencies, by the use of mutual information (see equation 5).

[15]Hindle, for example, attributes the failure of early work on automatic classification to the lack of a robust parser (Hindle, 1990). Many other methods also rely on a parsed corpus in identifying lexical relations (see section 2). However, our conclusion regarding the utility of cooccurrence pairs should be restricted for English, and may not hold for languages with freer word order.

'computer' and 'science', are found to be similar to each other by the similarity metric. This is because words that cooccur with the phrase 'computer science' appear in cooccurrence pairs with both words. This problem will not arise when using a parsed corpus, in which the two words are assigned different syntactic roles and therefore participate in different relationships. For cooccurrence pairs, it might be possible to use information on the distances between the two words in a pair (the distances for 'computer' and 'science' will have a consistent difference of one position), or to detect this undesired situation directly.

Beyond the scheme presented here, the similarity based approach can be extended in several ways. One possibility is to use the estimation method for reducing the number of distinct cooccurrences kept in the system's data base (such as an n-gram data base). As larger training corpora become available, many systems will not be able to store all the observed cooccurrences in their data base. It will then be necessary to devise criteria for deleting those cooccurrences whose probabilities can be best restored from the maintained data. As shown in the experiment of section 3.6, the similarity based estimation method can partially recover deleted probabilities, exploiting redundancy in cooccurrence data. Other extensions of the approach may use the similarity metric directly for disambiguation, along the lines suggested by Sadler (1989), without explicit estimation of unknown probabilities,

Acknowledgements

We would like to thank Alon Itai for help in initiating this research, and Ken Church for helpful comments on an earlier draft of this paper.

REFERENCES

David W. Aha, Dennis Kibler, and M. K. Albert. 1991. Instance-based learning algorithms. *Machine Learning*, 6:37–66.

James Allen. 1995. *Natural Language Understanding.* Benjamin/Cummings Publishing Company, Inc., second edition.

Eric Brill, David Magerman, Mitchell Marcus, and Beatrice Santorini. 1990. Deducing linguistic structure from the statistics of large corpora. In *DARPA Speech and Natural Language Workshop*, June.

P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. 1991. Word sense disambiguation using statistical methods. In *Proc. of the Annual Meeting of the ACL*, pages 264–270.

Peter Brown, Vincent Della Pietra, Peter deSouza, Jenifer Lai, and Robert Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4).

Kenneth W. Church and William A. Gale. 1991. A comparison of the enhanced Good-Turing and deleted estimation methods for estimating probabilities of English bigrams. *Computer Speech and Language*, 5:19–54.

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

Kenneth W. Church and Robert L. Mercer. 1993. Introduction to the special issue in computational linguistics using large corpora. *Computational Linguistics*, 19(1).

Ido Dagan and Alon Itai. 1991. A statistical filter for resolving pronoun references. In Y. A. Feldman and A. Bruckstein, editors, *Artificial Intelligence and Computer Vision*, pages 125–135. Elsevier Science Publishers B.V. (The Proc. of the 7th Israeli Sym. on Artificial Intelligence and Computer Vision, 1990).

Ido Dagan, Alon Itai, and Ulrike Schwall. 1991. Two languages are more informative than one. In *Proc. of the Annual Meeting of the ACL*, pages 130–137.

Ido Dagan, Shaul Marcus, and Shaul Markovitch. 1993. Contextual word similarity and estimation from sparse data. In *Proc. of the Annual Meeting of the ACL*, pages 164–171.

Ido Dagan, Fernando Pereira, and Lillian Lee. 1994. Similarity-based estimation of word cooccurrence probabilities. In *Proc. of the Annual Meeting of the ACL*, pages 272–278.

Ido Dagan. 1992. *Multilingual Statistical Approaches for Natural Language Disambiguation.* Ph.D. thesis, Computer Science Department, Technion - Israel Institute of Technology, Haifa, May. (in Hebrew).

Ute Essen and Volker Steinbiss. 1992. Coocurrence smoothing for stochastic language modeling. In *Proceedings of ICASSP*, volume I, pages 161–164. IEEE.

R. Fano. 1961. *Transmission of Information.* Cambridge, Mass., MIT Press.

William Gale, Kenneth Church, and David Yarowsky. 1992a. A method for disambiguating word senses in a large corpus. Technical Report Statistical Research Report, No. 104, AT&T Bell Laboratories.

William Gale, Kenneth Church, and David Yarowsky. 1992b. Work on statistcal methods for word sense disambiguation. In *Working Notes, AAAI Fall Symposium Series, Probabilistic Approaches to Natural Language*, pages 54–60.

I. J. Good. 1953. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40:237–264.

R. Grishman, L. Hirschman, and Ngo Thanh Nhan. 1986. Discovery procedures for sublanguage selectional patterns – initial experiments. *Computational Linguistics*, 12:205–214.

V. Gupta, M. Lennig, and P. Mermelstein. 1992. A language model for very large-vocabulary speech recognition. *Computer Speech and Language*, 6:331–344.

Zelig S. Harris. 1968. *Mathematical structures of language.* Wiley.

D. Hindle and M. Rooth. 1991. Structural ambiguity and lexical relations. In *Proc. of the Annual Meeting of the ACL*, pages 229–236.

D. Hindle. 1983. Deterministic parsing of syntactic non-fluencies. In *Proc. of the Annual Meeting of the ACL.*

D. Hindle. 1990. Noun classification from predicate-argument structures. In *Proc. of the Annual Meeting of the ACL*, pages 268–275.

L. Hirschman. 1986. Discovering sublanguage structures. In R. Grishman and R. Kittredge, editors, *Analyzing Language in Restricted Domains: Sublanguage Description and Processing*, pages 211–234. Lawrence Erlbaum Associates.

F. Jelinek and R. Mercer. 1985. Probability distribution estimation from sparse data. *IBM Technical Disclosure Bulletin*, 28:2591–2594.

Frederick Jelinek, Robert L. Mercer, and Salim Roukos. 1992. Principles of lexiacal language modeling for speech recognition. In Sadaoki Furui and M. Mohan Sondhi, editors, *Advances in Speech Signal Processing*, pages 651–699. Mercer Dekker, Inc.

Frederick Jelinek. 1990. Self-organized language modeling for speech recognition. In Alex Waibel and Kai-Fu Lee, editors, *Readings in Speech Recognition*, pages 450–506. Morgan Kaufmann Publishers, Inc., San Mateo, California.

Slava M. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, speech, and Signal Processing*, 35(3):400–401.

Raymond Lau, Ronald Rosenfeld, and Salim Roukos. 1993. Adaptive language modeling using the maximum entropy principle. In *ARPA.*

Yoelle Maarek and Frank Smadja. 1989. Full text indexing based on lexical relations – An application: Software libraries. In *Proc. of SIGIR.*

W.J.R. Martin, B.P.F. Al, and P.J.G. van Sterkenburg. 1983. On the processing of text corpus: from textual data to lexicographical information. In R.R.K. Hartman, editor, *Lexicography: Principles and Practice*, Applied Language Studies Series. Academic Press, London.

Fernando Pereira and Naftali Tishby. 1992. Distributional similarity, phase transitions and hierarchical clustering. In *Working Notes, AAAI Fall Symposium on Probabilistic Approaches to Natural Language*.

Fernando Pereira, Naftali Tishby, and Lillian Lee. 1993. Distributional clustering of English words. In *Proc. of the Annual Meeting of the ACL*.

Philip Resnik. 1992. Wordnet and distributional analysis: A class-based approach to lexical discovery. In *AAAI Workshop on Statistically-based Natural Language Processing Techniques*, July.

V. Sadler. 1989. *Working with analogical semantics: Disambiguation techniques in DLT*. Foris Publications.

Hinrich Schütze. 1993. Word space. In S. J. Hanson, J. D. Cowan, and C. L. Giles, editors, *Advances in Neural Information Processing Systems 5*, pages 895–902. Morgan Kaufman Publishers, San Mateo CA.

Frank Smadja and Katheleen McKeown. 1990. Automatically extracting and representing collocations for language generation. In *Proc. of the Annual Meeting of the ACL*.

Yorick Wilks. 1975. An intelligent analyzer and understander of english. *Communications of the ACM*, 18(5):264–274. reprinted in RNLP.