TECHNICAL REPORT 97-CSE-9

Unsupervised Text Mining

Ted Pedersen
Rebecca Bruce

Department of Computer Science and Engineering
Southern Methodist University
Dallas, TX 75275–0122 USA
{pedersen,rbruce}@seas.smu.edu

June 1997

Abstract

We describe the results of performing text mining on a challenging problem in natural language processing, word sense disambiguation. We compare two methods of unsupervised learning, Ward's minimum–variance clustering and the EM algorithm, that distinguish the meaning of an ambiguous word based only on features that can be automatically identified in text. This is a significant

advantage over most previous approaches which require a training sample where the meanings of ambiguous words have been manually disambiguated. The creation of sense tagged text sufficient to serve as a training sample is expensive and time consuming and is yet another example of the knowledge acquisition bottleneck. We present experimental results showing the application of each of these algorithms to the disambiguation of three nouns using five different feature sets. We find that these methods can distinguish two senses of *bill* with accuracy of up to 82 percent, three senses of *interest* with accuracy up to 69 percent, and three senses of *line* with accuracy up to 43 percent. These are improvments of 32, 36, and 10 percent over the lower bounds, respectively.

1 Introduction

A fundamental problem in any text mining or natural language processing application is the ambiguity of word meanings. For example, *bill* has a variety of senses – a piece of currency, a proposed law, the jaws of a bird, etc. In *The Senate bill is being voted on*, it is clear to a human reader that *bill* is used in the legislative sense. However, a text mining system searching for knowledge about bird anatomy might erroneously select this sentence unless *bill* is disambiguated. The ability to automatically annotate text with sense information can improve document classification (e.g., [33], [32]) and smooth the path for Web mining applications such as those described in [8].

Word sense disambiguation has commonly been cast as a problem in supervised learning (e.g., [2], [35], [36], [16], [3], [21], [22], [23]). However, these methods require text where ambiguous words have been manually tagged with sense information to train the learning algorithm. Such data exists only in very small quantities and is very expensive to create. Rather than assuming the availability of sense–tagged text, it seems more reasonable to develop unsupervised approaches that do not have such costly requirements.

We discuss two unsupervised learning algorithms, Ward's minimum–variance method [34] and the EM algorithm [6], that can distinguish among the known senses of an ambiguous word without the aid of disambiguated examples. The EM algorithm produces maximum likelihood estimates of the parameters of a parametric probabilistic model. Ward's method produces groupings of unlabeled observations that minimize the variance within clusters according to a measure of dissimilarity.

The rest of this paper is organized as follows. First, we present the five feature sets and three ambiguous words that form the basis of our experimental studies (Section 2). We provide introductions to Ward's minimum–variance method of clustering (Section 3) and the EM algorithm (Section 4). We present our experimental results (Section 5) and close with related work (Section 6) and a discussion of our results (Section 7).

2 Experimental Data

Experiments were conducted to distinguish the senses of three ambiguous nouns, *bill*, *interest*, and *line*, using five different feature sets. In order to evaluate the unsupervised learning algorithms we use sense–tagged text in these experiments. However, other than during evaluation these tags are disregarded.

2.1 Feature Sets

Raw text is devoid of consistent structure. It order to reduce text to a more structured form, we select certain features in the sentence that can be automatically identified and are imperfect indicators of the sense of the ambiguous word. Each sentence containing the ambiguous word is reduced to a vector of features (i.e., an observation) and the corpus of text under study is reduced to a matrix of observations.

2.1.1 Feature Sets A and B

Feature set A has been used in a variety of supervised learning experiments (e.g., [3], [22], [23]). A sentence with an ambiguous word is represented by a feature set composed of three types of contextual features: one morphological feature, four part–of–speech (POS) features, and three collocation features.

The morphological feature indicates if the ambiguous noun is plural or not. The POS features have one of 25 possible POS tags, derived from the first letter of the tags in the ACL/DCI Wall Street Journal corpus [19]. There are four features representing the part–of–speech of the two words immediately preceding and following the ambiguous word. The three binary collocation-specific features indicate the presence or

absence of a particular word in the same sentence as the ambiguous word. For *interest*, the collocations we seek are *in*, *percent* and *rate*. For *bill* the collocations are *auction*, *discount* and *treasury*.[1]

For *interest*, the number of possible combinations of these feature values is $25^4 \times 2^3 \times 2 \times 3 = 18,750,000$. For *bill*, there are $25^4 \times 2^3 \times 2 \times 2 = 9,375,000$ possible combinations. The last term in the two expressions above varies because it indicates the number of senses to be distinguished.

Feature set B is a simple modification of feature set A where the number of possible values for the POS tags is reduced from 25 to 5 (noun, verb, adjective, adverb, and other). The morphological and collocation features remain the same. While this modification reduces the information content of the feature set it also significantly reduces the dimensionality (i.e., the number of possible combinations of feature values) of the data.

For *interest* the number of possible combinations of feature values is $5^4 \times 2^3 \times 2 \times 3 = 30,000$ and for *bill* there are $5^4 \times 2^3 \times 2 \times 2 = 15,000$ possible values.

2.1.2 Feature Sets C, D, and E

It has been reported that human beings disambiguate word senses based on the words that occur two words to the left and two words to the right of the ambiguous word [5].

Feature sets C, D and E are based on this finding in that they consist only of four positional collocation features. The values of these features are the words that occur one and two positions to the right and left of the ambiguous word. Note that this is not a "bag of words" representation in that positional information is maintained. These feature sets differ from A and B in that the number of possible combinations of feature values varies with the sample size.

The possible values of the features in set C are the words that occur in each specified position with frequency greater than 20. For our experiments, the number of possible combinations of feature values ranged from 32,000 to 200,000.

The possible values of the features in set D consists of all nouns, verbs and adjectives that occur in each position with a frequency greater than 20. The number of possible values for these features ranged from 2,000 to 10,000 in our experiments.

The possible values of the features in set E are all words that occur within two positions of the ambiguous word. This feature set results in a very high dimensional representation. The number of possible combinations of the values of these features in our experiments was always in excess of 10^{10}. The dimensionality of this feature set exceeded the capability of our implementation of the EM algorithm.

2.2 Text

The *bill* and *interest* data consists of every sentence from the ACL/DCI WSJ corpus that contains either of those words. Each extracted sentence is tagged with a single sense defined in the Longman Dictionary of Contemporary English (LDOCE) [26]. This data is described in more detail in [4].

The *line* data comes from both the ACL/DCI WSJ corpus and the American Printing House for the Blind corpus. Each extracted sentence is tagged with a single sense of *line* defined in WordNet [20]. This data is described in more detail in [16].

In these experiments we distinguish among three senses of *interest* and *line* and between two senses of *bill*. Those senses are shown in Figure 1 along with an example and the number of instances of each sense in the corpus.

For each word, we create three test sets. Each test set is a random sample selected such that each sense of the ambiguous word occurs 200 times in that sample. Thus for *interest* and *line* there are three samples of 600 sentences each, and for *bill* there are three samples of 400 sentences each. This design establishes the

[1] Feature sets A and B are not used for the *line* data because it is not part–of–speech tagged.

word	sense	example	count
bill	prospective law	the immigration bill	930
	paper money	a five dollar bill	280
interest	ready to give attention	an interest in the case	350
	a share in a business	an interest in the venture	500
	money paid to borrow money	a high rate of interest	1250
line	telephone connection	the line is busy	430
	a product	a new line of cars	2220
	spoken or written text	the line about the President	400

Figure 1: Senses of *bill*, *interest*, and *line*

lower bound of disambiguation. That is, the accuracy that would result from using the majority classifier (i.e., assigning each word in the test set the most frequent sense in that test set). For *interest* and *line* the lower bound is 33 percent accuracy while for *bill* it is 50 percent.

3 Ward's minimum–variance method

In general, clustering methods define a distance measure between observations in a data set. Observations are grouped in the manner that minimizes the distance between the members of each group.

Our features represent the part–of–speech (POS) tags, morphological characteristics, or particular words in a sentence containing the ambiguous word. The values of these features are nominal and therefore do not have scale. For example, we could represent a POS feature in set B as a random variable using a mapping such as: noun=1, verb=2, adjective=3, adverb=4, and other= 5. The fact that adverb is represented by a higher number than noun is purely coincidental and does not indicate that adverbs are better or worse than nouns.

Thus, rather than using a distance measure such as Euclidean distance, we must employ Hamming distance, a measure of dissimilarity. Suppose we have N observations in a sample where each observation has n features. The Hamming distance is represented in an $N \times N$ dissimilarity matrix such that the value in cell (i, j) (where i represents the row number and j represents the column) is equal to the number of features in observations i and j that do not match.

Ward's minimum–variance method is an agglomerative clustering algorithm. All such clustering algorithms begin with N clusters, one for each observation in the sample. The two closest clusters are merged to form a new cluster that replaces them. Merging the two closest clusters continues until some specified number of clusters are obtained. In word–sense disambiguation the number of clusters corresponds to the number of senses we wish to distinguish.

In Ward's method, the internal variance of a cluster (i.e., inter-cluster variance) is the sum of squared distance between each observation in the cluster and the mean for that cluster (i.e., the average of the feature vectors in the cluster). At each step in Ward's method, a new cluster, C_{KL}, with the smallest possible inter-cluster variance, is created by merging the two clusters, C_K and C_L, that have the minimum intra-cluster variance. The intra-cluster variance between C_K and C_L is computed as follows:

$$V_{KL} = \frac{||\overline{x}_K - \overline{x}_L||^2}{\frac{1}{N_K} + \frac{1}{N_L}} \tag{1}$$

where $\overline{x}_K$ is the mean observation for cluster C_K and N_K is the number of observations in C_K. Since we

have nominal data, the observations referred to here are the observations (rows) of the dissimilarity matrix rather than the original matrix of feature values.

An assumption that is implicit in Ward's method is that the data comes from a mixture of normal distributions. While NLP data is typically not well characterized by a normal distribution (see, e.g. [38], [24]), our findings indicate that when the data is represented in terms of Hamming distances, it can be adequately characterized by the normal distribution. Also, the relative success of Ward's method indicates that the mixture of normals assumption holds. However, further experiments will be carried out to verify this assumption.

4 EM Algorithm

The expectation maximization algorithm [6], commonly known as the EM algorithm, estimates the values of model parameters when data is missing or unknown. In this work, the sense of the ambiguous word is represented by a feature whose values are missing from the data.

We also assume that we know the parametric form of the model that best represents the data. For these experiments we assume that the model form is the Naive Bayes [7]. This is a model in which all contextual features are conditionally independent given the value of the classification feature (i.e., the sense of the ambiguous word, represented as S). We make this assumption because a number of researchers have reported high accuracy when applying the Naive Bayes model to supervised word–sense disambiguation (e.g. [9], [16], [21],[23]).

The EM algorithm is an iterative estimation procedure in which the problem is recast to make use of complete data estimation techniques (i.e., techniques that can be used when there are no missing values). At the heart of the EM Algorithm lies the Q-function, defined in [6]. The Q-function is the expectation of the log-likelihood function for the complete data $D = (Y, S)$, where Y is the observed data and S is the missing sense value:

$$Q(\theta, \theta^i) \quad = \quad \int_S log[p(Y, S|\theta)]p(S|\theta^i, Y)dS \tag{2}$$

Here, θ^i is the current value of the maximum likelihood estimates of the model parameters and θ is the improved estimate that we are seeking; $p(Y, S|\theta)$ is the likelihood function augmented with the missing data S, and $p(S|\theta^i, Y)$ is the conditional probability of observing the missing data S given the current approximation of the model parameters and the observed data Y. Note that since the possible values of the sense variable S are discrete it is not necessary to integrate over S but rather simply to sum.

When approximating the maximum of the likelihood function, the EM algorithm starts from an initial estimate of θ^i and then replaces θ^i by the θ which maximizes $Q(\theta, \theta^i)$. This process is broken down into two steps: expectation (E) and maximization (M). The E-step finds the expected values of the sufficient statistics of the complete model using the current estimates of the model parameters. The M-step makes maximum likelihood estimates of the model parameters using the sufficient statistics from the E-step.

This process of estimating and maximizing will iterate producing a series of θ^i values until $||\theta - \theta^i||$ or $|Q(\theta, \theta^i) - Q(\theta^i, \theta^i)|$ is smaller than some pre–determined value ϵ.

There are a number of potential problems when using the EM algorithm. First, it is computationally expensive and convergence can be slow for problems with large numbers of parameters. Unfortunately there is little that can be done in this case other than formulating the problem so that fewer parameters need to be estimated. Second, the likelihood function may have local maxima. If the likelihood function is very irregular the EM algorithm may converge to a local maximum and not find the true maximum of the likelihood function. In this case the alternative is to use the more computationally expensive method of Gibbs Sampling [11] to find the complete distribution of either the likelihood or posterior probability function. We will evaluate Gibbs Sampling in future work.

	bill		interest		line	
Lower Bound	.5		.33		.33	
Feature Set	Ward	EM	Ward	EM	Ward	EM
A	.8150	.8208	**.5956**	**.6939**	na	na
B	.7958	.8075	**.4828**	**.5783**	na	na
C	**.6933**	**.5792**	.4989	.5211	.4000	.4128
D	**.7866**	**.6417**	**.5111**	**.4377**	.3856	.4083
E	.6766	na	.4772	na	.4261	na

Figure 2: Average Accuracy over Three Samples

5 Experimental Results

In this section, we present the results of all experiments. There are 4 parameters defining each experiment: the clustering method (EM or Ward's minimum-variance), the ambiguous word (*interest*, *bill*, or *line*), the feature set (A, B, C, D, or E), and the test set (sample 1, sample 2, or sample 3). In total, 90 different experiments are defined by these parameters of which 69 were performed; 21 experiments are eliminated because feature sets A and B can't be evaluated for *line* and the EM algorithm can't be run on feature set E.

In section 5.1 we present a summary of our findings by looking at the average performance of the clustering method over all three test sets for each ambiguous word. Averaging across test sets provides a better point estimate of performance. We summarize performance in terms of disambiguation accuracy and identify those tests where the difference in performance between clustering methods is statistically significant.

In these experiments, disambiguation accuracy is measured by comparing the word senses assigned via clustering to those assigned by a human judge; mismatches are considered to be errors on the part of the clustering algorithm. Clusters are automatically mapped to word senses in a manner that maximizes the agreement with the human judge.

In section 5.2, we analyze classifier performance by looking at the pattern of agreement and disagreement between between each clustering algorithm and the human judge. We analyze the sense distinctions made for the *interest* data using feature sets A and D. This analysis provides a more detailed look at the differences in the behavior of the two clustering algorithms.

5.1 Average Accuracy

Figure 2 shows the average disambiguation accuracy for each combination of feature set and learning algorithm. Those cases where the difference in the average accuracy between Ward's method and the EM algorithm for a given feature set is statistically significant are indicated in bold face. Statistical significance is assigned based on a test for marginal homogeneity in a 2x2 contingency table where the column space defines the number of test instances correctly and incorrectly classified, and the row space specifies the clustering method[2].

When using either feature set A or B and the *bill* data, there is no clear advantage for either Ward's method or the EM algorithm. Both achieve their maximum improvement over the lower bound (approx.

[2]The test for marginal homogeneity in a 2 dimensional contingency table is equivalent to a test for independence, i.e., is the classification performance independent of the method used.

		Discovered				Discovered			
		money	share	attention		money	share	attention	
	money	123	58	19	200	88	7	105	200
Actual	share	0	61	139	200	0	44	156	200
	attention	0	27	173	200	0	25	175	200
		123	146	331	600	88	76	436	600

Figure 3: interest - Ward - Feature Set A (left) and D (right)

30 percent) using feature set A. However, Ward's method is more accurate than the EM algorithm with feature sets C and D (69 vs. 58 percent and 79 vs. 64 percent, respectively).

The average accuracy for *interest* shows a clear advantage for the EM algorithm with feature set A. The average accuracy of 69 percent exceeds the EM algorithm by 10 percent and more than doubles the lower bound of 33 percent. However, as with *bill*, feature set D performs better with Ward's method than the EM algorithm (51 vs 44 percent).

The average accuracy for *line* is somewhat disappointing. The improvement over the lower bound is at most 10 percent and there is no clear advantage associated with any particular combination of learning method and feature set. The complexity of disambiguating *line* is discussed in [16]. They point out that the *text* sense of *line* is very difficult to distinguish because it is not tightly defined and it is not associated with a particular topic, a line of text can be about anything. We present evidence of the difficulty of disambiguating *interest* in the section below.

5.2 Analysis of interest data

Figures 3 and 4 show the confusion matrices associated with the disambiguation of *interest* using feature sets A and D. A confusion matrix shows the number of cases where raters (i.e., the persons or methods assigning classifications) agree along the main diagonal; the off-diagonal cells correspond to instances of disagreement. In this study, each confusion matrix compares the classifications "discovered" by an unsupervised learning algorithm to the "actual" classifications assigned by a human judge.

In general, these matrices reveal that the *money* sense of *interest* is easier to distinguish than either *share* or *attention*. Ward's method has more difficulty in distinguishing between *share* and *attention* than does the EM algorithm, although both clustering methods appear to strongly favor the *attention* sense over all other senses.

It seems reasonable that the *attention* and *share* senses of *interest* would be more difficult to distinguish than *money* and *attention* or *money* and *share*. Terminology such as "an interest in the product" can just as easily refer to being a shareholder as simply having curiosity.

Ward's method exhibits interesting behavior for the *money* sense. It never incorrectly classifies an ambiguous instance of *interest* as *money*. However, instances whose actual sense is *money* are more often misclassified by Ward's method than the EM algorithm.

The performance of both clustering algorithms is clearly worse on feature D than on feature set A. The degradation in performance on feature set D is more sever for the EM algorithm than it is for Ward's method. Although both methods lose some of their ability to distinguish the money sense of *interest*, this lose is more pronounced for the EM algorithm and it is accompanied by a strong bias for the "attention" sense of *interest* most of which are mis–tagged.

		Discovered				Discovered			
		money	share	attention		money	share	attention	
	money	170	11	19	200	25	66	109	200
Actual	share	19	87	94	200	13	104	83	200
	attention	6	35	159	200	11	55	134	200
		195	133	272	600	49	225	326	600

Figure 4: interest - EM - Feature Set A (left) and D (right)

6 Related Work

Bootstrapping approaches to word sense disambiguation require a small amount of disambiguated text in order to initialize the unsupervised learning algorithm. An early example of such an approach is described in [12]. A supervised learning algorithm is trained with a small amount of manually sense–tagged text and applied to a held out test set. Those examples in the test set that are most confidently disambiguated are added to the training sample.

A more recent bootstrapping approach is described in [37]. This algorithm requires a small number of training examples to serve as a seed. This approach relies upon the identification of collocations that uniquely distinguish between senses. To take an example from our data, the collocations *telephone line* and *production line* make such a distinction.

Clustering has previously been applied in natural language processing as an exploratory method for inducing syntactic or semantically related groupings of words (e.g., [30], [14], [29], [25], [27]).

An early application of clustering to word–sense disambiguation is described in [31]. There words are represented in terms of the co-occurrence statistics of four letter sequences. This representation uses 97 features to characterize a word, where each feature is a linear combination of letter four-grams formulated by a singular value decomposition of a 5000 by 5000 matrix of letter four-gram co-occurrence frequencies. The weight associated with each feature reflects all usages of the word in the data. A context vector is formed for each occurrence of an ambiguous word by summing the vectors of the contextual words (the number of contextual words considered in the sum is unspecified). The set of context vectors for the word to be disambiguated are then clustered, and the clusters are manually sense–tagged.

The features used in [31] are complex and difficult to interpret and it isn't clear that this complexity is required. [37] compares his method to [31] and shows that for four words the former performs significantly better in distinguishing between two senses.

Other clustering approaches to word–sense disambiguation have been based on measures of *semantic distance* defined with respect to a semantic network such as WordNet. Measures of semantic distance are based on the path length between concepts in a network and are used to group semantically similar concepts (e.g. [18]). [28] provides an information theoretic definition of semantic distance based on WordNet.

The only previous application of the EM algorithm to word–sense disambiguation is described in [10]. There the EM algorithm is used as part of a supervised learning algorithm to distinguish city names from people's names. A narrow window of context, one or two words to either side, was found to perform better than wider windows. The results presented are preliminary but show an accuracy percentage in the mid-nineties when applied to *Dixon*, a name found to be quite ambiguous.

It should be noted that the EM algorithm relates to a large body of work in speech processing. The forward–backward (also known as Baum-Welch) algorithm [1] is a specialized form of the EM algorithm that assumes the underlying parametric model is a hidden Markov model. The forward–backward algorithm has been used extensively in speech recognition (e.g. [17], [15]), [13]).

7 Conclusions

These experiments provide evidence that text can be disambiguated using unsupervised techniques. We also find that the relative success of Ward's minimum variance method as compared to the EM algorithm varies with the feature set. The EM algorithm performs consistently better than Ward's (although the difference is not always statistically significant) when applied to feature set A. This is also the feature set that provides the best overall disambiguation results, when applicable. Ward's method performs consistently better with feature set D. The difference in the formulation of these two methods is clear: the EM algorithm is designed to maximize the likelihood of the sense assignments according to a probabilistic model, while Ward's is designed to minimize the variance of clusters according to a measure of dissimilarity. It is not yet clear which features of a data set make it more appropriate for one method of unsupervised learning over another. These experiments indicate that the discrimination power of the feature set may be a factor.

References

[1] L. Baum. An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process. In O. Shisha, editor, *Inequalities*, volume 3, pages 1–8. Academic Press, New York, NY, 1972.

[2] E. Black. An experiment in computational discrimination of English word senses. *IBM Journal of Research and Development*, 32(2):185–194, 1988.

[3] R. Bruce and J. Wiebe. Word-sense disambiguation using decomposable models. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 139–146, 1994.

[4] R. Bruce, J. Wiebe, and T. Pedersen. The measure of a model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 101–112, 1996.

[5] Y. Choueka and S. Lusignan. Disambiguation by short contexts. *Computers and the Humanities*, 19:147–157, 1985.

[6] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38, 1977.

[7] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, NY, 1973.

[8] O. Etzioni. The World–Wide Web: Quagmire or gold mine? *Communications of the ACM*, 39(11):65–68, November 1996.

[9] W. Gale, K. Church, and D. Yarowsky. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26:415–439, 1992.

[10] W. Gale, K. Church, and D. Yarowsky. Discrimination decisions for 100,000 dimensional spaces. *Journal of Operations Research*, 55:323–344, 1995.

[11] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[12] M. Hearst. Noun homograph disambiguation using local context in large text corpora. In *Proceedings of the 7th Annual Conference of the UW Centre for the New OED and Text Research: Using Corpora*, Oxford, 1991.

[13] F. Jelinek. Self–organized language modeling for speech recognition. In Waibel and Lee, editors, *Readings in Speech Recognition*. Morgan Kaufmann, San Mateo, CA, 1990.

[14] G Kiss. Grammatical word classes: A learning process and its simulation. *Psychology of Learning and Motivation*, 7:1–41, 1973.

[15] J. Kupiec. Robust part-of-speech tagging using a hidden Markov model. *Computer Speech and Language*, 6:225–243, 1992.

[16] C. Leacock, G. Towell, and E. Voorhees. Corpus-based statistical sense resolution. In *Proceedings of the ARPA Workshop on Human Language Technology*, 1993.

[17] S. Levinson, L. Rabiner, and M. Sondhi. An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition. *Bell System Technical Journal*, 62:1035–1074, 1983.

[18] X. Li, S. Szpakowicz, and S. Matwin. A WordNet-based algorithm for word sense disambiguation. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Montreal, August 1995.

[19] M. Marcus, B. Santorini, and M. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

[20] G. Miller. WordNet: A lexical database. *Communications of the ACM*, 38(11):39–41, November 1995.

[21] R. Mooney. Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1996.

[22] H.T. Ng and H.B. Lee. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Annual Meeting of the Society for Computational Linguistics*, pages 40–47, 1996.

[23] T. Pedersen, R. Bruce, and J. Wiebe. Sequential model selection for word sense disambiguation. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, Washington, DC, 1997.

[24] T. Pedersen, M. Kayaalp, and R. Bruce. Significant lexical relationships. In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 455–460, 1996.

[25] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of English words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Columbus, OH, 1993.

[26] P. Procter, editor. *Longman Dictionary of Contemporary English*. Longman Group Ltd., Essex, UK, 1978.

[27] P. Resnik. Disambiguating noun groupings with respect to WordNet senses. In *Proceedings of the Third Workshop on Very Large Corpora*, MIT, June 1995.

[28] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Montreal, August 1995.

[29] H. Ritter and T. Kohonen. Self-organizing semantic maps. *Biological Cybernetics*, 62:241–254, 1989.

[30] A. Rosenfeld, H. Huang, and V. Schneider. An application of cluster detection to text and picture processing. *IEEE Transactions on Information Theory*, 15:672–681, 1969.

[31] H. Schütze. Dimensions of meaning. In *Proceedings of Supercomputing '92*, pages 787–796, Minneapolis, MN, 1992.

[32] H. Schütze and J. Pedersen. Information retrieval based on word senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, NV, 1995.

[33] E. Voorhees. Using WordNet to disambiguate word senses for text retrieval. In *Proceedings of SIGIR '93*, pages 171–180, Pittsburgh, PA, 1993.

[34] J. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

[35] D. Yarowsky. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 454–460, Nantes, France, July 1992.

[36] D. Yarowsky. One sense per collocation. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 266–271, 1993.

[37] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, MA, 1995.

[38] G. Zipf. *The Psycho-Biology of Language*. Houghton Mifflin, Boston, MA, 1935.