Knowledge Lean Word–Sense Disambiguation*

Ted Pedersen
Department of Computer Science and Engineering
Southern Methodist University
Dallas, TX 75275–0112
pedersen@seas.smu.edu

Rebecca Bruce
Department of Computer Science
University of North Carolina at Asheville
Asheville, NC 28804
bruce@cs.unca.edu

Abstract

We present a corpus–based approach to word–sense disambiguation that only requires information that can be automatically extracted from untagged text. We use unsupervised techniques to estimate the parameters of a model describing the conditional distribution of the sense group given the known contextual features. Both the EM algorithm and Gibbs Sampling are evaluated to determine which is most appropriate for our data. We compare their disambiguation accuracy in an experiment with thirteen different words and three feature sets. Gibbs Sampling results in small but consistent improvement in disambiguation accuracy over the EM algorithm.

Introduction

Resolving the ambiguity of words is a central problem in natural language processing. A wide range of approaches have been applied to word–sense disambiguation. However, most require manually crafted knowledge such as annotated text, machine readable dictionaries or thesari, semantic networks, or aligned bilingual corpora. We present a corpus–based approach to disambiguation that relies strictly on knowledge that is automatically identifiable within the text being processed. This avoids dependence on external knowledge sources and is therefore a *knowledge lean* approach.

We are given N sentences that each contain a particular ambiguous word. Each is converted into a feature vector $(F_1, F_2, \ldots, F_n, S)$ where $(F_1, \ldots, F_n)$ represent selected properties of the context in which the ambiguous word occurs and S represents the sense of the ambiguous word. Our objective is to divide these N instances of an ambiguous word into a specified number of sense groups. These sense groups must be mapped to sense tags in order to evaluate system performance. We use the mapping that results in the highest classification accuracy.

There are a wide range of unsupervised learning techniques that could be applied to this problem. We use a parametric model to assign a sense group to each ambiguous word. In each case, we assign the most probable group given the context as defined by the Naive Bayes model where the parameter estimates are formulated via unsupervised techniques. The advantage of this approach is two-fold: (1) there is a large body of evidence recommending the use of the Naive Bayes model in word–sense disambiguation (e.g., (Leacock, Towell, & Voorhees 1993), (Mooney 1996), (Ng 1997)) and (2) unsupervised techniques for parameter estimation, once developed, could be easily applied to other parametric forms in the class of decomposable models.

We employ the Expectation Maximization (EM) algorithm (Dempster, Laird, & Rubin 1977) and Gibbs Sampling (Geman & Geman 1984) to estimate model parameters from untagged data. Both are well known and widely used iterative algorithms for estimating model parameters in the presence of missing data; in our case, the missing data are the senses of the ambiguous words. The EM algorithm formulates a maximum likelihood estimate of each model parameter, while Gibbs Sampling is a simulation technique for estimating the mode of the posterior distribution of each model parameter. When the likelihood function is not well approximated by a normal distribution, simulation techniques often provide better estimates of the model parameters. Our data, as is typical of Natural Language Processing data, is sparse and skewed and therefore not necessarily well characterized by large sample approximations. In this study, we compare maximum likelihood estimates to those produced using a more expensive simulation technique.

First, we describe the application of the Naive Bayes model to word–sense disambiguation. The following sections introduce the EM algorithm and Gibbs Sampling, respectively. We present the results of an extensive evaluation of three different feature sets applied to each of thirteen ambiguous words. We close with a discussion of related work and our future directions.

Naive Bayes Model

In the Naive Bayes model, all features are assumed to be conditionally independent given the value of the

classification variable. When applied to word–sense disambiguation, the model specifies that all contextual features are conditionally independent given the sense of the ambiguous word. The joint probability of observing a certain combination of contextual features with a particular sense is expressed as:

$$p(F_1, F_2, \ldots, F_n, S) = p(S) \prod_{i=1}^{n} p(F_i|S) \qquad (1)$$

The parameters of this model are $p(S)$ and $p(F_i|S)$. The sufficient statistics, i.e., the summaries of the data needed for parameter estimation, are the frequency counts of events described by the interdependent variables (F_i, S). Given these marginal counts, parameter estimates follow directly. However, when the sense tags are missing, direct estimates are not possible; instead we use the EM algorithm and Gibbs Sampling to impute a sense group for the missing data and estimate the parameters.

EM Algorithm

There are two steps in the EM algorithm, expectation (E–step) and maximization (M–step). The E–step calculates the expected values of the sufficient statistics given the current parameter estimates. The M-step makes maximum likelihood estimates of the parameters given the imputed values of the sufficient statistics. These steps alternate until the parameter estimates in iteration $k-1$ and k differ by less than ϵ.

The EM algorithm for the exponential family of probabilistic models is introduced in (Dempster, Laird, & Rubin 1977). The Naive Bayes model is a decomposable model which is a member of the exponential family with special properties that simplify the formulation of the E-step (Lauritzen 1995).

The EM algorithm for Naive Bayes proceeds as follows:

1. randomly initialize $p(F_i|S)$, set $k = 1$
2. E–step: $count(F_i, S) = p(S|F_i) \times count(F_i)$
3. M–step: re–estimate $p(F_i|S) = \frac{count(F_i,S)}{count(S)}$
4. $k = k + 1$
5. go to step 2 if parameter estimates from iteration k and $k-1$ differ by more than ϵ.

Gibbs Sampling

Gibbs Sampling is a Markov Chain method of generating random samples from a distribution when sampling directly from that distribution is difficult. We use Gibbs Sampling to impute the missing values for S and then sample values for the parameters.

Gibbs Sampling is often cast as a stochastic version of the EM algorithm (e.g., (Meng & van Dyk 1997)). However, in general Gibbs Sampling is applicable to a wider class of problems than the EM algorithm.

A Gibbs Sampler generates chains of values for the missing senses S and the parameters $p(F_i|S)$ via iterative sampling. These chains will eventually converge to a stationary distribution. The early iterations of the sampler produce values that vary quite a bit. It is suggested that some portion of the early iterations be discarded. This process is commonly known as a "burn–in". We use a 500 iteration burn–in and monitor the following 1000 iterations for convergence using the measure proposed in (Geweke 1992). If the chains have not converged, then additional iterations are performed until they do. Below we show the general procedure for Gibbs Sampling with the Naive Bayes model. *burn_in* represents the number of initial iterations that are discarded and *chain_size* is the number of iterations that are monitored.

1. randomly initialize $p(F_i|S)$, set $k = 1$
2. sample value for S from $p(S|F_1, \ldots, F_n) = \frac{p(S)\prod_i^n p(F_i|S)}{p(F1, F2, \ldots, F_n)}$
3. sample from parameters $p(F_i|S)$
4. $k = k + 1$
5. if $k <$ *chain_size* goto 2
6. does chain from (*burn_in* to *chain_size*) converge?
7. if not, increase *chain_size* and go to step 2

Prior knowledge can be conveniently incorporated using the conjugate prior for a multinomial distribution, a Dirichlet prior. The resulting posterior Dirichlet distribution is the distribution sampled from in steps 2 and 3. However, in these experiments, we do not assume any prior knowledge and therefore use uninformative priors.

Methodology

A series of experiments were conducted to disambiguate all occurrences of thirteen different words. Three different feature sets were defined for each word and used to formulate a Naive Bayes model describing the distribution of sense groups of that word. The parameters of each model were estimated using both the EM algorithm and Gibbs Sampling. In total, this amounts to 78 different disambiguation experiments. In addition, each experiment was repeated 25 times in order to measure the variance introduced by randomly selecting the initial parameter estimates. The disambiguation accuracy figures reported for these experiments measure how well the automatically defined sense groups map to the sense groups established by a human judge.

Data

The words used in these experiments and their sense distributions, as determined by a human judge, are shown in Figures 1, 2, and 3. *Total count* is the number of occurrences of each word. Each word was limited

chief: (total count: 1048)	
highest in rank:	86%
most important; main:	14%
common: (total count: 1060)	
as in the phrase 'common stock':	84%
belonging to or shared by 2 or more:	8%
happening often; usual:	8%
last: (total count: 3004)	
on the occasion nearest in the past:	94%
after all others:	6%
public: (total count: 715)	
concerning people in general:	68%
concerning the government and people:	19%
not secret or private:	13%

Figure 1: Adjective Senses

bill: (total count: 1341)	
a proposed law under consideration:	68%
a piece of paper money or treasury bill:	22%
a list of things bought and their price:	10%
concern: (total count: 1235)	
a business; firm:	64%
worry; anxiety:	36%
drug: (total count: 1127)	
a medicine; used to make medicine:	57%
a habit-forming substance:	43%
interest: (total count: 2113)	
money paid for the use of money:	59%
a share in a company or business:	24%
readiness to give attention:	17%
line: (total count: 1149)	
a wire connecting telephones:	37%
a cord; cable:	32%
an orderly series:	30%

Figure 2: Noun Senses

to the 2 or 3 most frequent senses. The frequency–based features employed here do not lend themselves to distinguishing among very small minority senses. In addition, the *line* data was reduced from 6 to 3 senses despite having a fairly uniform distribution. Initially this was done to maintain a similar total count and number of senses with the other words. However, preliminary experiments with 6 senses show that accuracy degrades considerably, to approximately 25 to 30 percent, depending on the feature set. This indicates that different features may be needed to accommodate larger numbers of senses.

The *line* data (Leacock, Towell, & Voorhees 1993) is taken from the ACL/DCI Wall Street Journal corpus and the American Printing House for the Blind corpus and tagged with WordNet senses. The remaining twelve words (Bruce, Wiebe, & Pedersen 1996) were

agree: (total count: 1109)	
to concede after disagreement:	74%
to share the same opinion:	26%
close: (total count: 1354)	
to (cause to) end:	77%
to (cause to) stop operation:	23%
help: (total count: 1267)	
to enhance - inanimate object:	78%
to assist - human object:	22%
include: (total count: 1526)	
to contain in addition to other parts:	91%
to be a part of - human subject:	9%

Figure 3: Verb Senses

taken from the ACL/DCI Wall Street Journal corpus and tagged with senses from the Longman Dictionary of Contemporary English.[1]

Feature Sets

We defined three different feature sets for use in these experiments. Our objective in doing so is two-fold: (1) to study the impact of the dimensionality of the event space on unsupervised parameter estimation, and (2) to study the informativeness of different feature types in word–sense disambiguation. Our feature sets are composed of various combinations of the following five types of features.

Morphology The feature M represents the morphology of the ambiguous word. For nouns, M is binary indicating singular or plural. For verbs, the value of M indicates the tense of the verb and can have up to seven possible values.[2] This feature is not used for adjectives.

Part–of–Speech The features PL_i and PR_i represent the part–of–speech (POS) of the word i positions to the left or right, respectively, of the ambiguous word. In these experiments, $i = 1$ or 2. Each POS feature can have one of five possible values: noun, verb, adjective, adverb or other. These tags were assigned automatically using the Unix command *style -P*.

Co–occurrences The features C_i are binary variables representing whether the i^{th} most frequent content word in all sentences containing the ambiguous word occurs anywhere in the sentence being processed. In these experiments, $i = 1$, 2 and 3.

Unrestricted Collocations The features UL_i and UR_i indicate the word occurring in the position i places to the left or right, respectively, of the ambiguous word.

[1] In these experiments, sense tags are used only in the evaluation of the sense groups found by the unsupervised learning procedures. If sense–tagged text were not available, the evaluation process would require manually mapping the sense groups to sense tags.

[2] All morphologically equivalent verb tenses were grouped as one; ambiguous morphology was not addressed.

event count	Full Joint			Naive Bayes		
	A	B	C	A	B	C
0	98.7	99.9	99.9	6.9	22.5	33.3
1–5	1.1	0.1	0.1	8.0	25.7	5.2
6–10	0.1	0.0	0.0	4.6	11.6	3.0
11–100	0.1	0.0	0.0	33.3	31.7	31.1
100+	0.0	0.0	0.0	47.1	8.4	27.4

Figure 4: Event Distribution for Noun *interest*

In these experiments $i = 1$ or 2. All features of this form have twenty-one possible values. Nineteen correspond to the most frequent words that occur in that fixed position in all sentences that contain the particular ambiguous word.[3] There is also a value (*none*) that indicates when the position i to the left or right is occupied by a word that is not among the nineteen most frequent, and a value (*null*) indicating that the position i to the left or right falls outside of the sentence boundary.

Content Collocations The features CL_1 and CR_1 indicate the content word occurring in the position 1 place to the left or right, respectively, of the ambiguous word. The values of these features correspond to the nineteen most frequent content words in that position plus *none* and *null*.

The features described above are defined over a small contextual window (local–context) and are selected to produce low dimensional event spaces. Local–context features have been used successfully in a variety of supervised approaches to disambiguation (e.g., (Bruce & Wiebe 1994), (Ng & Lee 1996)).

Feature Sets A, B and C The 3 feature sets used in these experiments are designated A, B and C pand are formulated as shown below. The particular feature combinations chosen were found to yield reasonable results in a preliminary evaluation.

- A: $M, PL_2, PL_1, PR_1, PR_2, C_1, C_2, C_3$
  Joint Events: 10,000 – 105,000
  Marginal Events: 78 – 99
- B: $M, UL_2, UL_1, UR_1, UR_2$
  Joint Events: 388,962 – 4,084,101
  Marginal Events: 168 – 273
- C: $M, PL_2, PL_1, PR_1, PR_2, CL_1, CR_1$
  Joint Events: 551,250 – 5,788,125
  Marginal Events: 160 – 207

"Joint Events" shows the range of the number of possible combinations of feature values in the full joint distribution of each feature set. We contrast this with "Marginal Events", the range of the possible combinations of feature values in the marginal distributions

[3] Nineteen distinct word forms were recognized to control the dimensionality of the feature set while still allowing the recognition of relevant correlations. This value was arrived at empirically; other values considered were 5, 11, and 31.

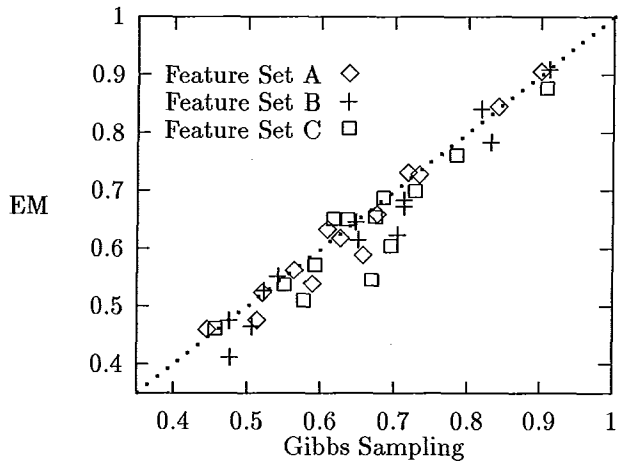

Figure 6: Accuracy of EM versus Gibbs

of the Naive Bayes model. Figure 4 shows an example of how the event count distribution of *interest* is smoothed by reducing the number of possible events though the use of the Naive Bayes model.

Discussion of Results

Figure 5 shows the average accuracy and standard deviation of disambiguation over 25 random trials for each combination of word, feature set and learning algorithm. Also included is the percentage of each sample that is composed of the majority sense. Figure 6 shows the correlation between the accuracy of disambiguation when using the EM algorithm versus Gibbs Sampling for all combinations of words and feature sets. Points that fall on or near the line $x = y$ are associated with words that were disambiguated with similar accuracy by both methods.

Method There are only a few cases where the use of Gibbs Sampling resulted in significantly more accurate disambiguation than the EM algorithm; this was judged by a two–tailed t-test with $p = .01$. The significant differences are shown in bold face. While the number of significant differences is small, Figure 6 shows a consistent increase in the accuracy of the Gibbs Sampler relative to the EM algorithm.

The EM algorithm found much the same parameter estimates as Gibbs Sampling. This is somewhat surprising given that the EM algorithm can converge to local maxima when the distribution of the likelihood function is not well approximated by the normal distribution. However, in our experiments the EM algorithm often converged quite quickly, usually within 20 iterations, to a global maximum. These results suggest that a combination of the EM algorithm and Gibbs Sampling might be appropriate. (Meng & van Dyk 1997) propose that the Gibbs Sampler start at the point the

	Maj.	Feature Set A		Feature Set B		Feature Set C	
		Gibbs	EM	Gibbs	EM	Gibbs	EM
chief	.861	.719±.01	.729±.06	.648±.00	.646±.01	.728±.04	.697±.06
common	.842	.522±.00	.521±.00	.507±.07	.464±.06	**.670±.11**	.543±.09
last	.940	.900±.00	.903±.00	.912±.00	.909±.00	.908±.00	.874±.07
public	.683	**.514±.00**	.473±.03	**.478±.04**	.411±.03	**.578±.00**	.507±.03
adjectives	.832	.663	.657	.636	.608	**.721**	.655
bill	.681	.590±.04	.537±.05	.705±.10	.624±.08	.592±.04	.569±.04
concern	.638	.842±.00	.842±.00	.819±.01	.840±.02	.785±.01	.758±.09
drug	.567	.676±.00	.658±.03	.543±.04	.551±.05	.674±.06	.652±.04
interest	.593	.627±.08	.616±.06	.652±.04	.615±.05	.617±.05	.649±.09
line	.373	.446±.02	.457±.01	.477±.03	.474±.03	.457±.01	.458±.01
nouns	.570	.636	.622	.639	.621	.625	.617
agree	.740	.609±.07	.631±.08	.714±.14	.683±.14	.685±.14	.685±.14
close	.771	.564±.09	.560±.08	.714±.05	.672±.06	.636±.05	.648±.05
help	.780	**.658±.04**	.586±.05	.524±.00	.526±.00	**.696±.05**	.602±.03
include	.910	.734±.08	.725±.02	**.833±.03**	.783±.07	.551±.06	.535±.00
verbs	.800	.641	.626	.696	.666	.632	.618
overall	.734	.646	.634	.657	.631	.659	.629

Figure 5: Experimental Results - accuracy ± standard deviation

EM algorithm converges rather than being randomly initialized. If the EM algorithm has found a local maximum then the Gibbs Sampler would be able to escape it and find the global maximum. However, if the EM algorithm has already found the global maximum then the Gibbs Sampler will converge quickly and confirm this result.

Feature Set The accuracy of disambiguation for nouns is fairly consistent across the feature sets. However, there are exceptions. The accuracy achieved for *bill* is much higher with feature set B than with A or C. The accuracy for *drug*, on the other hand, is much lower with feature set B than with A or C. This variation across feature sets may indicate that certain features are more or less appropriate for certain words.

The accuracy for verbs was highest with feature set B although *help* is a glaring exception. Feature set B is made up of local–context collocations.

The highest average accuracy achieved for adjectives occurs when Gibbs Sampling is used in combination with feature set C. This is a high dimensional feature set, additionally, the sense distributions of the adjectives are the most skewed. Under these circumstances, it seems unlikely that the EM algorithm would reliably find a global maximum, and, indeed, it appears that the EM algorithm found local maxima when processing *common* and *public*.

While frequency–based features, such as those used in this work, reduce sparsity, they are less likely to be useful in distinguishing among minority senses. Indeed, the more skewed the distribution of senses in the data sample, the more likely it is that frequency–based features will be indicative of only the majority sense.

Related Work

There is an abundance of literature on word–sense disambiguation. Our knowledge–lean approach differs from most in that it does not require any knowledge resources beyond raw text. Corpus–based approaches often use supervised learning algorithms with sense–tagged text (e.g., (Leacock, Towell, & Voorhees 1993), (Bruce & Wiebe 1994), (Mooney 1996)) or multi-lingual parallel corpora (e.g., (Gale, Church, & Yarowsky 1992)).

An approach that significantly reduces the amount of sense–tagged data required is described in (Yarowsky 1995). Yarowsky suggests a variety of options for automatically seeding a supervised disambiguation algorithm; one is to identify collocations that uniquely distinguish between senses. Yarowsky achieves an accuracy of more than 90% when disambiguating between two senses for twelve different words. This result demonstrates the effectiveness of a small number of representative collocations as seeds in an iterative bootstrapping approach.

A comparison of the EM algorithm and two agglomerative clustering algorithms as applied to unsupervised word–sense disambiguation is discussed in (Pedersen & Bruce 1997). Using the same data used in this study, (Pedersen & Bruce 1997) found that McQuitty's agglomerative algorithm is significantly more accurate for adjectives and verbs while the EM algorithm is significantly more accurate for nouns. These results indicate that McQuitty's analysis, which is based on counts of dissimilar features, is most appropriate for highly skewed data sets. The performance of Gibbs Sampling in the current study also falls short of that

of McQuitty's for adjectives and verbs which supports the previous conclusion.

The EM algorithm is used with a Naive Bayes classifier in (Gale, Church, & Yarowsky 1995) 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. They report an accuracy percentage in the mid-nineties when applied to *Dixon*, a name found to be quite ambiguous.

A recent knowledge–lean approach to sense discrimination is discussed in (Schütze in press 1998). Ambiguous words are clustered into sense groups based on *second–order co–occurrences*: two instances of an ambiguous word are assigned to the same sense if the words that they co–occur with likewise co–occur with similar words in the training data. Schütze evaluates sense groupings based on their effectiveness in several information retrieval problems.[4]

Future Work

There are several issues to address in future work. First, the possibility of using the EM algorithm as a starting point for Gibbs Sampling seems particularly intriguing in that it addresses the limitations of both approaches. Second, we would like to use models other than Naive Bayes in these knowledge–lean approaches. More complicated models, while potentially resulting in distributions that are inappropriate for the EM algorithm, could provide stronger disambiguation results when used in combination with Gibbs Sampling. We would also like to investigate the use of informative priors in Gibbs Sampling. Finally, we will continue investigating local–context features in the hopes of increasing our accuracy with minority senses without substantially increasing the dimensionality of the problem.

Acknowledgments

This research was supported by the Office of Naval Research under grant number N00014-95-1-0776.

References

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

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

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

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

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

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

Geweke, J. 1992. Evaluating the accuracy of sampling–based approaches to calculating posterior moments. In Bernardo, J.; Berger, J.; Dawid, A.; and Smith, A., eds., *Bayesian Statistics 4*. Oxford: Oxford University Press.

Lauritzen, S. 1995. The EM algorithm for graphical association models with missing data. *Computational Statistics and Data Analysis* 19:191–201.

Leacock, C.; Towell, G.; and Voorhees, E. 1993. Corpus-based statistical sense resolution. In *Proceedings of the ARPA Workshop on Human Language Technology*, 260–265.

Meng, X., and van Dyk, D. 1997. The EM algorithm – an old folk–song sung to a new fast tune (with discussion). *Journal of Royal Statistics Society, Series B* 59(3):511—567.

Mooney, R. 1996. 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*, 82–91.

Ng, H., and Lee, H. 1996. 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*, 40–47.

Ng, H. 1997. Exemplar–based word sense disambiguation: Some recent improvements. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 208–213.

Pedersen, T., and Bruce, R. 1997. Distinguishing word senses in untagged text. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 197–207.

Schütze, H. (in press) 1998. Automatic word sense discrimination. *Computational Linguistics*.

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

[4] In Schütze's evaluation, tagged text is not required to label the sense groupings and establish the accuracy of the disambiguation experiment. Thus the experiment is fully automatic and free from dependence on any external knowledge source.