Discriminating Among Word Senses Using McQuitty's Similarity Analysis

Amruta Purandare
Department of Computer Science
University of Minnesota
Duluth, MN 55812
pura0010@d.umn.edu

Abstract

This paper presents an unsupervised method for discriminating among the senses of a given target word based on the context in which it occurs. Instances of a word that occur in similar contexts are grouped together via McQuitty's Similarity Analysis, an agglomerative clustering algorithm. The context in which a target word occurs is represented by surface lexical features such as unigrams, bigrams, and second order co-occurrences. This paper summarizes our approach, and describes the results of a preliminary evaluation we have carried out using data from the SENSEVAL-2 English lexical sample and the *line* corpus.

1 Introduction

Word sense discrimination is the process of grouping or clustering together instances of written text that include similar usages of a given target word. The instances that form a particular cluster will have used the target word in similar contexts and are therefore presumed to represent a related meaning. This view follows from the strong contextual hypothesis of (Miller and Charles, 1991), which states that *two words are semantically similar to the extent that their contextual representations are similar.*

Discrimination is distinct from the more common problem of word sense disambiguation in at least two respects. First, the number of possible senses a target word may have is usually not known in discrimination, while disambiguation is often viewed as a classification problem where a word is assigned to one of several pre–existing possible senses. Second, discrimination utilizes features and information that can be easily extracted from raw corpora, whereas disambiguation often relies on supervised learning from sense–tagged training examples. However, the creation of sense–tagged data is time consuming and results in a knowledge acquisition bottleneck that severely limits the portability and scalability of techniques that employ it. Discrimination does not suffer from this problem since there is no expensive preprocessing, nor are any external knowledge sources or manually annotated data required.

The objective of this research is to extend previous work in discrimination by (Pedersen and Bruce, 1997), who developed an approach using agglomerative clustering. Their work relied on McQuitty's Similarity Analysis using localized contextual features. While the approach in this paper also adopts McQuitty's method, it is distinct in that it uses a larger number of features that occur both locally and globally in the instance being discriminated. It also incorporates several ideas from later work by (Schütze, 1998), including the reliance on a separate "training" corpus of raw text from which to identify contextual features, and the use of second order co–occurrences (socs) as feature for discrimination.

Our near term objectives for this research include determining to what extent different types of features impact the accuracy of unsupervised discrimination. We are also interested in assessing how different measures of similarity such as the matching coefficient or the cosine affect overall performance. Once we have refined our clustering techniques, we will incorporate them into a method that automatically assigns sense labels to discovered clusters by using information from a machine readable dictionary.

This paper continues with a more detailed discussion of the previous work that forms the foundation for our research. We then present an overview of the features used to represent the context of a target word, and go on to describe an experimental evaluation using the SENSEVAL-2 lexical sample data. We close with a discussion of our results, a summary of related work, and an outline of our future directions.

2 Previous Work

The work in this paper builds upon two previous approaches to word sense discrimination, those of (Pedersen and Bruce, 1997) and (Schütze, 1998). Pedersen and Bruce developed a method based on agglomerative clustering using McQuitty's Similarity Analysis (McQuitty, 1966), where the context of a target word is represented using localized contextual features such as collocations and part of speech tags that occur within one or two positions of the target word. Pedersen and Bruce demonstrated that despite it's simplicity, McQuitty's method was more accurate than Ward's Method of Minimum Variance and the EM Algorithm for word sense discrimination.

McQuitty's method starts by assuming that each instance is a separate cluster. It merges together the pair of clusters that have the highest average similarity value. This continues until a specified number of clusters is found, or until the similarity measure between every pair of clusters is less than a predefined cutoff. Pedersen and Bruce used a relatively small number of features, and employed the matching coefficient as the similarity measure. Since we use a much larger number of features, we are experimenting with the cosine measure, which scales similarity based on the number of non–zero features in each instance.

By way of contrast, (Schütze, 1998) performs discrimination through the use of two different kinds of context vectors. The first is a word vector that is based on co–occurrence counts from a separate training corpus. Each word in this corpus is represented by a vector made up of the words it co-occurs with. Then, each instance in a test or evaluation corpus is represented by a vector that is the average of all the vectors of all the words that make up that instance. The context in which a target word occurs is thereby represented by second order co–occurrences, which are words which co–occur with the co–occurrences of the target word. Discrimination is carried out by clustering instance vectors using the EM Algorithm.

The approach described in this paper proceeds as follows. Surface lexical features are identified in a training corpus, which is made up of instances that consists of a sentence containing a given target word, plus one or two sentences to the left or right of it. Similarly defined instances in the test data are converted into vectors based on this feature set, and a similarity matrix is constructed using either the matching coefficient or the cosine. Thereafter McQuitty's Similarity Analysis is used to group together instances based on the similarity of their context, and these are evaluated relative to a manually created gold standard.

3 Discrimination Features

We carry out discrimination based on surface lexical features that require little or no preprocessing to identify. They consist of unigrams, bigrams, and second order co–occurrences.

Unigrams are single words that occur in the same context as a target word. Bag–of–words feature sets made up of unigrams have had a long history of success in text classification and word sense disambiguation (Mooney, 1996), and we believe that despite creating quite a bit of noise can provide useful information for discrimination.

Bigrams are pairs of words which occur together in the same context as the target word. They may include the target word, or they may not. We specify a window of size five for bigrams, meaning that there may be up to three intervening words between the first and last word that make up the bigram. As such we are defining bigrams to be non–consecutive word sequences, which could also be considered a kind of co–occurrence feature. Bigrams have recently been shown to be very successful features in supervised word sense disambiguation (Pedersen, 2001). We believe this is because they capture middle distance co–occurrence relations between words that occur in the context of the target word.

Second order co–occurrences are words that occur with co-occurrences of the target word. For example, suppose that *line* is the target word. Given *telephone line* and *telephone bill*, *bill* would be considered a second order co–occurrence of *line* since it occurs with *telephone*, a first order co–occurrence of *line*.

We define a window size of five in identifying second order co–occurrences, meaning that the first order co–occurrence must be within five positions of the target word, and the second order co–occurrence must be within five positions of the first order co–occurrence. We only select those second order co–occurrences which co–occur more than once with the first order co-occurrences which in turn co-occur more than once with the target word within the specified window.

We employ a stop list to remove high frequency non–content words from all of these features. Unigrams that are included in the stop list are not used as features. A bigram is rejected if any word composing it is a stop word. Second order co–occurrences that are stop words or those that co–occur with stop words are excluded from the feature set.

After the features have been identified in the training data, all of the instances in the test data are converted into binary feature vectors $(F_1, F_2, \ldots, F_q)$ that represent whether the features found in the training data have occurred in a particular test instance. In order to cluster these instances, we measure the pair–wise similarities between them using matching and cosine coefficients.

These values are formatted in a $N \times N$ similarity matrix such that cell (l, k) contains the similarity measure between instances l and k. This information serves as the input to the clustering algorithm that groups together the most similar instances.

4 Experimental Methodology

We evaluate our method using two well known sources of sense–tagged text. In supervised learning sense–tagged text is used to induce a classifier that is then applied to held out test data. However, our approach is purely unsupervised and we only use the sense tags to carry out an automatic evaluation of the discovered clusters. We follow Schütze's strategy and use a "training" corpus only to extract features and ignore the sense tags.

In particular, we use subsets of the *line* data (Leacock et al., 1993) and the English lexical sample data from the SENSEVAL-2 comparative exercise among word sense disambiguation systems (Edmonds and Cotton, 2001).

The *line* data contains 4,146 instances, where each consists of two to three sentences where a single occurrence of *line* has been manually tagged with one of six possible senses. We randomly select 100 instances of each sense for test data, and 200 instances of each sense for training. This gives a total of 600 evaluation instances, and 1200 training instances. This is done to test the quality of our discrimination method when senses are uniformly distributed and where no particular sense is dominant.

The standard distribution of the SENSEVAL-2 data consists of 8,611 training instances and 4,328 test instances. Each instance is made up of two to three sentences where a single target word has been manually tagged with a sense (or senses) appropriate for that context. There are 73 distinct target words found in this data; 29 nouns, 29 verbs, and 15 adjectives. Most of these words have less than 100 test instances, and approximately twice that number of training examples. In general these are relatively small samples for an unsupervised approach, but we are developing techniques to increase the amount of training data for this corpus automatically.

We filter the SENSEVAL-2 data in three different ways to prepare it for processing and evaluation. First, we insure that it only includes instances whose actual sense is among the top five most frequent senses as observed in the training data for that word. We believe that this is an aggressive number of senses for a discrimination system to attempt, considering that (Pedersen and Bruce, 1997) experimented with 2 and 3 senses, and (Schütze, 1998) made binary distinctions.

Second, instances may have been assigned more than one correct sense by the human annotator. In order to simplify the evaluation process, we eliminate all but the most frequent of multiple correct answers.

Third, the SENSEVAL-2 data identifies target words that are proper nouns. We have elected not to use that information and have removed these P tags from the data. After carrying out these preprocessing steps, the number of training and test instances is 7,476 and 3,733.

5 Evaluation Technique

We specify an upper limit on the number of senses that McQuitty's algorithm can discover. In these experiments this value is five for the SENSEVAL-2 data, and six for *line*. In future experiments we will specify even higher values, so that the algorithm is forced to create larger number of clusters with very few instances when the actual number of senses is smaller than the given cutoff. About a third of the words in the SENSEVAL-2 data have fewer than 5 senses, so even now the clustering algorithm is not always told the correct number of clusters it should find.

Once the clusters are formed, we access the actual correct sense of each instance as found in the sense–tagged text. This information is never utilized prior to evaluation. We use the sense–tagged text as a gold standard by which we can evaluate the discovered sense clusters. We assign sense tags to clusters such that the resulting accuracy is maximized.

For example, suppose that five clusters (C1 – C5) have been discovered for a word with 100 instances, and that the number of instances in each cluster is 25, 20, 10, 25, and 20. Suppose that there are five actual senses (S1 – S5), and the number of instances for each sense is 20, 20, 20, 20, and 20. Figure 1 shows the resulting confusion matrix if the senses are assigned to clusters in numeric order. After this assignment is made, the accuracy of the clustering can be determined by finding the sum of the diagonal, and dividing by the total number of instances, which in this case leads to accuracy of 10% (10/100). However, clearly there are assignments of senses to clusters that would lead to better results.

Thus, the problem of assigning senses to clusters becomes one of reordering the columns of the confusion such that the diagonal sum is maximized. This corresponds to several well known problems, among them the Assignment Problem in Operations Research, and determining the maximal matching of a bipartite graph. Figure 2 shows the maximally accurate assignment of senses to clusters, which leads to accuracy of 70% (70/100).

During evaluation we assign one cluster to at most one sense, and vice versa. When the number of discovered clusters is the same as the number of senses, then there is a 1 to 1 mapping between them. When the number of clusters is greater than the number of actual senses, then some clusters will be left unassigned. And when the

	S1	S2	S3	S4	S5	
C1:	5	20	0	0	0	25
C2:	10	0	5	0	5	20
C3:	0	0	0	0	10	10
C4:	0	0	15	5	5	25
C5:	5	0	0	15	0	20
	20	20	20	20	20	100

Figure 1: Numeric Assignment

	S2	S1	S5	S3	S4	
C1:	20	5	0	0	0	25
C2:	0	10	5	5	0	20
C3:	0	0	10	0	0	10
C4:	0	0	5	15	5	25
C5:	0	5	0	0	15	20
	20	20	20	20	20	100

Figure 2: Maximally Accurate Assignment

number of senses is greater than the number of clusters, some senses will not be assigned to any cluster.

We determine the precision and recall based on this maximally accurate assignment of sense tags to clusters. Precision is defined as the number of instances that are clustered correctly divided by the number of instances clustered, while recall is the number of instances clustered correctly over the total number of instances.

To be clear, we do not believe that word sense discrimination must be carried out relative to a pre–existing set of senses. In fact, one of the great advantages of an unsupervised approach is that it need not be relative to any particular set of senses. We carry out this evaluation technique in order to improve the performance of our clustering algorithm, which we will then apply on text where sense–tagged data is not available.

An alternative means of evaluation is to have a human inspect the discovered clusters and judge them based on the semantic coherence of the instances that populate each cluster, but this is a more time consuming and subjective method of evaluation that we will pursue in future.

6 Experimental Results

For each word in the SENSEVAL-2 data and *line*, we conducted various experiments, each of which uses a different combination of measure of similarity and features.

Features are identified from the training data. Our features consist of unigrams, bigrams, or second order co–occurrences. We employ each of these three types of features separately, and we also create a mixed set that is the union of all three sets. We convert each evaluation instance into a feature vector, and then convert those into a similarity matrix using either the matching coefficient or the cosine.

Table 1 contains overall precision and recall for the nouns, verbs, and adjectives overall in the SENSEVAL-2 data, and for *line*. The SENSEVAL-2 values are derived from 29 nouns, 28 verbs, and 15 adjectives from the SENSEVAL-2 data. The first column lists the part of speech, the second shows the feature, the third lists the measure of similarity, the fourth and the fifth show precision and recall, the sixth shows the percentage of the majority sense, and the final column shows the number of words in the given part of speech that gave accuracy greater than the percentage of the majority sense. The value of the majority sense is derived from the sense–tagged data we use in evaluation, but this is not information that we would presume to have available during actual clustering.

Table 1: Experimental Results

pos	feat	meas	prec	rec	maj	> maj
noun	soc	cos	0.49	0.48	0.57	6/29
		mat	0.54	0.52	0.57	7/29
	big	cos	0.53	0.50	0.57	5/29
		mat	0.52	0.49	0.57	3/29
	uni	cos	0.50	0.49	0.57	7/29
		mat	0.52	0.50	0.57	8/29
	mix	cos	0.50	0.48	0.57	6/29
		mat	0.54	0.51	0.57	5/29
verb	soc	cos	0.51	0.49	0.51	11/28
		mat	0.50	0.47	0.51	6/28
	big	cos	0.54	0.45	0.51	5/28
		mat	0.53	0.43	0.51	5/28
	uni	cos	0.42	0.41	0.51	13/28
		mat	0.43	0.41	0.51	9/28
	mix	cos	0.43	0.41	0.51	12/28
		mat	0.42	0.41	0.51	7/28
adj	soc	cos	0.59	0.54	0.64	1/15
		mat	0.59	0.55	0.64	1/15
	big	cos	0.56	0.51	0.64	0/15
		mat	0.55	0.50	0.64	0/15
	uni	cos	0.55	0.50	0.64	1/15
		mat	0.58	0.53	0.64	0/15
	mix	cos	0.50	0.44	0.64	0/15
		mat	0.59	0.54	0.64	2/15
line	soc	cos	0.25	0.25	0.17	1/1
		mat	0.23	0.23	0.17	1/1
	big	cos	0.19	0.18	0.17	1/1
		mat	0.18	0.17	0.17	1/1
	uni	cos	0.21	0.21	0.17	1/1
		mat	0.20	0.20	0.17	1/1
	mix	cos	0.21	0.21	0.17	1/1
		mat	0.20	0.20	0.17	1/1

For the SENSEVAL-2 data, on average the precision and recall of the clustering as determined by our evaluation method is less than that of the majority sense, regardless of which features or measure are used. However, for nouns and verbs, a relatively significant number of individual words have precision and recall values higher than that of the majority sense. The adjectives are an exception to this, where words are very rarely disambiguated more accurately than the percentage of the majority sense. However, many of the adjectives have very high frequency majority senses, which makes this a difficult standard for an unsupervised method to reach. When examining the distribution of instances in clusters, we find that the algorithm tends to seek more balanced distributions, and is unlikely to create a single long cluster that would result in high accuracy for a word whose true distribution of senses is heavily skewed towards a single sense.

We also note that the precision and recall of the clustering of the *line* data is generally better than that of the majority sense regardless of the features or measures employed. We believe there are two explanations for this. First, the number of training instances for the line data is significantly higher (1200) than that of the SENSEVAL-2 words, which typically have 100–200 training instances per word. The number and quality of features identified improves considerably with an increase in the amount of training data. Thus, the amount of training data available for feature identification is critically important. We believe that the SENSEVAL-2 data could be augmented with training data taken from the World Wide Web, and we plan to pursue such approaches and see if our performance on the evaluation data improves as a result.

At this point we do not observe a clear advantage to using the cosine measure or matching coefficient. This surprises us somewhat, as the number of features employed is generally in the thousands, and the number of non–zero features can be quite large. It would seem that simply counting the number of matching features would be inferior to the cosine measure, but this is not the case. This remains an interesting issue that we will continue to explore, with these and other measures of similarity.

Finally, there is not a single feature that does best in all parts of speech. Second order co–occurrences seem to do well with nouns and adjectives, while bigrams result in accurate clusters for verbs. We also note that second order co–occurrences do well with the *line* data. As yet we have drawn no conclusions from these results, but it is clearly a vital issue to investigate further.

7 Related Work

Unsupervised approaches to word sense discrimination have been somewhat less common in the computational linguistics literature, at least when compared to supervised approaches to word sense disambiguation.

There is a body of work at the intersection of supervised and unsupervised approaches, which involves using a small amount of training data in order to automatically create more training data, in effect bootstrapping from the small sample of sense–tagged data. The best example of such an approach is (Yarowsky, 1995), who proposes a method that automatically identifies collocations that are indicative of the sense of a word, and uses those to iteratively label more examples.

While our focus has been on Pedersen and Bruce, and on Schütze, there has been other work in purely unsupervised approaches to word sense discrimination.

(Fukumoto and Suzuki, 1999) describe a method for discriminating among verb senses based on determining which nouns co–occur with the target verb. Collocations are extracted which are indicative of the sense of a verb based on a similarity measure they derive.

(Pantel and Lin, 2002) introduce a method known as Committee Based Clustering that discovers word senses. The words in the corpus are clustered based on their distributional similarity under the assumption that semantically similar words will have similar distributional characteristics. In particular, they use Pointwise Mutual Information to find how close a word is to its context and then determine how similar the contexts are using the cosine coefficient.

8 Future Work

Our long term goal is to develop a method that will assign sense labels to clusters using information found in machine readable dictionaries. This is an important problem because clusters as found in discrimination have no sense tag or label attached to them. While there are certainly applications for unlabeled sense clusters, having some indication of the sense of the cluster would bring discrimination and disambiguation closer together. We will treat glosses as found in a dictionary as vectors that we project into the same space that is populated by instances as we have already described. A cluster could be assigned the sense of the gloss whose vector it was most closely located to.

This idea is based loosely on work by (Niwa and Nitta, 1994), who compare word co–occurrence vectors derived from large corpora of text with co–occurrence vectors based on the definitions or glosses of words in a machine readable dictionary. A co–occurrence vector indicates how often words are used with each other in a large corpora or in dictionary definitions. These vectors can be projected into a high dimensional space and used to measure the distance between concepts or words. Niwa and Nitta show that while the co–occurrence data from a dictionary has different characteristics that a co–occurrence vector derived from a corpus, both provide useful information about how to categorize a word based on its meaning. Our future work will mostly attempt to merge clusters found from corpora with meanings in dictionaries where presentation techniques like co–occurrence vectors could be useful.

There are a number of smaller issues that we are investigating. We are also exploring a number of other types of features, as well as varying the formulation of the features we are currently using. We have already conducted a number of experiments that vary the window sizes employed with bigrams and second order co–occurrences, and will continue in this vein. We are also considering the use of other measures of similarity beyond the matching coefficient and the cosine. We do not stem the training data prior to feature identification, nor do or employ fuzzy matching techniques when converting evaluation instances into feature vectors. However, we believe both might lead to increased numbers of useful features being identified.

9 Conclusions

We have presented an unsupervised method of word sense discrimination that employs a range of surface lexical features, and relies on similarity based clustering. We have evaluated this method in an extensive experiment that shows that our method can achieve precision and recall higher than the majority sense of a word for a reasonably large number of cases. We believe that increases in the amount of training data employed in this method will yield to considerably improved results, and have outlined our plans to address this and several other issues.

10 Acknowledgments

This research is being conducted as a part of my M.S. thesis in Computer Science at the University of Minnesota, Duluth. I am grateful to my thesis advisor, Dr. Ted Pedersen, for his help and guidance.

I have been fully supported by a National Science Foundation Faculty Early CAREER Development Award (#0092784) during the 2002–2003 academic year.

I would like to thank the Director of Computer Science Graduate Studies, Dr. Carolyn Crouch, and the Associate Vice Chancellor, Dr. Stephen Hedman, for their support in providing a travel award to attend the Student Research Workshop at HLT-NAACL 2003.

References

P. Edmonds and S. Cotton, editors. 2001. *Proceedings of the Senseval–2 Workshop*. Association for Computational Linguistics, Toulouse, France.

F. Fukumoto and Y. Suzuki. 1999. Word sense disambiguation in untagged text based on term weight learning. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 209–216, Bergen.

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

L. McQuitty. 1966. Similarity analysis by reciprocal pairs for discrete and continuous data. *Educational and Psychological Measurement*, 26:825–831.

G.A. Miller and W.G. Charles. 1991. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

R. Mooney. 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*, pages 82–91, May.

Y. Niwa and Y. Nitta. 1994. Co-occurrence vectors from corpora versus distance vectors from dictionaries. In *Proceedings of the Fifteenth International Conference on Computational Linguistics*, pages 304–309, Kyoto, Japan.

P. Pantel and D. Lin. 2002. Discovering word senses from text. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining-2002*.

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

T. Pedersen. 2001. A decision tree of bigrams is an accurate predictor of word sense. In *Proceedings of the Second Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 79–86, Pittsburgh, July.

H. Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

D. Yarowsky. 1995. 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.