Discriminating Among Word Meanings By Identifying Similar Contexts

Amruta Purandare and **Ted Pedersen**
Department of Computer Science
University of Minnesota
Duluth, MN 55812
{pura0010,tpederse}@d.umn.edu
http://senseclusters.sourceforge.net

Abstract

Word sense discrimination is an unsupervised clustering problem, which seeks to discover which instances of a word/s are used in the same meaning. This is done strictly based on information found in raw corpora, without using any sense tagged text or other existing knowledge sources. Our particular focus is to systematically compare the efficacy of a range of lexical features, context representations, and clustering algorithms when applied to this problem.

Introduction

The goal of word sense discrimination is to group multiple instances of a word/s into clusters, where each cluster represents a distinct meaning of that word/s. For example, we might wish to differentiate among sentences containing the word *line* that refer to a *product line* versus those that refer to a *telephone line.* Or, we might wish to identify which instances of *bat* and *club* refer to a stick used for hitting. Thus, word sense discrimination seeks to identify different words that refer to the same meaning (synonyms), and also discover the different senses of a given word.

Word sense discrimination is often based on the premise that words that are used in similar contexts will tend to have similar meanings (Miller & Charles 1991). This allows word sense discrimination to be reduced to the problem of finding classes of similar contexts such that each discovered class represents a word sense. Two very distinct approaches that rely on this premise have been proposed by (Pedersen & Bruce 1998) and (Schütze 1998). We compare a number of their techniques via an experimental evaluation, and propose extensions to these methods based on these results.

Our strategy is to represent the contexts in which words occur using a variety of lexical features that are easy to identify in large corpora. As a result our approach conveniently scales to larger data since no manually annotated text is required. These contexts are then converted into similarity or vector spaces which can then be clustered using a variety of different algorithms. The objective of our research is to determine which combinations of features, context representations, and clustering algorithms result in better word sense discrimination.

Context Representation

We maintain a separation between the instances to be discriminated (i.e., the test data) and the data from which features are selected (i.e., the training data). This allows us to explore variations in the training data while maintaining a consistent test set, and also avoid any limitations that might be caused by selecting features from test data when discriminating a small number of instances.

We identify various lexical features in the training data using a combination of frequency counts and measures of association. These features include unigrams, bigrams, and co-occurrences. Unigrams are individual words that occur with high frequency, while bigrams are strongly associated pairs of words that occur within a few positions of each other. Co-occurrences are strongly associated unordered pairs of words that include the word to be discriminated.

Once the features are selected, each word to be discriminated is represented in terms of the features that directly occur in the surrounding context, loosely following (Pedersen & Bruce 1998). We refer to this as a first order context vector representation, since the vector represents the immediate context in which the word to be discriminated occurs.

We also represent context indirectly with a second order context vector representation, as suggested by (Schütze 1998). This technique creates first order vectors for the individual words that occur in a context, and then averages those together to create a generalized vector that captures second order relationships among words in that context.

Clustering

Vector space clustering algorithms directly use the vector representations of the contexts as their input. However, similarity space algorithms require a similarity matrix that provides the pair–wise similarities between the given contexts.

In both similarity and vector space, we have used a hierarchical agglomerative clustering algorithm with the average link criteria function, and a hybrid algorithm called bisected K-means that combines the partitional K-means algorithm with hierarchical divisive clustering.

We do not know a-priori the number of possible senses to be discriminated. Hence, in our experiments, we specify an upper limit on the number of clusters to be discovered. Our belief is that a successful sense discrimination algorithm

will automatically discover approximately the same number of clusters as actual senses for a word, and that the excess clusters will contain very few instances and can be safely discarded without fear of affecting overall performance.

Experimental Results

We have carried out experiments using the *line*, *hard*, and *serve* sense–tagged corpora, each of which has approximately 4,000 instances. We have also used a subset of the words from the sense–tagged SENSEVAL-2 corpus. These represent much smaller quantities of data, since there are 50–200 sense tagged instances for each word. We evaluate our results in terms of precision and recall, which measure the degree to which the discovered clusters correspond with the true word senses as indicated by the sense–tags.

Experimental results described in (Purandare 2003) and (Purandare & Pedersen 2004b) show that bigrams and co-occurrences with high degrees of association according to the log–likelihood ratio prove to be particularly useful. We have also found that the first order context representation results in more accurate discrimination when discriminating the larger *line*, *hard* and *serve* corpora. The second order representation performs better when given smaller corpora such as the SENSEVAL-2 words.

We hypothesize that the sparseness of the features in smaller corpora causes the first order methods to perform poorly, while second order methods are able to better identify features when given limited data. However, both representations lead to very sparse representations regardless of the corpora size. As such we are experimenting with Singular Value Decomposition (SVD) to reduce sparsity and convert the word level feature space into a conceptual semantic space, as is done in Latent Semantic Analysis (Landauer, Foltz, & Laham 1998).

Future Work

In our experiments to date, our training data has consisted of instances of the word/s to be discriminated. Thus, the training and test data are fairly homogeneous, and focus on the particular word to be discriminated. We plan to conduct experiments where the features are selected from a huge corpus that is not specific to the words being discriminated. We will draw from a variety of sources, including the British National Corpus, the English GigaWord Corpus, and the Web.

Our motivation is that huge corpora will provide more generic co–occurrence information about words without regard to a particular word to be discriminated. It is not clear if this will be more effective than our current approach, which captures co–occurrence behavior in the immediate context of the word to be discriminated.

We are also developing a method to attach descriptive labels to the discovered clusters. These labels will define the sense of the cluster, and will be based on the most characteristic features of the instances that belong to that cluster. We will then map the discovered clusters to established dictionary senses by matching the automatically derived labels to the existing definitions. This will allow us to associate our clusters with an existing sense inventory, making it possible to perform fully automatic word sense disambiguation that does not rely on any manually annotated text.

Conclusions

We have conducted an extensive comparative analysis of word sense discrimination techniques using first order and second order context vectors, where both can be employed in similarity and vector space. We conclude that for larger amounts of homogeneous data such as the *line*, *hard* and *serve* data, the first order context vector representation and average link clustering algorithm as proposed by (Pedersen & Bruce 1998) is most effective. We believe this is the case because in a large sample of data, it is very likely that the features that occur in the training data will also occur in the test data, making it possible to represent test instances with fairly rich feature sets.

When given smaller amounts of data like the SENSEVAL-2 words, second order context vectors and a hybrid clustering method such as repeated K–means perform better. We believe that this occurs because in small and sparse data, direct first order features are seldom observed in both the training and the test data. However, the indirect second order co–occurrence relationships that are captured by these methods provide sufficient information for discrimination to proceed.

Acknowledgments

This research is supported by a National Science Foundation Faculty Early CAREER Development Award (#0092784).

All of the experiments described in this paper were conducted using version 0.47 of the SenseClusters package (Purandare & Pedersen 2004a), which is freely available from SourceForge.

References

Landauer, T.; Foltz, P.; and Laham, D. 1998. An introduction to latent semantic analysis. *Discourse Processes* 25:259–284.

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

Pedersen, T., and Bruce, R. 1998. Knowledge lean word sense disambiguation. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, 800–805.

Purandare, A., and Pedersen, T. 2004a. Senseclusters - finding clusters that represent word senses. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*.

Purandare, A., and Pedersen, T. 2004b. Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of the Conference on Computational Natural Language Learning*.

Purandare, A. 2003. Discriminating among word senses using mcquitty's similarity analysis. In *Proceedings of the HLT-NAACL 2003 Student Research Workshop*, 19–24.

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