Identifying Similar Words and Contexts in Natural Language with SenseClusters

Ted Pedersen and **Anagha Kulkarni**
Department of Computer Science
University of Minnesota
Duluth, MN 55812
{tpederse,kulka020}@d.umn.edu
http://senseclusters.sourceforge.net

Abstract

SenseClusters is a freely available intelligent system that clusters together similar contexts in natural language text. Thereafter it assigns identifying labels to these clusters based on their content. It is a purely unsupervised approach that is language independent, and uses no knowledge other than what is available in raw un-annotated corpora. In addition to clustering similar contexts, it can be used to identify synonyms and sets of related words. It has been applied to a diverse range of problems, including proper name disambiguation, word sense discrimination, email organization, and document clustering. SenseClusters is a complete system that supports feature selection from large corpora, several different context representation schemes, various clustering algorithms, the creation of descriptive and discriminating labels for the discovered clusters, and evaluation relative to gold standard data.

Introduction

Many problems in natural language processing reduce to that of identifying units of text (i.e., contexts) that are similar syntactically and/or semantically. For example, word sense disambiguation assigns meanings to words based on the contexts in which they occur, where words that occur in similar contexts are presumed to have the same meaning (Miller & Charles 1991). Email organization tries to group together messages based on the similarity of their content. Paraphrase detection finds sentences that express approximately the same idea using different words.

SenseClusters can be applied to all of these problems, since it allows a user to measure the similarity of contexts, regardless of how large or small they may be. A context may be a few words that surround a given target word in a sentence, or an email message of few paragraphs or an entire article. SenseClusters provides a general framework that can be applied in much the same way regardless of the size or type of context to be clustered.

One key to this generality is that SenseClusters relies on lexical features made up of single words or pairs of words that can be easily identified in a wide variety of corpora. This is by design, so as to allow SenseClusters to be highly portable across languages and domains. While parsers and other tools for extracting syntactic features are widely available for English, this is not true for many other languages.

These lexical features are used to create direct first order or indirect second order representation of the contexts. Thereafter these new representations of the contexts are clustered, and each resulting cluster is assigned a label that has a descriptive and discriminating component that both summarizes and identifies the contents of the cluster. For some problems, it may be feasible for a human to create a gold standard of comparison by manually clustering some contexts. If such data is available, SenseClusters can measure how closely it agrees with the standard in order to assess its performance.

Feature Selection

The Ngram Statistics Package[1] is used to identify the following types of lexical features: unigrams, bigrams, co-occurrences, and target co-occurrences.

Unigrams are individual words that occur above a certain frequency cutoff. These are often effective (but noisy) indicators of content. Bigrams are ordered pairs of words that are judged statistically significant by a measure of association. There may optionally be intervening words between the two words that are ignored. Bigrams can often provide very specific unambiguous clues regarding the content of a context. Co-occurrences are unordered bigrams, and if one of the words is a designated target word, then it is referred to as a target co–occurrence. We maintain a list of stop–words, and exclude these words as unigram features, and eliminate any bigram, co-occurrence, or target co–occurrence that is made up of one or two stop–words.

SenseClusters identifies lexical features from the contexts to be clustered (called the test data) or from a separate sample of data. Selecting features from separate data may be particularly useful when the test data is too small to extract meaningful features.

Context Representation

SenseClusters represents the contexts to be clustered using either a first order or second order representation. For the first order representation, a matrix where each row represents a context and the columns represent the identified lex-

[1] http://ngram.sourceforge.net

ical features is created. The cell values of this matrix indicates which of the previously identified lexical features occur in that particular context. This follows from the word sense discrimination approach of (Pedersen & Bruce 1997), who used first order vectors based on local syntactic features to represent contexts.

Our use of second order contexts is adapted from the word sense discrimination method of (Schütze 1998). A set of bigram, co-occurrence, or target co–occurrence features identified during the feature selection step are used to construct a co-occurrence matrix. The bigram features result in an asymmetric matrix where the rows represent the first word, and the columns represent the second word. In the case of either type of co-occurrence feature, the matrix is symmetric.

Each row in this co-occurrence matrix can be treated as a vector that represents the word at that row. A context vector is represented by taking the vectors of the words in the context and averaging them together.

The context by features matrix created for first order representation or the co-occurrence matrix of second order context representation may have its dimensionality reduced by using Singular Value Decomposition (SVD). Note that SVD causes reduction in column dimensionality only, the number of rows remain unchanged.

The resulting context vectors represented using either first order representation or second order representation are clustered in the next step.

Clustering

SenseClusters provides a seamless interface to CLUTO, a Clustering Toolkit (Karypis 2002), which implements a range of clustering techniques suitable for both vector and similarity spaces. Vector spaces take the context vectors created as described above and cluster them directly. As an alternative, pair–wise similarity among the context vectors can be computed using a variety of metrics (cosine, Jaccard, Dice, etc.) and stored in a similarity matrix which is then clustered.

Labeling of Clusters

Once the clusters of contexts are identified, SenseClusters examines the contents of each cluster to arrive at a descriptive and a discriminating label that identifies the contents of the cluster. We use measures of association to identify significant bigrams that occur in each cluster, and use the top 5 to 10 ranked bigrams in each cluster as a descriptive label. However, these descriptive labels may exist in a number of clusters, so we also identify those bigrams that are highly ranked and unique to each cluster, and use those to create a discriminating label for each cluster.

Applications

SenseClusters can be applied to any language processing task that is based on identifying similar contexts. from a sentence (or phrase) to a paragraph or to a much longer documents. It can be used to identify sets of related words that occur in similar contexts.

Email often consists of approximately one or two paragraphs of context, and as such it is an ideal length for processing by SenseClusters. We have carried out experiments clustering email and newsgroup text using SenseClusters, and are able to re–create a hierarchical classification of email messages by topic.

We have used SenseClusters to perform name discrimination (e.g., (Pedersen, Purandare, & Kulkarni 2005), (Kulkarni 2005)) which identifies contexts associated with different individuals who happen to have the same name.

SenseClusters was originally developed to carry out word sense discrimination (e.g., (Purandare 2003), (Purandare & Pedersen 2004)). This groups together the contexts in which a given word occurs, such that each cluster represents a different sense of that word.

Finally, SenseClusters creates a word by word co–occurrence matrix to support second order representations. This matrix can also be used to identify sets of words that are distribution-ally similar based on the contexts in which they occur.

Acknowledgments

This work has been partially supported by a National Science Foundation Faculty Early CAREER Development award (Grant #0092784).

SenseClusters is an open source software project that is freely distributed under the GNU Public License (GPL) via http://senseclusters.sourceforge.net/

References

Karypis, G. 2002. CLUTO - a clustering toolkit. Technical Report 02-017, University of Minnesota, Department of Computer Science.

Kulkarni, A. 2005. Unsupervised discrimination and labeling of ambiguous names. In *Companion Volume to the Proceedings of ACL 2005 - Student Research Workshop*.

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

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.

Pedersen, T.; Purandare, A.; and Kulkarni, A. 2005. Name discrimination by clustering similar contexts. In *Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, 220–231.

Purandare, A., and Pedersen, T. 2004. Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of the Conference on Computational Natural Language Learning*, 41–48.

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

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