CS229 Final Project
Using WordNet and Clustering for Semantic Role Labeling

Richard Fulton
rafulton@stanford.edu

Ebrahim Parvand
eparvand@cs.stanford.edu

December 14, 2007

Abstract

In this paper, we compare the performance of several clustering algorithms on the task of semantic role labeling. We use a baseline system based on logistic regression classifiers, and also a distributional clustering algorithm based on word association lists. We use a Latent Semantic Analysis system to compare to two previously implemented clustering algorithms, k-means and a more comprehensive discriminative clustering algorithm. We also focus on features we can extract through specialized corpuses such as WordNet. Overall, we demonstrate that semantically clustering text leads to significant improvements on semantic role labeling tasks.

1 Introduction

Semantic Role Labeling (*SRL*) is the process of annotating the predicate-argument structure in text with semantic labels [3, 8]. To make this slightly clearer, we are attempting to label the arguments of a verb, which are labeled sequentially from Arg0 upwards. Arg0 is generally the subject of transitive verbs, Arg1 the direct object, and so on. Consider, for example, the following sentence:

Then [$_{Arg0}$ John] gave [$_{Arg2}$ Jim] [$_{Arg1}$ the apple].

Arg0 is John, the subject of the verb gave, Arg1 is the apple, the direct object, and Arg2 is Jim, the indirect object (the one to whom the apple is given).

While many systems that perform SRL use Support Vector Machines, treating the problem of tagging parsed constituents as multi-class classification problems [7], we use ℓ_2-regularized logistic regression for our models because of its vastly quicker training time. We use multiclass logistic regression as our baseline. At points in this paper, we also compare to a K-means implementation. For our K-Means, the feature we extract for our logistic model is simply the cluster in which a contituent headword appears. We also compare to a discriminative clustering algorithm which was developed in our earlier research, but that is not the focus of this paper. In the discriminative clustering algorithm, each word is given a score as to how much it is associated with each cluster, as opposed to discrete assignment.

2 Dataset and Associated Learning Tasks

Our datasets are drawn from PropBank, which is an annotated corpus of semantic roles [5]. We take the parses from PropBank to extract headwords for each verb, then ignore the parse once we have extracted our data. Our dataset consists of a series of verbs, arguments to those verbs, and a list of headwords of noun (or adjective or verb) phrases which are examples of those arguments that we have in the data. We divide the dataset into a training set and a test set. For each verb, for each argument of that verb, we place in our training set the first 70% of constituent headwords in the corresponding list. We test on two data sets: 50_10 and 0_0. 0_0 is the complete data

set, whereas in the 50_0 set we remove all constituent headwords that occur less than fifty times and all verb arguments with less than ten constituents.

We wish to extract as much information as possible without the benefit of context. Context provides many beneficial features, but if we can show improvements on semantic role labeling without context, it seems likely that systems using contextual information as features in their parses or semantic role labeling will benefit from our findings.

3 Extracting Features from WordNet

3.1 Overview

WordNet [2] is a lexicon of the English language that also captures the semantic relationships between words. It also contains information about different senses of words, combines synonyms into structures called *synsets*, and facilitates the processing of these features via an API. WordNet is made freely available for processing and analysis from its developers / maintainers at Princeton University.

Of particular importance to us are the hypernym / hyponym relationships captured by WordNet. These terms deserve some explanation:

- **Hypernym**
  Word A is a hypernym of word B if B is a type of A (the "IS A" relationship).
  e.g. A car is a vehicle, so *vehicle* is a hypernym of *car*.
- **Hyponym**
  The converse of hypernym.
  e.g. A duck is a bird, so *duck* is a hyponym of *bird*.

3.2 Features

We hypothesize that two nouns that have a large intersection of hypernyms are likely to play the same semantic role in a sentence for a given verb. For instance, for the word *backpack*, we find from WordNet that a

backpack *is a* bag, which *is a* container, which *is an* instrumentality or instrument, which *is an* artifact or artefact, which *is a* whole or unit, which *is a* object or phyical object, which *is a* physical entity, which *is an* entity

It comes as no surprise that the noun *knapsack* and *backpack* have the exact same hypernyms and are even in the same synset in WordNet. Therefore, we strongly suspect that the two nouns would almost if not always be the same argument for a given verb, e.g. *I wear a {knapsack, backpack} to school everyday.* This is the justification for our WordNet features.

Using WordNet, we extracted the distinct hypernym synsets of given nouns, generated unique integer keys for these synsets, and appended the keys to the nouns' feature vectors using a sparse representation (i.e. only the indices that are present are actually

in the vector). Since some words have multiple senses (e.g. *dog* has both the sense of "a member of the genus Canis" and "someone who is morally reprehensible" according to WordNet), we ensured that only distinct hypernyms were placed in the feature vectors (e.g. for the above example, both senses of *dog* have *entity* as hypernyms, but we only insert it once into the feature vector for *dog*).

4 Distributional Clustering

Distributional clustering in relation to NLP tasks refers to the method of clustering words according to different distributions in particular syntactic contexts [6]. We implement distributional clustering using both a dependency based thesaurus developed by Dekang Lin [4], and a database of semantic distances within WordNet.

As a side note, we mentioned in our project proposal that we would be experimenting with unlabeled data. After further consideration, we decided that distributional clustering serves the same purpose as unlabeled data, namely the introduction of large amounts of outside information to the training set. Therefore we thought it not necessary to experiment with unlabeled data at this time.

4.1 Dependency-Based Thesaurus

As part of our research, we extracted features from a database of word associations developed by Dekang Lin [4]. For a given input word, the database returns a list of words and scores indicating how closely each word is related to the input word. For some of the associations for the word car, see table 1(a).

index	word	score
2	truck	0.895040
17	taxi	0.455772
50	tank	0.343982
102	road	0.279985
200	village	0.233997

(a)

index	word	score
1	linebacker	0.881162
8	player	0.714504
52	dimaggio	0.523563
225	official	0.251566
352	republican	0.182153

(b)

Figure 1: Word Association Lists

For every word x, we compute its association features as follows:
1. For every word in the word-verb pairs of the training examples, precompute and store each word's association list.
2. Each time word x appears in a training word's association list, add a feature for that list.

This provides an alternate way of determining relationships between different words. If we see a word in the test set that we have never seen before in the training set, if that word appears on another word's association list we know that the two words are similar to some degree.

4.2 Semantic Distance in WordNet

We also explored word associations in the WordNet database as an alternate dataset for distributional clustering. Of the several distance metrics proposed in [1], we chose to utilize the Jiang-Conrath measure, which was shown to give the best results. For word associations for the word quarterback, see table 1(b).

5 Latent Semantic Analysis for Semantic Role Labeling

We implemented a Latent Semantic Analysis(LSA) system that uses Singular Value Decomposition (SVD) to reduce the dimensionality of the training matrix to find meaningful semantic patterns. We ran SVD on the matrix of word counts versus training examples. In the following decomposition equation, M is our data matrix.

$$M = U\Sigma V^* \tag{5.1}$$

Specifically, we examined the U decomposition matrix that corresponded to the training words versus training examples. For each training example, if that training example had a high relation to a particular word according to the decomposition matrix, we add that training example as a feature. If a relation score was too low, we would skip it.

Not all of the words we are labeling in the training set are nouns, so we also experimented with running SVD on only the nouns. Though we did not have time to try other variations of SVD like weighting the terms using *tf-idf*, running SVD on nouns only had a similar effect to that of *tf-idf*, because the nouns are in general the more important words in the data set.

6 Results and Discussion

6.1 WordNet Features

As the results show, our extracted WordNet features increased performance from baseline significantly. This confirms our hypothesis that using outside features (i.e. features from the entire English language) rather than extracting features just from the data set is indeed advantageous.

We suspect that this is the case for the following reason. Note that the incraese from baseline is actually *larger* in the full data set than in the smaller. We believe this is because in the testing phase of the larger data set, there is a higher probability of having seen a word or a synonym of that word or a word with which it shares significant hypernyms during *training*, and therefore the classifier has already learned parameters for the some of the test word's hypernym synset features for the given verb.

6.2 Distributional Clustering Features

The results we achieved for distributional clustering were comparable to the results of the WordNet features. On the smaller 50_10 data set, we matched the k-means accuracy. However on the full data set, we achieved significant accuracy gains over our k-means baseline. We hypothesize that the reasons are similar to those described in 6.1.

6.3 Latent Semantic Analysis

LSA was meant to be another baseline to compare with our k-means baseline, and though it improved a great deal over the baseline accuracy, it never reached the level of k-means results. We found that by running SVD on nouns only, the accuracy increased by more than a percentage point.

6.4 Feature Combination

Though we did not experiment a great deal with the combinations of different feature sets, we did combine the features we thought would yield the highest score. By combining the baseline logistic regression features, discriminative clustering features, and the distributional clustering features, we achieved higher accuracy on non-contextual semantic role labeling than all previous attempts.

Method	**50_10 (Small)**	**0_0 (Full)**
Logistic Regression (baseline)	81.8%	70.7%
Baseline + K-Means Cluster IDs	86.0%	74.0%
Baseline + WordNet Features	84.8%	76.0%
Baseline + LSA	83.5%	71.7%
Baseline + LSA w/ nouns only	84.4%	72.7%
Baseline + Distributional Clusters	86.0%	75.7%
Discriminative Clustering	88.3%	78.7%
Baseline + Distributional Clustering (WordNet)	84.4%	74.9%
Baseline + Distributional + Discriminative Clustering	88.7%	80.4%

Figure 2: Test accuracy on the small and large datasets using various methods

7 Conclusion

In this paper, we have presented results on context free semantic role labeling using clustering algorithms and feature extraction from outside data sources. We saw that adding extra "semantic" information via automatic clustering of constituent words to our logistic classifier significantly improved performance.

8 Acknowledgments

We would like to thank David Vickrey for many useful conversations and algorithmic advice, as well as providing our datasets.

References

[1] A. Budanitsky and G. Hirst. Semantic distance in wordnet. Semantic Distance.

[2] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, Massachusetts, 1998.

[3] D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 28:245–288, 2002.

[4] D. Lin. Automatic retrieval and clustering of similar words. *COLING-ACL98*, 1998.

[5] M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31:71–106, 2005.

[6] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. Clustering Words Paper.

[7] S. Pradhan, K. Hacioglu, W. Ward, J. H. Martin, and D. Jurafsky. Semantic role parsing: Adding semantic structure to unstructured text. *Proceedings of ICDM*, 2003.

[8] S. Pradhan, W. Ward, K. Hacioglu, J. H. Martin, and D. Jurafsky. Semantic role labeling using different syntactic views. *Proceedings of ACL*, pages 581–588, 2005.