A Practical Solution to the Problem of Automatic Part-of-Speech Induction from Text

Reinhard Rapp
University of Mainz, FASK
D-76711 Germersheim, Germany
rapp@mail.fask.uni-mainz.de

Abstract

The problem of part-of-speech induction from text involves two aspects: Firstly, a set of word classes is to be derived automatically. Secondly, each word of a vocabulary is to be assigned to one or several of these word classes. In this paper we present a method that solves both problems with good accuracy. Our approach adopts a mixture of statistical methods that have been successfully applied in word sense induction. Its main advantage over previous attempts is that it reduces the syntactic space to only the most important dimensions, thereby almost eliminating the otherwise omnipresent problem of data sparseness.

1 Introduction

Whereas most previous statistical work concerning parts of speech has been on tagging, this paper deals with part-of-speech induction. In part-of-speech induction two phases can be distinguished: In the first phase a set of word classes is to be derived automatically on the basis of the distribution of the words in a text corpus. These classes should be in accordance with human intuitions, i.e. common distinctions such as nouns, verbs and adjectives are desirable. In the second phase, based on its observed usage each word is assigned to one or several of the previously defined classes.

The main reason why part-of-speech induction has received far less attention than part-of-speech tagging is probably that there seemed no urgent need for it as linguists have always considered classifying words as one of their core tasks, and as a consequence accurate lexicons providing such information are readily available for many languages. Nevertheless, deriving word classes automatically is an interesting intellectual challenge with relevance to cognitive science. Also, advantages of the automatic systems are that they should be more objective and can provide precise information on the likelihood distribution for each of a word's parts of speech, an aspect that is useful for statistical machine translation.

The pioneering work on class based n-gram models by Brown et al. (1992) was motivated by such considerations. In contrast, Schütze (1993) by applying a neural network approach put the emphasis on the cognitive side. More recent work includes Clark (2003) who combines distributional and morphological information, and Freitag (2004) who uses a hidden Marcov model in combination with co-clustering.

Most studies use abstract statistical measures such as perplexity or the F-measure for evaluation. This is good for quantitative comparisons, but makes it difficult to check if the results agree with human intuitions. In this paper we use a straightforward approach for evaluation. It involves checking if the automatically generated word classes agree with the word classes known from grammar books, and whether the class assignments for each word are correct.

2 Approach

In principle, word classification can be based on a number of different linguistic principles, e.g. on phonology, morphology, syntax or semantics. However, in this paper we are only interested in syntactically motivated word classes. With syntactic classes the aim is that words belonging to the same class can substitute for one another in a sentence without affecting its grammaticality.

As a consequence of the substitutability, when looking at a corpus words of the same class typically have a high agreement concerning their left and right neighbors. For example, nouns are frequently preceded by words like *a*, *the*, or *this*, and succeeded by words like *is*, *has* or *in*. In statistical

Proceedings of the ACL Interactive Poster and Demonstration Sessions,
pages 77–80, Ann Arbor, June 2005.

terms, words of the same class have a similar frequency distribution concerning their left and right neighbors. To some extend this can also be observed with indirect neighbors, but with them the effect is less salient and therefore we do not consider them here.

The co-occurrence information concerning the words in a vocabulary and their neighbors can be stored in a matrix as shown in table 1. If we now want to discover word classes, we simply compute the similarities between all pairs of rows using a vector similarity measure such as the cosine coefficient and then cluster the words according to these similarities. The expectation is that unambiguous nouns like *breath* and *meal* form one cluster, and that unambiguous verbs like *discuss* and *protect* form another cluster.

Ambiguous words like *link* or *suit* should not form a tight cluster but are placed somewhere in between the noun and the verb clusters, with the exact position depending on the ratios of the occurrence frequencies of their readings as either a noun or a verb. As this ratio can be arbitrary, according to our experience ambiguous words do not severely affect the clustering but only form some uniform background noise which more or less cancels out in a large vocabulary.[1] Note that the correct assignment of the ambiguous words to clusters is not required at this stage, as this is taken care of in the next step.

This step involves computing the differential vector of each word from the centroid of its closest cluster, and to assign the differential vector to the most appropriate other cluster. This process can be repeated until the length of the differential vector falls below a threshold or, alternatively, the agreement with any of the centroids becomes too low. This way an ambiguous word is assigned to several parts of speech, starting from the most common and proceeding to the least common. Figure 1 illustrates this process.

[1] An alternative to relying on this fortunate but somewhat unsatisfactory effect would be not to use global co-occurrence vectors but local ones, as successfully proposed in word sense induction (Rapp, 2004). This means that every occurrence of a word obtains a separate row vector in table 1. The problem with the resulting extremely sparse matrix is that most vectors are either orthogonal to each other or duplicates of some other vector, with the consequence that the dimensionality reduction that is indispensable for such matrices does not lead to sensible results. This problem is not as severe in word sense induction where larger context windows are considered.

The procedure that we described so far works in theory but not well in practice. The problem with it is that the matrix is so sparse that sampling errors have a strong negative effect on the results of the vector comparisons. Fortunately, the problem of data sparseness can be minimized by reducing the dimensionality of the matrix. An appropriate algebraic method that has the capability to reduce the dimensionality of a rectangular matrix is *Singular Value Decomposition* (SVD). It has the property that when reducing the number of columns the similarities between the rows are preserved in the best possible way. Whereas in other studies the reduction has typically been from several ten thousand to a few hundred, our reduction is from several ten thousand to only three. This leads to a very strong generalization effect that proves useful for our particular task.

	left neighbors				right neighbors			
	a	we	the	you	a	can	is	well
breath	11	0	18	0	0	14	19	0
discuss	0	17	0	10	9	0	0	8
link	14	6	11	7	10	9	14	3
meal	15	0	17	0	0	9	12	0
protect	0	15	1	12	14	0	0	4
suit	5	0	8	3	0	8	16	2

Table 1. Co-occurrence matrix of adjacent words.

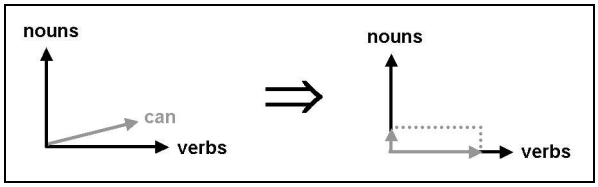

Figure 1. Constructing the parts of speech for *can*.

3 Procedure

Our computations are based on the unmodified text of the 100 million word *British National Corpus* (BNC), i.e. including all function words and without lemmatization. By counting the occurrence frequencies for pairs of adjacent words we compiled a matrix as exemplified in table 1. As this matrix is too large to be processed with our algorithms (SVD and clustering), we decided to restrict the number of rows to a vocabulary appropriate for evaluation purposes. Since we are not aware of any standard vocabulary previously used in related work, we manually selected an ad hoc list of 50

words with BNC frequencies between 5000 and 6000 as shown in table 2. The choice of 50 was motivated by the intention to give complete clustering results in graphical form. As we did not want to deal with morphology, we used base forms only. Also, in order to be able to subjectively judge the results, we only selected words where we felt reasonably confident about their possible parts of speech. Note that the list of words was compiled before the start of our experiments and remained unchanged thereafter.

The co-occurrence matrix based on the restricted vocabulary and all neighbors occurring in the BNC has a size of 50 rows times 28,443 columns. As our transformation function we simply use the logarithm after adding one to each value in the matrix.[2] As usual, the one is added for smoothing purposes and to avoid problems with zero values. We decided not to use a sophisticated association measure such as the log-likelihood ratio because it has an inappropriate value characteristic that prevents the SVD, which is conducted in the next step, from finding optimal dimensions.[3]

The purpose of the SVD is to reduce the number of columns in our matrix to the main dimensions. However, it is not clear how many dimensions should be computed. Since our aim of identifying basic word classes such as nouns or verbs requires strong generalizations instead of subtle distinctions, we decided to take only the three main dimensions into account, i.e. the resulting matrix has a size of 50 rows times 3 columns.[4] The last step in our procedure involves applying a clustering algorithm to the 50 words corresponding to the rows in the matrix. We used hierarchical clustering with average linkage, a linkage type that provides considerable tolerance concerning outliers.

4 Results and Evaluation

Our results are presented as dendrograms which in contrast to 2-dimensional dot-plots have the advantage of being able to correctly show the true distances between clusters. The two dendrograms in figure 2 where both computed by applying the procedure as described in the previous section, with the only difference that in generating the upper dendrogram the SVD-step has been omitted, whereas in generating the lower dendrogram it has been conducted. Without SVD the expected clusters of verbs, nouns and adjectives are not clearly separated, and the adjectives *widely* and *rural* are placed outside the adjective cluster. With SVD, all 50 words are in their appropriate clusters and the three discovered clusters are much more salient. Also, *widely* and *rural* are well within the adjective cluster. The comparison of the two dendrograms indicates that the SVD was capable of making appropriate generalizations. Also, when we look inside each cluster we can see that ambiguous words like *suit*, *drop* or *brief* are somewhat closer to their secondary class than unambiguous words.

Having obtained the three expected clusters, the next investigation concerns the assignment of the ambiguous words to additional clusters. As described previously, this is done by computing differential vectors, and by assigning these to the most similar other cluster. Hereby for the cosine similarity we set a threshold of 0.8. That is, only if the similarity between the differential vector and its closest centroid was higher than 0.8 we assigned the word to this cluster and continued to compute differential vectors. Otherwise we assumed that the differential vector was caused by sampling errors and aborted the process of searching for additional class assignments.

The results from this procedure are shown in table 2 where for each of the 50 words all computed classes are given in the order as they were obtained by the algorithm, i.e. the dominant assignments are listed first. Although our algorithm does not name the classes, for simplicity we interpret them in the obvious way, i.e. as nouns, verbs and adjectives. A comparison with WordNet 2.0 choices is given in brackets. For example, +N means that WordNet lists the additional assignment *noun*, and -A indicates that the assignment *adjective* found by the algorithm is not listed in WordNet.

According to this comparison, for all 50 words the first reading is correct. For 16 words an additional second reading was computed which is correct in 11 cases. 16 of the WordNet assignments are missing, among them the verb readings for *reform*, *suit*, and *rain* and the noun reading for *serve*. However, as many of the WordNet assignments seem rare, it is not clear in how far the omissions can be attributed to shortcomings of the algorithm.

[2] For arbitrary vocabularies the row vectors should be divided by the corpus frequency of the corresponding word.
[3] We are currently investigating if replacing the log-likelihood values by their ranks can solve this problem.
[4] Note that larger matrices can require a few more dimensions.

accident N	expensive A	reform N (+V)
belief N	familiar A (+N)	rural A
birth N (+V)	finance N V	screen N (+V)
breath N	grow V N (-N)	seek V (+N)
brief A N	imagine V	serve V (+N)
broad A (+N)	introduction N	slow A V
busy A V	link N V	spring N A V (-A)
catch V N	lovely A (+N)	strike N V
critical A	lunch N (+V)	suit N (+V)
cup N (+V)	maintain V	surprise N V
dangerous A	occur V N (-N)	tape N V
discuss V	option N	thank V A (-A)
drop V N	pleasure N	thin A (+V)
drug N (+V)	protect V	tiny A
empty A V (+N)	prove V	widely A N (-N)
encourage V	quick A (+N)	wild A (+N)
establish V	rain N (+V)	

Table 2. Computed parts of speech for each word.

5 Summary and Conclusions

This work was inspired by previous work on word sense induction. The results indicate that part of speech induction is possible with good success based on the analysis of distributional patterns in text. The study also gives some insight how SVD is capable of significantly improving the results.

Whereas in a previous paper (Rapp, 2004) we found that for word sense induction the local clustering of local vectors is more appropriate than the global clustering of global vectors, for part-of-speech induction our conclusion is that the situation is exactly the other way round, i.e. the global clustering of global vectors is more adequate (see footnote 1). This finding is of interest when trying to understand the nature of syntax versus semantics if expressed in statistical terms.

Acknowledgements

I would like to thank Manfred Wettler and Christian Biemann for comments, Hinrich Schütze for the SVD-software, and the DFG (German Research Society) for financial support.

References

Brown, Peter F.; Della Pietra, Vincent J.; deSouza, Peter V.; Lai, Jennifer C.; Mercer, Robert L. (1992). Class-based n-gram models of natural language. *Computational Linguistics* 18(4), 467-479.

Clark, Alexander (2003). Combining distributional and morphological information for part of speech induction. *Proceedings of 10th EACL*, Budapest, *59-66.*

Freitag, Dayne (2004). Toward unsupervised whole-corpus tagging. *Proceedings of COLING*, Geneva, 357-363.

Rapp, Reinhard (2004). A practical solution to the problem of automatic word sense induction. *Proceedings of ACL (Companion Volume)*, Barcelona, 195-198.

Schütze, Hinrich (1993). Part-of-speech induction from scratch. *Proceedings of ACL*, Columbus, 251-258.

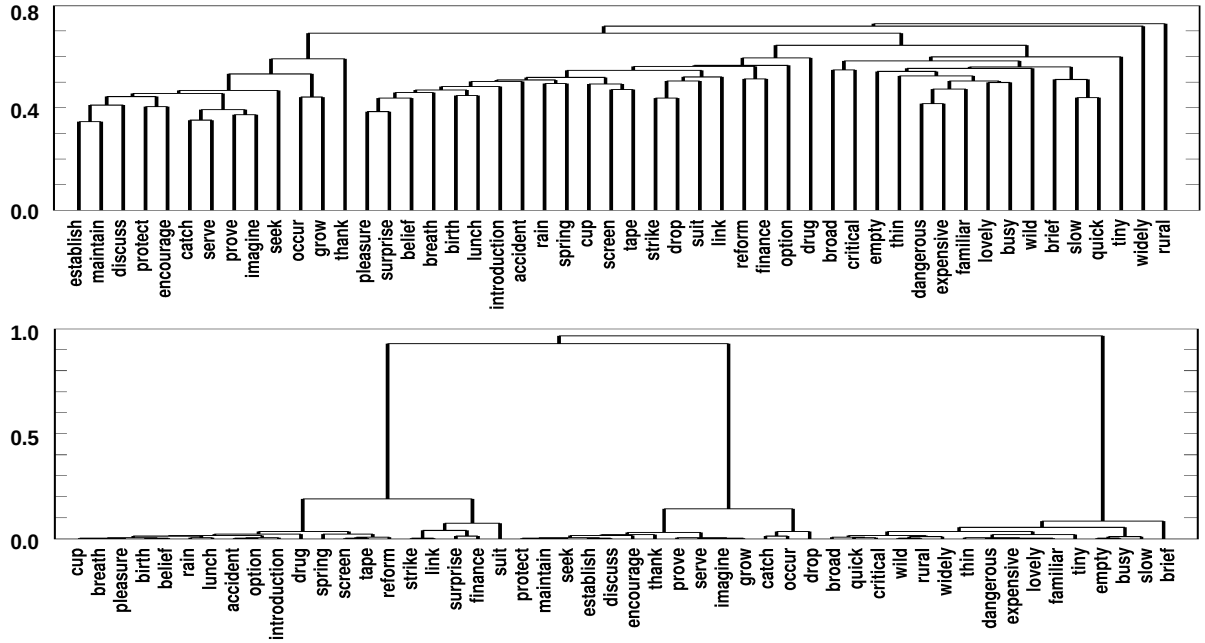

Figure 2. Syntactic similarities with (lower dendrogram) and without SVD (upper dendrogram).