A Practical Solution to the Problem of Automatic Word Sense Induction

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

Abstract

Recent studies in word sense induction are based on clustering global co-occurrence vectors, i.e. vectors that reflect the overall behavior of a word in a corpus. If a word is semantically ambiguous, this means that these vectors are mixtures of all its senses. Inducing a word's senses therefore involves the difficult problem of recovering the sense vectors from the mixtures. In this paper we argue that the demixing problem can be avoided since the contextual behavior of the senses is directly observable in the form of the local contexts of a word. From human disambiguation performance we know that the context of a word is usually sufficient to determine its sense. Based on this observation we describe an algorithm that discovers the different senses of an ambiguous word by clustering its contexts. The main difficulty with this approach, namely the problem of data sparseness, could be minimized by looking at only the three main dimensions of the context matrices.

1 Introduction

The topic of this paper is word sense induction, that is the automatic discovery of the possible senses of a word. A related problem is word sense disambiguation: Here the senses are assumed to be known and the task is to choose the correct one when given an ambiguous word in context. Whereas until recently the focus of research had been on sense disambiguation, papers like Pantel & Lin (2002), Neill (2002), and Rapp (2003) give evidence that sense induction now also attracts attention.

In the approach by Pantel & Lin (2002), all words occurring in a parsed corpus are clustered on the basis of the distances of their co-occurrence vectors. This is called global clustering. Since (by looking at differential vectors) their algorithm allows a word to belong to more than one cluster, each cluster a word is assigned to can be considered as one of its senses. A problem that we see with this approach is that it allows only as many senses as clusters, thereby limiting the granularity of the meaning space. This problem is avoided by Neill (2002) who uses local instead of global clustering. This means, to find the senses of a given word only its close associations are clustered, that is for each word new clusters will be found.

Despite many differences, to our knowledge almost all approaches to sense induction that have been published so far have a common limitation: They rely on global co-occurrence vectors, i.e. on vectors that have been derived from an entire corpus. Since most words are semantically ambiguous, this means that these vectors reflect the sum of the contextual behavior of a word's underlying senses, i.e. they are mixtures of all senses occurring in the corpus.

However, since reconstructing the sense vectors from the mixtures is difficult, the question is if we really need to base our work on mixtures or if there is some way to directly observe the contextual behavior of the senses thereby avoiding the mixing beforehand. In this paper we suggest to look at local instead of global co-occurrence vectors. As can be seen from human performance, in almost all cases the local context of an ambiguous word is sufficient to disambiguate its sense. This means that the local context of a word usually carries no ambiguities. The aim of this paper is to show how this observation whose application tends to severely suffer from the sparse-data problem can be successfully exploited for word sense induction.

2 Approach

The basic idea is that we do not cluster the global co-occurrence vectors of the words (based on an entire corpus) but local ones which are derived from the contexts of a single word. That is, our computations are based on the concordance of a word. Also, we do not consider a term/term but a term/context matrix. This means, for each word that we want to analyze we get an entire matrix.

Let us exemplify this using the ambiguous word *palm* with its *tree* and *hand* senses. If we assume that our corpus has six occurrences of *palm*, i.e. there are six local contexts, then we can derive six local co-occurrence vectors for *palm*. Considering only strong associations to *palm*, these vectors could, for example, look as shown in table 1.

The dots in the matrix indicate if the respective word occurs in a context or not. We use binary

vectors since we assume short contexts where words usually occur only once. By looking at the matrix it is easy to see that contexts c1, c3, and c6 seem to relate to the *hand* sense of *palm*, whereas contexts c2, c4, and c5 relate to its *tree* sense. Our intuitions can be resembled by using a method for computing vector similarities, for example the cosine coefficient or the (binary) Jaccard-measure. If we then apply an appropriate clustering algorithm to the context vectors, we should obtain the two expected clusters. Each of the two clusters corresponds to one of the senses of *palm*, and the words closest to the geometric centers of the clusters should be good descriptors of each sense.

However, as matrices of the above type can be extremely sparse, clustering is a difficult task, and common algorithms often deliver sub-optimal results. Fortunately, the problem of matrix 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 or square matrix in an optimal way is *singular value decomposition* (SVD). As shown by Schütze (1997) by reducing the dimensionality a generalization effect can be achieved that often improves the results. The approach that we suggest in this paper involves reducing the number of columns (contexts) and then applying a clustering algorithm to the row vectors (words) of the resulting matrix. This works well since it is a strength of SVD to reduce the effects of sampling errors and to close gaps in the data.

	c1	c2	c3	c4	c5	c6
arm	•		•			
beach		•			•	
coconut		•		•	•	
finger	•		•			
hand	•		•			•
shoulder	•					•
tree		•		•		

Table 1: Term/context matrix for the word *palm*.

3 Algorithm

As in previous work (Rapp, 2002), our computations are based on a partially lemmatized version of the British National Corpus (BNC) which has the function words removed. Starting from the list of 12 ambiguous words provided by Yarowsky (1995) which is shown in table 2, we created a concordance for each word, with the lines in the concordances each relating to a context window of ±20 words. From the concordances we computed 12 term/context-matrices (analogous to table 1) whose binary entries indicate if a word occurs in a particular context or not. Assuming that the amount of information that a context word provides depends on its association strength to the ambiguous word, in each matrix we removed all words that are not among the top 30 first order associations to the ambiguous word. These top 30 associations were computed fully automatically based on the log-likelihood ratio. We used the procedure described in Rapp (2002), with the only modification being the multiplication of the log-likelihood values with a triangular function that depends on the logarithm of a word's frequency. This way preference is given to words that are in the middle of the frequency range. Figures 1 to 3 are based on the association lists for the words *palm* and *poach*.

Given that our term/context matrices are very sparse with each of their individual entries seeming somewhat arbitrary, it is necessary to detect the regularities in the patterns. For this purpose we applied the SVD to each of the matrices, thereby reducing their number of columns to the three main dimensions. This number of dimensions may seem low. However, it turned out that with our relatively small matrices (matrix size is the occurrence frequency of a word times the number of associations considered) it was sometimes not possible to compute more than three singular values, as there are dependencies in the data. Therefore, we decided to use three dimensions for all matrices.

The last step in our procedure involves applying a clustering algorithm to the 30 words in each matrix. For our condensed matrices of 3 rows and 30 columns this is a rather simple task. We decided to use the hierarchical clustering algorithm readily available in the MATLAB (MATrix LABoratory) programming language. After some testing with various similarity functions and linkage types, we finally opted for the cosine coefficient and single linkage which is the combination that apparently gave the best results.

axes: grid/tools	bass: fish/music
crane: bird/machine	drug: medicine/narcotic
duty: tax/obligation	motion: legal/physical
palm: tree/hand	plant: living/factory
poach: steal/boil	sake: benefit/drink
space: volume/outer	tank: vehicle/container

Table 2: Ambiguous words and their senses.

4 Results

Before we proceed to a quantitative evaluation, by looking at a few examples let us first give a qualitative impression of some results and consider the contribution of SVD to the performance of our algorithm. Figure 1 shows a dendrogram for the word *palm* (corpus frequency in the lemmatized BNC: 2054) as obtained after applying the algo-

rithm described in the previous section, with the only modification that the SVD step was omitted, i.e. no dimensionality reduction was performed. The horizontal axes in the dendrogram is dissimilarity (1 – cosine), i.e. 0 means identical items and 1 means no similarity. The vertical axes has no special meaning. Only the order of the words is chosen in such a way that line crossings are avoided when connecting clusters.

As we can see, the dissimilarities among the top 30 associations to *palm* are all in the upper half of the scale and not very distinct. The two expected clusters for *palm*, one relating to its *hand* and the other to its *tree* sense, have essentially been found. According to our judgment, all words in the upper branch of the hierarchical tree are related to the *hand* sense of *palm*, and all other words are related to its *tree* sense. However, it is somewhat unsatisfactory that the word *frond* seems equally similar to both senses, whereas intuitively we would clearly put it in the *tree* section.

Let us now compare figure 1 to figure 2 which has been generated using exactly the same procedure with the only difference that the SVD step (reduction to 3 dimensions) has been conducted in this case. In figure 2 the similarities are generally at a higher level (dissimilarities lower), the relative differences are bigger, and the two expected clusters are much more salient. Also, the word *frond* is now well within the *tree* cluster. Obviously, figure 2 reflects human intuitions better than figure 1, and we can conclude that SVD was able to find the right generalizations. Although space constraints prevent us from showing similar comparative diagrams for other words, we hope that this novel way of comparing dendrograms makes it clearer what the virtues of SVD are, and that it is more than just another method for smoothing.

Our next example (figure 3) is the dendrogram for *poach* (corpus frequency: 458). It is also based on a matrix that had been reduced to 3 dimensions. The two main clusters nicely distinguish between the two senses of *poach*, namely *boil* and *steal*. The upper branch of the hierarchical tree consists of words related to cooking, the lower one mainly contains words related to the unauthorized killing of wildlife in Africa which apparently is an important topic in the BNC.

Figure 3 nicely demonstrates what distinguishes the clustering of local contexts from the clustering of global co-occurrence vectors. To see this, let us bring our attention to the various species of animals that are among the top 30 associations to *poach*. Some of them seem more often affected by cooking (pheasant, chicken, salmon), others by poaching (elephant, tiger, rhino). According to the diagram only the rabbit is equally suitable for both activities, although fortunately its affinity to cooking is lower than it is for the chicken, and to poaching it is lower than it is for the rhino.

That is, by clustering local contexts our algorithm was able to separate the different kinds of animals according to their relationship to *poach*. If we instead clustered global vectors, it would most likely be impossible to obtain this separation, as from a global perspective all animals have most properties (context words) in common, so they are likely to end up in a single cluster. Note that what we exemplified here for animals applies to all linkage decisions made by the algorithm, i.e. all decisions must be seen from the perspective of the ambiguous word.

This implies that often the clustering may be counterintuitive from the global perspective that as humans we tend to have when looking at isolated words. That is, the clusters shown in figures 2 and 3 can only be understood if the ambiguous words they are derived from are known. However, this is exactly what we want in sense induction.

In an attempt to provide a quantitative evaluation of our results, for each of the 12 ambiguous words shown in table 1 we manually assigned the top 30 first-order associations to one of the two senses provided by Yarowsky (1995). We then looked at the first split in our hierarchical trees and assigned each of the two clusters to one of the given senses. In no case was there any doubt on which way round to assign the two clusters to the two given senses. Finally, we checked if there were any misclassified items in the clusters.

According to this judgment, on average 25.7 of the 30 items were correctly classified, and 4.3 items were misclassified. This gives an overall accuracy of 85.6%. Reasons for misclassifications include the following: Some of the top 30 associations are more or less neutral towards the senses, so even for us it was not always possible to clearly assign them to one of the two senses. In other cases, outliers led to a poor first split, like if in figure 1 the first split would be located between *frond* and the rest of the vocabulary. In the case of *sake* the beverage sense is extremely rare in the BNC and therefore was not represented among the top 30 associations. For this reason the clustering algorithm had no chance to find the expected clusters.

5 Conclusions and prospects

From the observations described above we conclude that avoiding the mixture of senses, i.e. clustering local context vectors instead of global co-occurrence vectors, is a good way to deal with the problem of word sense induction. However, there is a pitfall, as the matrices of local vectors are extremely sparse. Fortunately, our simulations

suggest that computing the main dimensions of a matrix through SVD solves the problem of sparseness and greatly improves clustering results.

Although the results that we presented in this paper seem useful even for practical purposes, we can not claim that our algorithm is capable of finding all the fine grained distinctions that are listed in manually created dictionaries such as the Longman Dictionary of Contemporary English (LDOCE), or in lexical databases such as WordNet. For future improvement of the algorithm we see two main possibilities:

1) Considering all context words instead of only the top 30 associations would further reduce the sparse data problem. However, this requires finding an appropriate association function. This is difficult, as for example the log-likelihood ratio, although delivering almost perfect rankings, has an inappropriate value characteristic: The increase in computed strengths is over-proportional for stronger associations. This prevents the SVD from finding optimal dimensions.

2) The principle of avoiding mixtures can be applied more consequently if not only local instead of global vectors are used, but if also the parts of speech of the context words are considered. By operating on a part-of-speech tagged corpus those sense distinctions that have an effect on part of speech can be taken into account.

Acknowledgements

I would like to thank Manfred Wettler, Robert Dale, Hinrich Schütze, and Raz Tamir for help and discussions, and the DFG for financial support.

References

Neill, D. B. (2002). *Fully Automatic Word Sense Induction by Semantic Clustering*. Cambridge University, Master's Thesis, M.Phil. in Computer Speech.

Pantel, P.; Lin, D. (2002). Discovering word senses from text. In: *Proceedings of ACM SIGKDD*, Edmonton, 613–619.

Rapp, R. (2002). The computation of word associations: comparing syntagmatic and paradigmatic approaches. *Proc. of 19th COLING*, Taipei, ROC, Vol. 2, 821–827.

Rapp, R. (2003). Word sense discovery based on sense descriptor dissimilarity. In: *Ninth Machine Translation Summit*, New Orleans, 315–322.

Schütze, H. (1997). *Ambiguity Resolution in Language Learning: Computational and Cognitive Models*. Stanford: CSLI Publications.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In: *Proc. of 33rd ACL*, Cambridge, MA, 189–196.

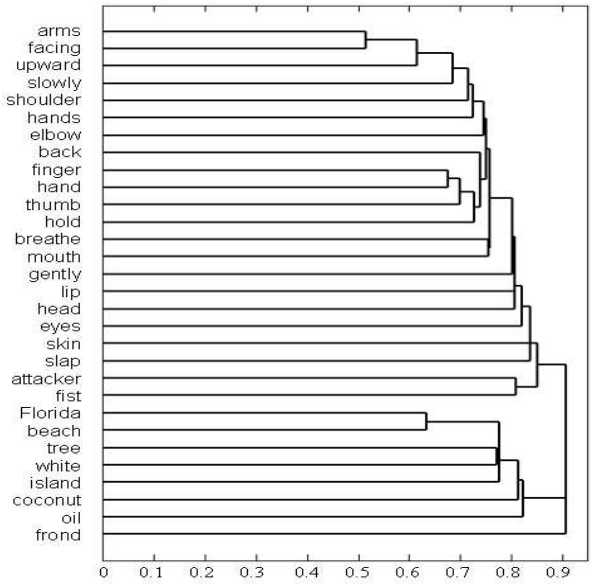

Figure 1: Clustering results for *palm* without SVD.

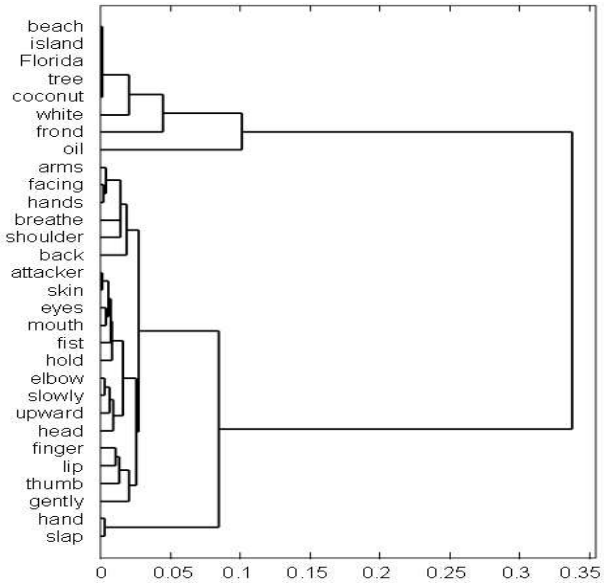

Figure 2: Clustering results for *palm* with SVD.

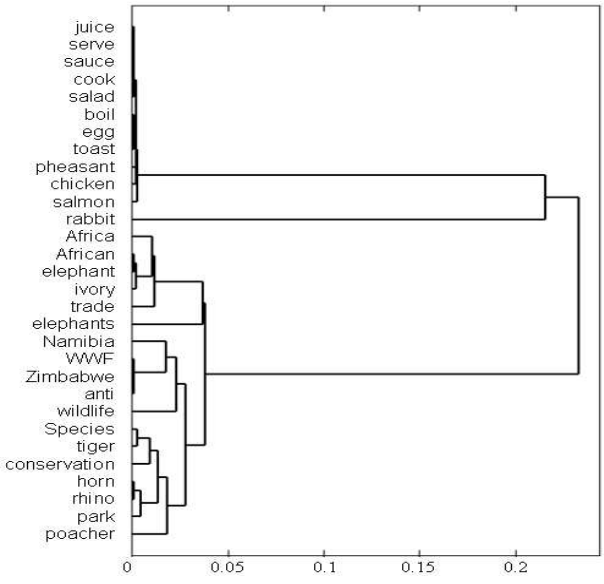

Figure 3: Clustering results for *poach* with SVD.