

Mining Text for Word Senses Using Independent Component Analysis

Reinhard Rapp*

Abstract

The assumption that the problem of ambiguity in text analysis can only be solved if statistical dependencies of higher than second order are considered leads us to independent component analysis (ICA), a statistical formalism that takes higher-order dependencies into account. By assuming independence, ICA is capable of detecting a set of hidden vectors if only different linear mixtures of these vectors are observable. As a test case for ICA's applicability to natural language processing we look at the task of word sense induction. Our starting point is that we consider the co-occurrence vector of an ambiguous word as a linear mixture of its unknown sense vectors. If corpora from different domains are available, this should give us the different linear mixtures that are required for ICA. It turns out that the independent sense vectors derived by ICA from the distributional differences of word usage reflect a word's meanings surprisingly well.

1 Introduction

Many problems in natural language processing can be successfully approached by using methods that rely on feature vectors [1]. Since entities in natural language tend to be ambiguous, the feature vectors that we derive from natural language can be assumed to be mixtures of the vectors of some underlying unambiguous entities. The problem in understanding and simulating natural language is that we can only observe and study the complicated behavior of the ambiguous entities, whereas the presumably simpler behavior of the underlying unambiguous entities remains hidden.

To be more concrete, let us look at word meaning. In this case, the ambiguous entities we consider are words, the unambiguous entities are their senses, and as the relevant feature vectors their co-occurrence vectors can be used. Using co-occurrence vectors is appropriate as it has been shown that the meanings of a word are well reflected in its lexical neighborhood, that is, the neighbors of a word can be considered to be its features [2].

Past work on word senses has concentrated on disambiguation, that is, on choosing among a predefined set of senses when given an ambiguous word in context. However, very little work has been done on sense induction, that is, the automatic discovery of the possible senses for a word. One of the first attempts was made by Arns [3] who clustered human associations to homographs and hoped to obtain clusters corresponding to each meaning. The clustering algorithm used in this early work relied on a first- instead of a second-order association measure, which is probably the main reason why the results were not good enough for practical purposes.

A recent attempt to sense induction was made by Pantel and Lin [4]. They clustered the words in a large corpus using a clustering algorithm relying on a mutual information-based distance measure. Since 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. One problem with this approach is that it allows only as many senses as clusters, thereby adversely affecting the granularity of the meaning space.

The approaches of Arns [3] and Pantel and Lin [4] both consider the underlying large corpus as homogeneous, that is, they do not exploit information derived from the observation that ambiguous words are often used differently in texts from different domains. In other words, neither method takes the "one-sense-per-discourse" criterion into account, which has been shown to be effective in word sense disambiguation [5].

This consideration is the starting point for the research presented in this paper. We assume that it is likely that the senses of an ambiguous word are distributed unevenly in texts from different domains. For example, in economic texts the "factory" meaning of *plant* may prevail, whereas in biological texts the meaning in the sense of "living organism" will probably be more frequent. Since the context words usually reflect the senses, words like *chemical* or *power* will be frequent neighbors of *plant* in economic texts, and words like *living* or *seed* will be frequent neighbors in biological texts. Our assumption is now that those words that are characteristic of the senses of a given ambiguous word have the greatest variation in their co-occurrence counts when comparing counts derived from corpora of different domains.

However, to be able to decide that *chemical* and *power* belong to the same sense, but that *chemical* and *living* belong to different senses of *plant*, we also need to look at the directions of variation. We find that in those corpora where *chemical* co-occurs frequently with *plant*, *power* also has a high co-occurrence frequency with *plant*, and that in those other corpora where *living* can often be observed together with *plant*, *seed* is also likely to co-occur frequently with it. That is, those words that show a high covariance belong to the same sense. If we take the average vector of the words with high covariance, we get the position in semantic space that is characteristic for a sense of the ambiguous word.

To implement this we could have developed an algorithm from scratch. However, in the form of ICA a powerful statistical formalism that has the required capabilities already exists. It has the advantage that it is not necessary for each of the senses to be discovered to actually predominate in one of the corpora considered. Instead, little variations in sense distributions can be exploited to find the senses. So even if the main sense of a word prevails in all corpora, this does not mean that the other senses cannot be found.

* University of Mainz, rapp@mail.fask.uni-mainz.de.
This research was supported by the DFG.

The outline of this paper is as follows: We first give a short overview of ICA. Next we describe our method for sense induction and present some results. We then discuss the approach and conclude with an outlook.

2 Independent component analysis

As ICA is a relatively new statistical technique, to our knowledge it has not often been used in computational linguistics so far. The main application of which we are currently aware is in text classification [6]. However, since – other than the better known *support vector machines* [7] – ICA allows to solve problems that cannot be approached with conventional methods, we predict that this will change rapidly.

The starting point for using ICA here was our conclusion from previous research that the problem of ambiguity in natural language can only be solved if statistical dependencies of higher than second order are considered. This is a capability that ICA offers. It can be seen as an extension of *principal component analysis* (PCA)¹ towards higher order dependencies. Since PCA can only deal with second-order dependencies [10], the problem with it is ambiguity; ICA promises to solve this problem.

An in-depth treatment of ICA is the subject of an entire book [10]; therefore, we can only give a rough overview here. ICA was developed for signal processing to solve the problem of *blind source separation* (BSS). The aim of BSS is the recovery of a number of source signals when only mixtures of these signals are available. A problem of this kind occurs, for example, at a cocktail party, when many people speak at the same time, but a listener wants to concentrate on a single voice. Other applications include measuring electrical brain activity when electrodes fastened to the head can only record signal mixes, or in stock market prediction when the factors underlying stock price changes (e.g. interest rates, company earnings, currency exchange rates, etc.) are to be detected.

Let us look at an instructive example: Figure 1(a) shows two signals, s_1 and s_2 , each consisting of a number of pulses. In figure 1(b) we see two mixes of these signals. The mixing factors can be arbitrary. In this case the upper signal was computed as $1.5 s_1 + s_2$, the lower one as $s_1 + 2 s_2$. The mixed signals were processed with the *FastICA* algorithm described in section 3.5. The *independent components* computed by the algorithm are given in figure 1(c). Although only the mixed signals were available to the program, by maximizing the statistical independence between signals² it was possible to approximately recover the original signals. In this example, the process of blind source separation can be thought of as iteratively maximizing the discrep-

ancies between the mixed signals in such a way that the pulses with the same direction of variation (high covariance) are assigned to the same signal.³

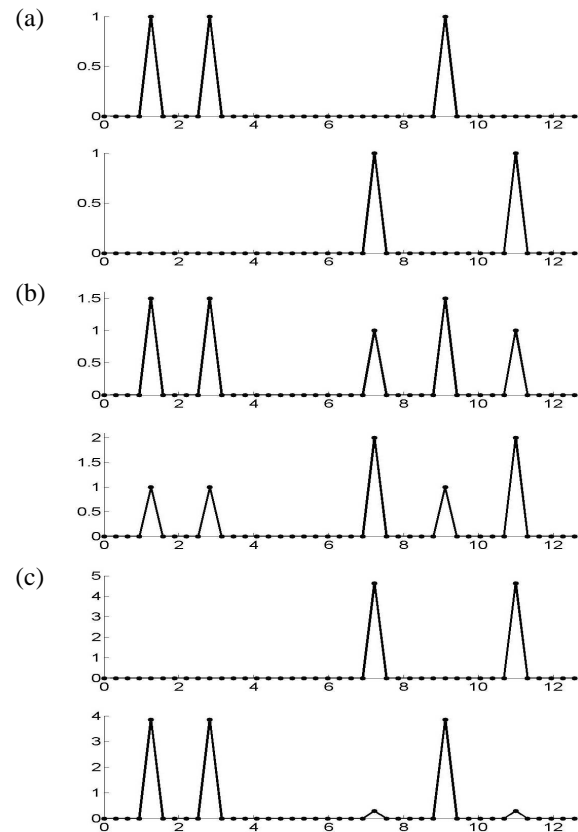


Fig. 1: Signal separation using ICA.

However, it is only in our instructive example that it is so easy to see where the covariances are. ICA offers a general solution to this problem that works for signals of any shape if the condition of independence is (approximately) fulfilled,⁴ and if there are at least as many different mixed signals available as there are independent components to be detected. In cases where it is not possible to find perfectly independent signals, ICA tries to approximate independence.

As what we called “signals” in figure 1 are actually vectors of 41 values each (indicated by the little dots in the curves), this approach not only works for time-dependent signals but also for vectors of any kind. What we present in this paper is the application of ICA to vectors of word co-occurrence.

3 Word sense induction

3.1 Corpora

For word-sense disambiguation or induction it is advisable to use large corpora since all (main) senses of each word of

¹ References in the computational linguistics literature are usually to *singular value decomposition* (SVD), which is a method similar to PCA that can be applied to non-square matrices [8]. The use of SVD in information retrieval is called *latent semantic indexing* (LSI, see [9])

² Two signals are independent of each other if – given one of the signals – it is impossible to make any predictions on the shape of the other signal. A perfect example of this is random signals (noise). A counterexample is signals of the same frequency. They are generally not independent as identical correlation patterns repeat over and over again.

³ Please note that this is exactly what we need for sense induction. Recall the example with *plant* in the introduction.

⁴ In particular, the signals need not be continuous for ICA to work. Actually, the best separation is achieved with random signals (noise), as they perfectly fulfill the condition of independence.

interest should be represented. Also, for our method we need corpora of different domains. These were the three corpora chosen:

- *British National Corpus* (BNC, approx. 100 million words)
- Newspaper *The Guardian*, years 1992 to 1994 (approx. 95 million words)
- Scientific and technical abstracts from the psychological database *PsycLIT* (1988/89) and from the American *Department of Energy* (DOE). The DOE abstracts were taken from the CD-ROM 1 of the Association for Computational Linguistic's Data Collection Initiative (about 30 million words from each of the two sources).

Of course these corpora, especially the BNC, are not at all pure in the sense that they contain texts from only a single domain. However, because they are certainly different mixes, we thought they would serve the purpose of having different distributions of word senses.

Since function words were judged irrelevant for our semantic considerations, to save disk space and processing time we decided to remove them from the corpora. This was done on the basis of a list of approximately 200 English function words. Not so much for computational reasons, but mainly to reduce the sparse data problem, we also lemmatized the corpora using a lexicon of full forms (for details see [11]). Since most word forms are unambiguous concerning their possible lemmas, we conducted only a partial lemmatization process that leaves words with several possible lemmas unchanged.⁵

3.2 Evaluation data

To evaluate the results we took the list of 12 ambiguous words used by Yarowsky [5]. Each of these words is considered to have two main senses, and for each sense a word characteristic of that sense is provided. Table 1 shows the list of words together with their sense descriptors.

Table 1: List of 12 ambiguous words and their senses.

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

⁵ Since a context-sensitive lemmatizer requires sense disambiguation, which in turn has sense induction as a prerequisite, using such a lemmatizer in a pre-processing step would mean anticipating the purpose of the whole paper, namely, sense induction.

3.3 Association matrices

For counting word co-occurrences, as in most other studies a fixed window size is chosen and it is determined how often each pair of words occurs within a text window of this size. Choosing a window size usually means a trade-off between two parameters: specificity versus the sparse-data problem. The smaller the window, the more salient the associative relations between the words inside the window, but the more severe the sparse-data problem. In our case, with ± 2 words, the window size looks rather small. However, this can be justified since we have reduced the effects of the sparse-data problem by using rather large corpora and by lemmatizing the corpora. It also should be noted that a window size of ± 2 applied after elimination of the function words is comparable to a window size of ± 4 applied to the original texts (assuming that roughly every second word is a function word).

Based on the window size of ± 2 , we computed co-occurrence matrices for each of the three corpora as well as for a concatenated corpus comprising all three. By storing them as sparse matrices, it was feasible to include all of the approximately 700 000 lemmas occurring in the concatenated corpus.

To eliminate word-frequency-effects, the log-likelihood test was applied to all values in the matrices. The idea is to emphasize significant and to weaken incidental word pairs by comparing their observed co-occurrence counts with their expected co-occurrence counts. We chose the log-likelihood test instead of the better known χ^2 test because Dunning [12] showed that the log-likelihood test is more appropriate for sparse data.

For efficient computation of the log-likelihood ratio we used the following formula [11]:⁶

$$\begin{aligned}
 -2 \log \lambda &= \sum_{i,j \in \{1,2\}} k_{ij} \log \frac{k_{ij} N}{C_i R_j} \\
 &= k_{11} \log \frac{k_{11} N}{C_1 R_1} + k_{12} \log \frac{k_{12} N}{C_1 R_2} \\
 &\quad + k_{21} \log \frac{k_{21} N}{C_2 R_1} + k_{22} \log \frac{k_{22} N}{C_2 R_2}
 \end{aligned}$$

where

$$\begin{aligned}
 C_1 &= k_{11} + k_{12} & C_2 &= k_{21} + k_{22} \\
 R_1 &= k_{11} + k_{21} & R_2 &= k_{12} + k_{22} \\
 N &= k_{11} + k_{12} + k_{21} + k_{22}
 \end{aligned}$$

Parameters k_{ij} can be expressed in terms of corpus frequencies as follows:

$$\begin{aligned}
 k_{11} &= \text{frequency of common occurrence of word } A \\
 &\quad \text{and word } B \\
 k_{12} &= \text{corpus frequency of word } A - k_{11} \\
 k_{21} &= \text{corpus frequency of word } B - k_{11} \\
 k_{22} &= \text{size of corpus (no. of tokens) - corpus} \\
 &\quad \text{frequency of } A - \text{corpus frequency of } B
 \end{aligned}$$

Due to the different sizes of the corpora, after transforming the four matrices, all vectors (lines in the matrices) were

⁶ In cases where the argument of a logarithm was zero we replaced it by a small constant.

normalized in such a way that for each vector the sum of its entries was one. In the remainder of the paper, we refer to the transformed and normalized matrices or vectors as association matrices or association vectors.

3.4 Vector similarities

For measuring semantic similarity we use the city block metric, which computes the distance between two association vectors X and Y as the sum of the absolute differences of corresponding vector positions:

$$s = \sum_{i=1}^n |X_i - Y_i|$$

Since we are using normalized vectors, more sophisticated and computationally more demanding similarity measures (e.g. the cosine coefficient) cannot be expected to yield substantially better estimates.⁷

3.5 FastICA algorithm

For independent component analysis the FastICA algorithm was used with default parameters. It is implemented in the MATLAB (MATrix LABoratory) programming language and is kindly made available by the developers from their website at <http://www.cis.hut.fi/projects/ica/fastica>. A description of the algorithm is given in [10].

4 Results

For each of the 12 ambiguous words shown in table 1 three association vectors were extracted from the association matrices corresponding to each of the three sub-corpora. With the aim of obtaining the two main senses for each word, by applying the FastICA algorithm to every vector triplet, two independent components were computed.⁸

To be able to describe the meaning of the independent components by their closest neighbors in semantic space, each of them was compared to all vectors in the association matrix that had been derived from the concatenated corpus. Although the words showing the highest similarity were generally very specific, to suppress uncommon terms, all words with a corpus frequency of less than 100 in the concatenated corpus were eliminated.⁹ Among the remaining

words the top three showing the highest similarity were selected as being suitable to characterize the meaning of the respective independent component.

Table 2: Computed senses for 12 ambiguous words.

axes grid / tools		bass fish / music	
machete	axis	catfish	guitar
axe	crystallographic	sturgeon	saxophone
crowbar	orthogonal	minnow	Solo
crane bird / machine		drug medicine / narcotic	
manipulator	winch	tranquilizer	medication
monorail	truck	heroin	polydrug
equipment	bulldozer	cocaine	inhalant
duty tax / obligation		motion legal / physical	
obligation	motorcycles	velocity	amendment
Customs	drinker	nonlinear	vote
responsibility	snowfall	trajectory	MPs
palm tree / hand		plant living / factory	
shrub	cupping	hydroelectric	shrub
trunks	cupped	Bonneville	tree
falling	outstretch	PowerGen	flower
poach steal / boil		sake benefit / drink	
boil	tab	almighty	brevity
omelette	LSP	bless	cheapness
yolk	app	crucify	meaningful
space volume / outer		tank vehicle / container	
Nasa	correspond	retrievable	artillery
astronaut	dimensional	SMES	howitzer
laboured	surface	pressurizer	infantry

For each of the computed independent components these words are shown in table 2. Although the results leave room for individual interpretation, let us discuss our judgment. In our view, the results for the words *axes*, *bass*, *drug*, *motion*, *plant*, and *space* well reflect the senses suggested by Yarowsky. To a lesser degree this is also true for *palm* and *tank*, especially since the abbreviation *SMES* occurring in the DOE abstracts stands for *superconducting magnetic energy storage*. In the cases of *duty* and *sake* the distinctions made by the program, although different from those of Yarowsky, nevertheless show some plausibility. For *duty* the computed senses are related to *obligation* and *heavy duty*; for *sake* the uses *sake of God* versus *sake of clarity* are distinguished. In the case of *sake* the reason for not finding both of Yarowsky's senses seems pretty obvious, since the beverage sense of *sake* rarely occurs in our corpora.

For the remaining two ambiguous words, namely *crane* and *poach*, only one of Yarowsky's senses was found with

⁷ Our results based on log-likelihood-transformed data were consistently slightly better when using the city block metric instead of the cosine coefficient. However, in other experiments that used entropy-based weights and matrices whose dimensionality had been reduced by applying a singular value decomposition, the cosine coefficient performed almost always somewhat better.

⁸ Since FastICA is an iterative algorithm that starts with a random initialization, depending on the data it can run into local minima that – although showing certain patterns – may be different from run to run. For example, if we compute two independent components, but there are three main senses of the ambiguous word, then in one run we may obtain independent components relating to senses 1 and 2, in another run independent components relating to senses 1 and 3, and in a third run independent components relating to senses 2 and 3. Although it may improve results, in the present research we made no attempt to evaluate several local minima and to filter out the best one.

⁹ Alternatively, candidate selection could also be based on association strength to the ambiguous word.

the other sense belonging either to the same category (*crane*) or seeming inappropriate to us (*poach*).

Given our subjective judgment, we obtain for our admittedly very small sample of 12 words the following quantitative results: In about 67% of the cases the expected result was obtained, in almost 17% of the cases we obtained a result different from the expectation that seemed nevertheless plausible, and in another 17% of the cases we obtained an erroneous result.

5 Discussion, conclusions, and prospects

These results indicate that by analyzing differences in the use of ambiguous words in texts from different domains ICA is capable of inducing word senses. Although we did not take syntax into account, our system was able to predict plausible sense descriptors for the majority of the test words.

One might argue that the method is not generally applicable since a variation of sense distribution may not be observable for all ambiguous words. However, we believe that this is only a question of how fine the domain distinctions are. For example, we could simply consider each document in a corpus as belonging to a separate domain thus providing a much richer input for ICA. From another point of view, one can consider such an approach a new realization of the “one-sense-per-discourse” observation formulated by Yarowsky [5].

The reason why we were unable to implement this so far is the sparse-data problem. The sampling error for short documents is so high that ICA is not capable of making any sense out of it. This becomes clear when one considers that ICA has no other chance than to derive a relationship between two association vectors from the words they have in common, but that short documents usually have little overlap in their content words. This means that vectors based on such documents are already independent of each other so that ICA leads to nothing.

However, we see a possible solution to this problem. If by performing a singular value decomposition (SVD; see [8] and [9]) we reduce the number of columns in our matrices from about 700 000 to typically 300, we will get such a dramatic increase in the overlap between vectors that sense induction by ICA may work even for relatively short documents (whose vectors can be easily folded to the reduced space; see [8]).

This application would give new importance to SVD. Whereas so far in the context of text analysis SVD has often been considered as just another (complicated and computationally expensive) smoothing method that only slightly improves the computation of co-occurrence-based word similarities, it may become invaluable for providing input to ICA.¹⁰

¹⁰ However, preliminary experiments indicate that ICA may not lead to satisfactory results when applied to dimensionality-reduced co-occurrence data. Should this finding be confirmed, we suggest an alternative approach for word sense induction based on clustering the local contexts of a word. The outline of an algorithm is as follows: A term/document-matrix is constructed whose columns correspond to all documents of a corpus that contain a given ambiguous word and whose lines correspond to those words in the documents that are strongly associated to the given word. Let us assume

Our vision for the future is that in our association matrices SVD gives us the principal components of the columns whereas ICA gives us the independent components of the rows. Together they provide a mathematically sound framework for an optimal clustering of the semantic space. Thus SVD and ICA move us a step forward in our attempts to solve the problem of ambiguity in natural language processing in a psychologically plausible way [9].

Acknowledgements

I would like to thank Manfred Wettler, Erkki Oja, Aapo Hyvärinen, Hugo Gävert, Jarmo Hurri, and Jaakko Särelä.

References

- [1] R. Rapp. Unsupervised learning of second order dependencies from corpora. In: *Tagungsband der 6. KONVENS*, DFKI, Saarbrücken, 155–162, 2002.
- [2] H. Schütze. *Ambiguity Resolution in Language Learning: Computational and Cognitive Models*. Stanford: CSLI Publications, 1997.
- [3] U. Arns. *Sprachstatistische Analysen lexikalischer Mehrdeutigkeiten*. Diplomarbeit an der Universität-GH Paderborn, Fachbereich Psychologie, 1994.
- [4] P. Pantel, D. Lin. Discovering Word Senses from Text. In: *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Edmonton, 613–619, 2002.
- [5] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In: *Proc. of the 33rd Annual Meeting of the ACL*, Cambridge, 189–196, 1995.
- [6] M.A. Girolami. Latent class and trait models for data classification and visualisation. In: S. Roberts, R. Everson (eds.): *Independent Component Analysis. Principles and Practice*. Cambridge Univ. Press, 254–279, 2001.
- [7] M.A. Hearst. Trends & controversies: Support vector machines. *IEEE Intelligent Systems*, 13(4), 18–28, 1998.
- [8] C.D. Manning, H. Schütze. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press, 1999.
- [9] T.K. Landauer, S.T. Dumais. A solution to Plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2), 211–240, 1997.
- [10] A. Hyvärinen, J. Karhunen, E. Oja. *Independent Component Analysis*. New York: Wiley, 2001.
- [11] R. Rapp. Automatic identification of word translations from unrelated English and German corpora. In: *Proceedings of the 37th Annual Meeting of the ACL*, College Park, Maryland, 519–526, 1999.
- [12] T. Dunning. Accurate methods for the statistics of surprise and coincidence. In: *Computational Linguistics*, 19(1), 61–74, 1993.

that we want to compute the two main senses of the given word. We accomplish this by applying a SVD in such a way that the number of columns in the matrix is reduced to two. This operation has an effect similar to optimally clustering the documents into two thematic classes, each corresponding to one of the two main senses of the given word. The maxima in the two resulting column vectors correspond to those words that are suitable to be used as sense descriptors.