Independent Component Analysis of Word Contexts and Comparison with Traditional Categories

Jaakko Väyrynen[1†], Timo Honkela[1‡], and Aapo Hyvärinen[2]

[1]Helsinki University of Technology
Neural Networks Research Center
Laboratory of Computer and Information Science
[†]jaakko.j.vayrynen@hut.fi
[‡]timo.honkela@hut.fi

[2]Helsinki Institute of Information Technology
Basic Research Unit
Department of Computer Science
University of Helsinki
aapo.hyvarinen@helsinki.fi

ABSTRACT

We study written language as if it were a multidimensional signal rather than a stream of symbols. We show that it is possible to find emergent features by independent component analysis from word contexts. The closeness of match between the learned features and traditional linguistic word categories is examined. It is shown that independent component analysis performs better than principle component analysis.

1. INTRODUCTION

Contextual information has widely been used in statistical analysis of natural language corpora. For instance, latent semantic analysis (LSA) finds latent concepts, which enables the analysis of the latent concepts and comparing documents for instance in information retrieval [1, 2]. A self-organizing map (SOM) [3] taught on contextual information reflects implicit semantic and syntactic categories of words [4, 5]. Clustering of words based on contextual information reveals syntactic and semantic clusters [6, 7, 8]. With independent component analysis (ICA) [9, 10] emerging explicit features reflect syntactic and semantic categories [11]. The reduced vector representation for words based on the ICA analysis can be applied to various applications.

In this paper, we examine the emerging features found from words in their contexts by independent component analysis (see also [11] in which no comparison to traditional categories was made). For comparison, we performed a similar study on features found by singular value decomposition (SVD) as a baseline method. Performing principle component analysis (PCA) is the equivalent of performing SVD on the covariance matrix of the data.

In linguistics, words are categorized by syntactic features such as noun, verb, plural, past tense, etc. A word can have several features, e.g. the verb *goes* is in present tense and in 3rd person. Two words belong to the same syntactic category if one can be replaced by the other without affecting the grammaticality of the sentence. This is called the replacement test.

In this article, we compare the closeness of match between the learned features, i.e, the independent components, and the manually determined syntactic word categories. First we present the methods and data that we have used in the experiments, and finally show the experimental results and discuss their status. In general, our approach is based on the idea that even written language can be processed like a multidimensional signal rather than as a stream of symbols.

2. METHODS

In the following, we present shortly the basics of independent component analysis and analysis of word contexts.

2.1. Independent component analysis

Independent component analysis is a latent variable model, where the random observation vector $\mathbf{x}$ with components $x_1, \ldots, x_n$ is assumed to be generated as a linear mixture of latent sources $s_1, \ldots, s_n$, denoted by a random vector $\mathbf{s}$, weighted by the rows of the mixing matrix $\mathbf{A}$. In matrix notation the mixing model is

$$\mathbf{x} = \mathbf{A}\mathbf{s} = \sum_{i=1}^{n} \mathbf{a}_i s_i \tag{1}$$

where $\mathbf{a}_i$ are the columns of the mixing matrix $\mathbf{A}$. The components s_i are assumed to be independent and nongaussian. Given some observations $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, the goal is to estimate both the mixing matrix $\mathbf{A}$ and the sources $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_n)^T$.

We used the FastICA [12] package for Matlab, that implements a fast fixed-point algorithm for ICA.

2.2. Analysis of word contexts

In order to obtain a meaningful numerical representation for the words in the texts we take into account the sentential context in which the words occur. First, we represent each word by a vector in an n-dimensional space, and then code each context as an average of vectors representing the words in that context. In the simplest case, the dimension K can be taken equal to the number of different words, and each word is represented by a vector with one element equal to one and others equal to zero. Then the context vector simply gives the frequency of each word in the context.

In our experiments, we calculated histograms of words w_i in different contexts c_j. The histograms can be seen as unnormalized conditional probability distributions $P(w_i|c_j)$. A context c_j can be defined in terms of the neighbor of the target word w_i. A single left-side context would calculate the number of pairs of context words and a target word that are adjacent in the text, i.e., bigrams. The histograms can be created, for instance, by assigning initially zero vectors $\mathbf{c}_j$ for contexts, and then counting the instances of target word w_i being adjacent to the context word w_{c_j} and storing the result in the vector $\mathbf{c}_j$ in position i.

A context could also use syntactic or semantic information of the text, for instance the sentence structure. In this paper, we consider only simple word co-occurrences, n-grams, that can be calculated efficiently with the CMU Language Modeling Toolkit [13].

The histograms of words w_i in contexts c_j are combined into matrix $\mathbf{C} = C_{ij} = \#\{w_i|c_j\}$. The columns $\mathbf{c}_j$ are the histograms over words in context c_j, and the rows tell how the word w_i occurs in contexts.

Independent component analysis is applied to the histograms in order to explain the context histograms as a weighted sum of learned features. The generative model of Eq. 1 is illustrated in Fig 1.

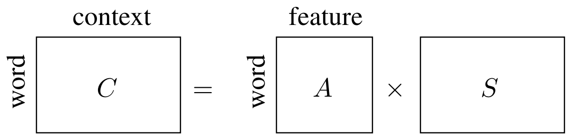

Fig. 1. The histograms of the word-context matrix can be constructed as a linear mixture of features.

3. CORPORA

In our study, we used two corpora, Gutenberg corpus and Brown corpus. The latter is a tagged corpus, i.e., each word is tagged manually with a traditional linguistic category.

3.1. Gutenberg Corpus

The context histograms were calculated from the Gutenberg corpus of electronic texts[1]. After preprocessing the corpus consists of over 21 million tokens in running text and 188,386 types, i.e, different word forms. The preprocessing consisted of selecting English texts, removing portions related to project identification of the texts and removing most of the non-alphabetical characters.

3.2. Traditional linguistic categories

The manual syntactic categorization for words was extracted from a 300k word subset of the tagged Brown corpus. For each category tag t, the words w_i that were assigned to it were collected into a vector $\mathbf{l}_t$, where one in position i meant that word w_i had been assigned the tag in the Brown corpus, and zero that it had not.

4. EXPERIMENTS

The context histograms $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_N)$ were calculated using the most frequent thousand words in the Gutenberg corpus as the target words w_i. The words were filtered to include only words that were also present in the Brown corpus. The context words were chosen to be the most frequent words in the Gutenberg corpus. In our experiments, only left word contexts were used, so that the single context word and the target word were consecutive with no words in between them.

Large differences between the raw frequency counts were lowered by taking the logarithm of the frequencies added by one. For more extensive experiments a more elaborate global weighting might be necessary.

As a preprocessing step to ICA, the context histograms were whitened with PCA. Dimension was reduced simultaneously to equal the number of estimated sources K, which was usually around 50.

[1] `http://gutenberg.net`

FastICA was applied to the whitened context histograms to extract the features $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_k)$ and the sources $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_k)^T$. The symmetric approach was used with a skewed tanh nonlinearity

$$g(u) = \begin{cases} \tanh(u), & u < 0 \\ a \tanh(u), & u \geq 0 \end{cases} \tag{2}$$

with $a = 4$. The added skewness in the nonlinearity reflects the non-negativity of the sources. The found sources were further forced to have positive skewness by multiplying the source and the corresponding feature by -1, if the source had negative skewness. This can be done because the ICA model cannot learn the signs of the sources.

The extracted features $\mathbf{a}_k$ can be interpreted as a grouping of words, that occur in similar contexts. This relates to the replacement test of testing whether two words belong to the same category. Here the test is statistical: if two words occur in similar contexts, they are assigned to the same category. The learned features $\mathbf{a}_k$ are interpreted as the learned categories, where the magnitude of the component i relates to the degree of membership for word w_i.

We will examine the closeness of match between the syntactic categories and the learned features.

4.1. Correlation measure

To compare the learned features $\mathbf{a}_k$ and the traditional syntactic categories $\mathbf{l}_t$ a normalized correlation

$$M_{kt} = \frac{\mathbf{a}_k^T \mathbf{l}_t}{\|\mathbf{a}_k\| \|\mathbf{l}_t\|} \tag{3}$$

was calculated between all features and categories. The higher the correlation M_{kt} is, the better match there is between the feature $\mathbf{a}_k$ and the category $\mathbf{l}_t$.

An example feature found with ICA is shown in Fig. 2. It is the best feature for plural nouns. The feature has a high correlation with plural nouns and adjectives. Plural nouns clearly dominate the highest component values with adjectives following. Other words have component values near zero.

Fig. 3 shows the maximum correlation $\max_k M_{kt}$ for each category t over features $\mathbf{a}_k$ estimated with ICA. The data and preprocessing were as explained, with the context matrix $\mathbf{C}$ of size 1000×1000 was calculated with single left word context. The maximum correlations were calculated from the estimated 50 features, and the process was repeated five times. Mean and one standard deviation is shown in the y-axis.

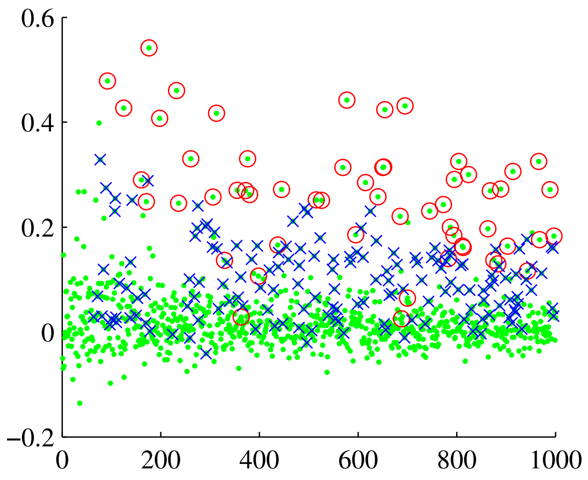

Fig. 2. An example feature found with ICA with the component values for words w_i (dot) in the y-axis. The feature has a high correlation with plural nouns (circle) and adjectives (cross).

Fig. 4 shows the results for features calculated using singular value decomposition with the same data and parameters. The features correspond to the dimension reduction done as a preprocessing step to ICA. A comparison with the results in Fig. 3 shows that independent component analysis performs better in learning traditional syntactic categories than principal component analysis. In the average, the features found by ICA have a better match with linguistic syntactic categories than the features found by PCA, when the feature having the highest correlation with the category is selected as the best matching feature. Similar experiments were conducted with varying number of components K and different contexts, and the results were similar.

5. CONCLUSIONS

We have shown how independent component analysis can automatically, without human supervision, find explicit linguistic features of words. Independent component analysis appears to make possible a qualitatively new kind of result which have earlier been obtainable only through handmade analysis. We also showed that independent component analysis performs better in learning traditional syntactic categories than principal component analysis. The experiments were conducted with the choice of different contexts and a varying number of components and different contexts with similar results.

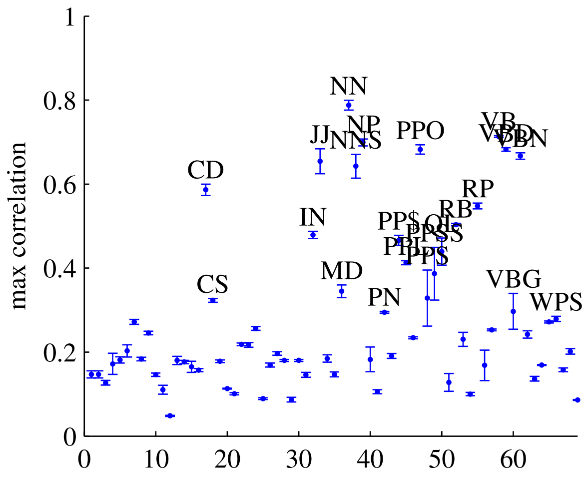

Fig. 3. Maximum correlation $\max_k M_{kt}$ for each Brown tag t calculated from 50 features that were estimated with ICA. The sources were forced to have positive skewness and the resulting very low negative correlations were not considered. For illustration purposes only some of the tag names are shown. The higher the correlation is, the better match there is between the best feature and the syntactic category. The shown mean and one standard deviation of the maximum correlation for each tag were calculated using five feature sets that were estimated starting from random initialization. Context histogram matrix $\mathbf{C}$ was of size 1000×1000 using single left word context, and 50 features were calculated.

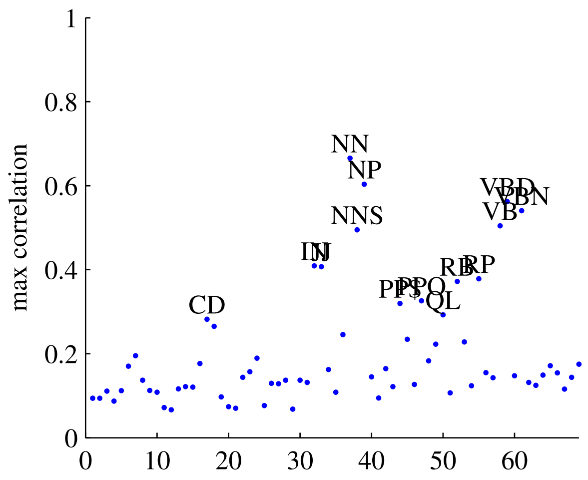

Fig. 4. Maximum correlation $\max_k M_{kt}$ for each Brown tag t for 50 features calculated with SVD. The data and preprocessing are the same as in the results of Fig. 3.

6. REFERENCES

[1] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, vol. 41, no. 6, pp. 391–407, 1990. Available: citeseer.nj.nec.com/deerwester90indexing.html

[2] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse Processes*, vol. 25, pp. 259–284, 1998.

[3] T. Kohonen, *Self-Organizing Maps*. Berlin, Heidelberg: Springer, 2001.

[4] H. Ritter and T. Kohonen, "Self-organizing semantic maps," *Biological Cybernetics*, vol. 61, pp. 241–254, 1989.

[5] T. Honkela, V. Pulkki, and T. Kohonen, "Contextual relations of words in Grimm tales analyzed by self-organizing map," in *Proc. ICANN-95, International Conference on Artificial Neural Networks*, vol. 2. Nanterre, France: EC2, 1995, pp. 3–7.

[6] F. Pereira, N. Tishby, and L. Lee, "Distributional clustering of English words," in *Proc. the 31st annual meeting of the Association for Computational Linguistics*, 1993.

[7] A. Clark, "Inducing syntactic categories by context distribution clustering," in *Proc. CoNLL-2000 and LLL-2000*, Lisbon, Portugal, 2000, pp. 91–94.

[8] J. P. Levy and J. A. Bullinaria, "Learning lexical properties from word usage patterns: Which context words should be used?" in *Development and Evolution: Proceedings of the Sixth Neural Computation and Psychology Workshop*. London: Springer, 2001, pp. 273–282.

[9] C. Jutten and J. Hérault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, pp. 1–10, 1991.

[10] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. John Wiley & Sons, 2001.

[11] T. Honkela, A. Hyvärinen, and J. Väyrynen, "Emergence of linguistic representation by independent component analysis," Helsinki University of Technology, Laboratory of Computer and Information Science, Tech. Rep. A72, 2003.

[12] "The FastICA MATLAB package." Available: http://www.cis.hut.fi/projects/ica/fastica/

[13] P. Clarkson and R. Rosenfeld, "Statistical language modeling using the CMU-cambridge toolkit," in *Proc. Eurospeech '97*, Rhodes, Greece, 1997, pp. 2707–2710.