COMPARISON OF INDEPENDENT COMPONENT ANALYSIS AND SINGULAR VALUE DECOMPOSITION IN WORD CONTEXT ANALYSIS

Jaakko J. Väyrynen and Timo Honkela

Neural Networks Research Centre, Helsinki University of Technology,
P.O. Box 5400, FI-02015 TKK, FINLAND,
{jaakko.j.vayrynen, timo.honkela}@hut.fi

ABSTRACT

In earlier studies we have been able show that Independent Component Analysis is able to extract automatically meaningful linguistic features. The emergent syntactic and semantic features are based on an analysis of the words in their contexts in a large corpus. We have also shown that there is a reasonably strong correlation between traditional features and categories defined by linguists and the emergent features. In this article, we introduce a new measure for comparing the emergent and the traditionally defined features. We apply this measure to compare the emergent features produced by Singular Value Decomposition (SVD) and Independent Component Analysis (ICA). The conclusion is that the ICA-based features correspond to the human intuitions much more closely than the SVD-based features not only in a visual inspection but also in a systematic and principled comparison.

1. INTRODUCTION

Earlier we have shown how the Independent Component Analysis can automatically extract meaningful linguistic features [1, 2]. We have processed the contextual information in a commonly used manner that has earlier been presented, e.g., by [3, 4, 5, 6]. Earlier research in which ICA has been used in analyzing text data includes [7, 8].

We have also shown that there is a reasonably strong correlation between the traditional categories defined by linguists and the emergent features [9, 10]. In [9], among the ICA-based features the one that had the closest match with the traditional category was first selected. In the study, 1000 most common words for which the traditional linguistic category was known were considered. The comparison was made using the Brown corpus. A successful match between the ICA-based feature and the traditional category is rather apparent. However, we still wished to develop a more accurate method for comparisons. That kind of quantitative measure we present in this paper.

In this article, we introduce a new measure for comparing the emergent and the traditionally defined features. We apply this separation measure to compare the emergent features produced by Singular Value Decomposition (SVD) and Independent Component Analysis (ICA).

The method and the data used in the comparison are presented in the next section. Thereafter we show the comparison results that clearly indicate that the ICA-based features correspond to the human intuitions much more closely than the SVD-based features. One important potential consequence is that as ICA may provide a better model for emergent semantic representations than SVD, this result can be generalized into all uses of SVD in Latent Semantic Analysis (LSA) and related approaches.

2. METHODS AND DATA

We will briefly give an overview of the mathematical methods we have used, namely Independent Component Analysis and Singular Value Decomposition, and explain how they differ as feature extraction techniques. The text corpus used as a source for statistical information is mentioned, as well as the source for linguistic information of word categories. A novel method for quantitative analysis between the extracted features and linguistic word categories is introduced.

2.1. Data

We have used the same English corpus of a collection of different texts from Project Gutenberg[1] as in [9, 10]. Also the preprocessing was conducted in a standard manner. In summary, most of the non-alphanumeric characters were removed and the remaining characters were converted to lowercase. The resulting corpus consisted of 21,951,835 instances of words (tokens) with 188,386 unique words (types). The word category information was extracted from a subset of the tagged Brown corpus[2] that had a single word category tag t_k assigned to each word instance. We collected the possible tags for each unique word. See [9] and [10] for a more thorough explanation.

The collection of contextual data differed slightly from [1, 2, 9, 10]. Instead of collecting a word-context matrix, we collected a context-word matrix. This means that we were assuming independence of the contexts and not the analyzed words when applying independent component analysis. This is analogous to the temporal versus spatial domain analysis with fMRI data [11].

For the analysis, the $N = 10000$ most common words from the Gutenberg corpus were selected as the vocabulary to be analyzed. Additionally, the selected words had

[1] Available on-line at `http://www.gutenberg.org`

[2] Available on-line at `http://www.ldc.upenn.edu`

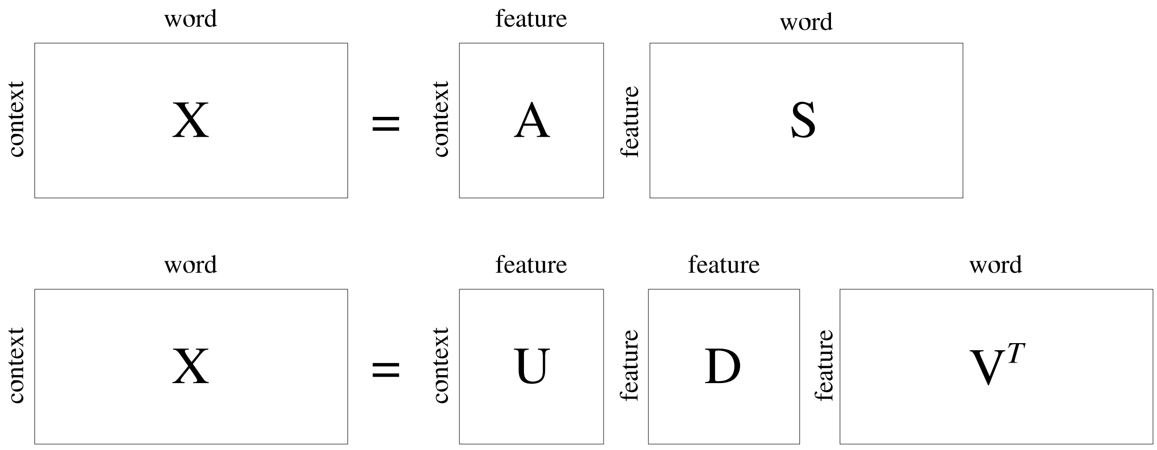


Figure 1. Illustration on how Independent Component Analysis (Equation 4, top illustration) and Singular Value Decomposition (Equation 5, bottom illustration) are used in analyzing context data. The extracted features $\vec{f_i}$ are the columns of S^T and V.

to be present in the tagged Brown corpus. Each word w_n was encoded as a column vector $\vec{v}_n$ of length N, where the n:th element was one for the word w_n and the other elements were zero.

The word categories collected from the tagged Brown corpus were encoded as column vectors $\vec{c}_k$. The vectors were constructed as the sum of unique words marked with the word category tag t_k

$$\vec{c}_k = \sum_{w_n \text{ with } t_k} \vec{v}_n \tag{1}$$

which makes the vectors have only zeros and ones in the elements. Each word w_n belongs to at least one word category. Word categories without any words in the analyzed vocabulary were removed which left us $K = 58$ word categories and the corresponding word category vectors $\vec{c}_k$.

A single row $\vec{x}_c$ in the context-word matrix X was calculated for a context c. In our experiments, the context c consisted of one context word and the position of the analyzed word related to the context word. For instance, one could consider the word preceding or following the context word. The vector $\vec{x}_c$ was created by examining all the instances of the context c in the corpus and taking the sum

$$\vec{x}_c = \sum_{w_n \text{ in context } c} \vec{v}_n^T \tag{2}$$

of the analyzed words found in the instances of the particular context c. For example, analysis of the three words in the corpus "$w_1\ w_2\ w_3\ w_2\ w_1\ w_2$" with the two contexts being the immediately following words for w_1 and w_2 creates the data matrix X

$$\mathrm{X} = \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \end{pmatrix} = \begin{pmatrix} \vec{v}_2^T + \vec{v}_2^T \\ \vec{v}_1^T + \vec{v}_3^T \end{pmatrix} \tag{3}$$

because w_2 follows w_1 twice and both w_1 and w_3 follow w_2 once.

For the feature extraction, a data matrix X of size 2000×10000 was created by calculating contextual information with the most common one thousand words in the Gutenberg corpus. Contexts were calculated separately for the words immediately following and preceding the analyzed word, resulting in 2000 different contexts calculated for the N words. Also, the data matrix was preprocessed by taking the logarithm of the elements increased by one to lessen the differences in the frequencies.

2.2. Feature extraction

Our goal is to extract a number of features $\vec{f_i}$ from the context-word matrix X.

The linear generative model

$$\mathrm{X} = \mathrm{AS} \tag{4}$$

of Independent Component Analysis [12, 13] in matrix form explains the rows of the data matrix X in terms of a mixing matrix A and independent components S by assuming the independence of the rows of S. We used the FastICA [14] Matlab package to extract a wanted number of features. Parameter selections were the same as reported in [1].

Singular Value Decomposition[3]

$$\mathrm{X} = \mathrm{UDV}^T \tag{5}$$

explains the data in terms of left singular vectors in U, singular values in D and right singular vectors in V. The singular vectors are orthogonal

$$\mathrm{U}^T\mathrm{U} = \mathrm{V}^T\mathrm{V} = \mathrm{I} \tag{6}$$

and D is a square diagonal matrix. In order to extract a wanted number of the largest singular values and vectors, the Matlab command `svds` was applied to the data X. Usually, the data X is a word-context matrix in LSA. Our choice of a context-word matrix does not affect the results, as it only changes the roles of the matrices U and V.

[3] As a textbook account on SVD one can use, e.g.,[15]. An important related application is presented in [16]. The history of the SVD method has been presented in [17].

The two feature extracting methods are illustrated in Figure 1. The matrices S^T and V have the feature vectors $\vec{f_i}$ as columns. We will use the notation $\vec{f_i}$ for the feature vectors regardless of the extraction method. The feature vectors $\vec{f_i}$ were scaled to unit variance as a post-processing step after the feature extraction. The feature vectors can be understood to be encoded similarly to the rows of the data matrix X, where the value of the n:th element is the activity of the feature for the n:th word.

A schematic illustration of the possible nature of the emergent features for two words is shown in Figure 2, where the height of the bar represents the value of the feature for the word. More than one feature may be active for each word, for instance, the word 'ate' has both the verb and the past tense features very active.

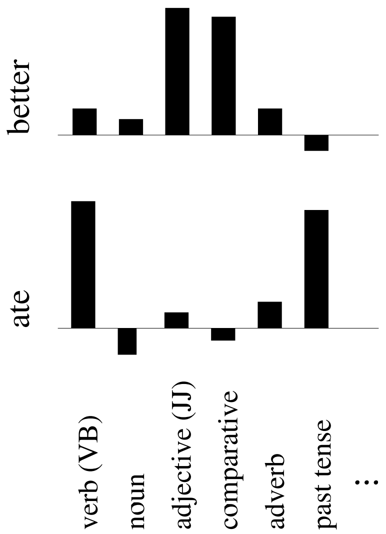


Figure 2. A schematic illustration of the possible nature of the emergent features. The heights of the bars represents the activities of each feature (bottom) for the words (left).

2.3. Comparison method

Considering two word categories k and l and the corresponding word category vectors $\vec{c}_k$ and $\vec{c}_l$, the words w_n in the vocabulary can be divided into four types:

1. words belonging to category k and not to l,
2. words belonging to category l and not to k,
3. words belonging to both categories k and l and
4. words belonging neither to category k nor to l.

Each word belongs to exactly one of these types.

A two-dimensional subspace of the feature vectors $\vec{f_i}$ and $\vec{f_j}$ can be visualized as a scatterplot, where the points $(x_n, y_n) = (f_i(n), f_j(n))$ are the N words in the vocabulary. The separation capability of the individual features can be studied by examining the placement of the four different types of words in the plane. What we would want to see is type 1) and 2) words in different axis away from the origin, type 3) words away from the origin, and type 4) words in the origin. The novel separation measure that we present in this paper rewards or penalizes words depending on word's position on the plane and the type of the word. The words in type 3) represent ambiguity in language and are considered separately of the words in the first two types.

Next we will introduce the separation measure we use in comparing the learned features. The measure works in three parts. First we calculate the amount of separation that two features create for two word categories. Next we choose the best feature pair separating two categories. Finally we consider the mean separation capability of a set of features for a set of categories.

The Cartesian coordinates (x_n, y_n) can be represented in polar coordinates (r_n, θ_n) and collected into a radial vector $\vec{r}$ and an angle vector $\vec{\theta}$. The radial distance r_n from origin, the angle θ_n from the first feature axis and the type of the word w_n are used in calculating a feedback value that matches with our idea of where the word should be if the features separate the word categories. The plane and the optimal positions of words with different types are illustrated in Figure 3. A positive value is given to a word located in a place that is beneficial to the separation of the word categories and acts as a reward, whereas a negative value is a penalty.

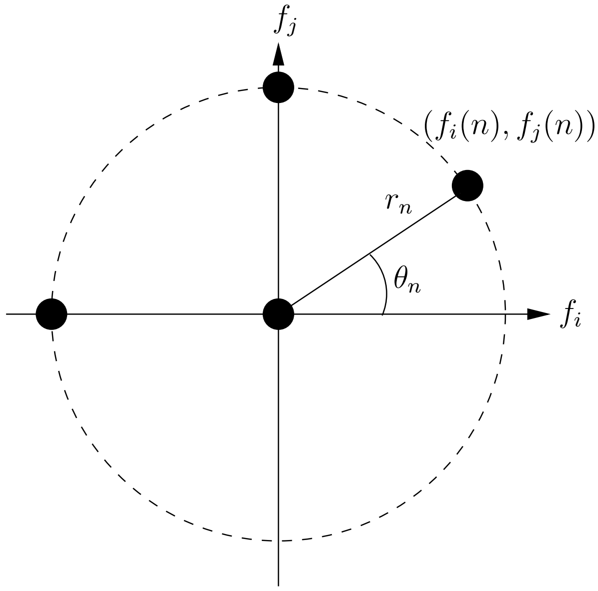


Figure 3. A single word w_n plotted to the two-dimensional plane defined by two feature vectors is rewarded or given a penalty based on the location and the type of the word. The most beneficial points to the separation and also the most rewarded and less penalized are the different coordinate axes at distance r_n for types 1) and 2), a circle of radius r_n for type 3) and the origin for type 4) words.

The separation capability of two features i and j and two word categories k and l is calculated as the mean of the rewards and penalties over all the N words w_n with the formula

$$
\begin{aligned}
&\operatorname{sep}(k,l,i,j) = \\
&\quad ((\vec{c}_k - \vec{c}_l)^T \mathrm{R}^P (|\cos\vec{\theta}| - |\sin\vec{\theta}|) + \\
&\quad (\vec{c}_k + \vec{c}_l - \vec{1})^T \mathrm{R}^P \vec{1}) \frac{1}{N}
\end{aligned} \tag{7}
$$

where the diagonal matrix $\mathrm{R} = \operatorname{diag}(\vec{r})$ has the radial distances from origin in the diagonal values, sin, cos and $|\cdot|$ are element-wise functions, and $\vec{1}$ is a column vector of length N with ones in all of the elements. The scalar value P can be used to scale the radial distances. A value $P = 2$ was used in our experiments. The larger the value of $\operatorname{sep}(k,l,i,j)$, the better the separation is according to the measure.

This method allows us to do a more comprehensive analysis of the learned features compared with the analysis in [10]. For instance, we are able to see if a word category is always best separated by the same feature when compared against other word categories. If the learned features are mixtures of the word categories, the best separating feature might differ when tested against different word categories.

The best feature pair (i_{kl}, j_{kl}) for the word category pair (k,l) is selected as the pair giving the highest value with the separation measure

$$
\begin{aligned}
\operatorname{sep}(k,l) &= \max_{i,j} \operatorname{sep}(k,l,i,j) && (8)\\
(i_{kl}, j_{kl}) &= \arg\max_{i,j} \operatorname{sep}(k,l,i,j) && (9)
\end{aligned}
$$

The separation capability of a feature set is measured as the mean of the best separation over all word category pairs

$$
\operatorname{sep} = \frac{2}{K^2 + K} \sum_{k \geq l} \operatorname{sep}(k,l) \tag{10}
$$

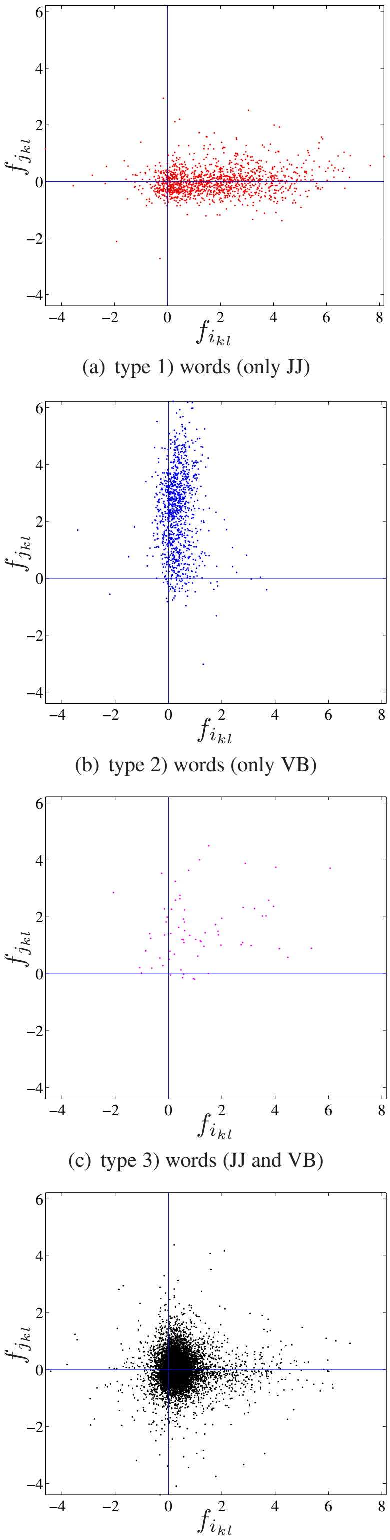
where K is the number of word categories.

3. COMPARISON RESULTS

The data and methods that we have used were introduced in Section 2. Next we will show the most interesting results that will show how the ICA and SVD methods compare against each other.

Figures 4 and 5 compare a single word category pair (JJ for adjectives and VB for verbs[4]) with the best found feature pair separating the word categories according to Equation 9. In both figures sixty features were extracted using SVD and ICA. The best found feature pair separating the two categories are more clearly aligned with the coordinate axis for the ICA-based features. With both methods, the shown features were consistently chosen to represent the JJ and VB features when the JJ and VB categories were tested against all the word categories and can be considered as the best possible features for the word categories.

[4] We use the tag naming adopted in the Brown corpus.



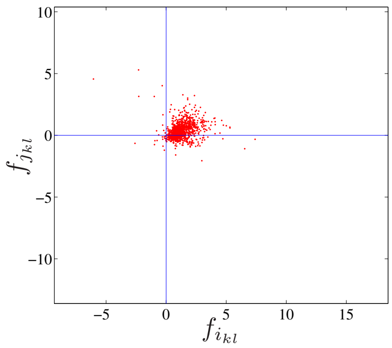
(a) type 1) words (only JJ)

(b) type 2) words (only VB)

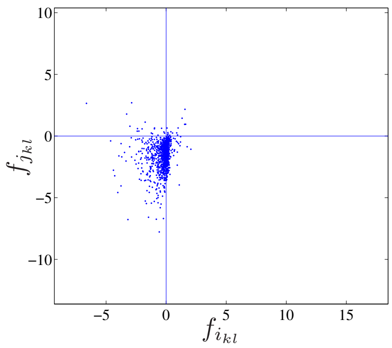
(c) type 3) words (JJ and VB)

(d) type 4) words (neither JJ nor VB)

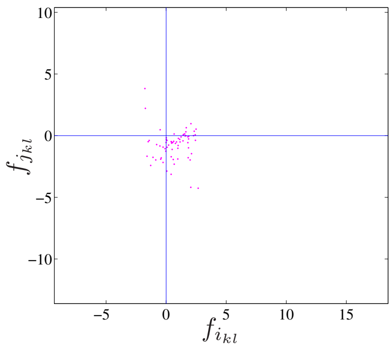
Figure 4. The found best ICA-feature pair separating the JJ (adjective) and the VB (verb) categories with Equation 9. (a)-(d) shows the words in the four types, listed in Section 2.3, plotted in the subspace created by the two features. The words in the two categories are clearly aligned along the two axes.



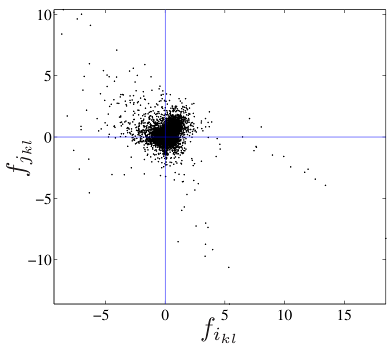
(a) type 1) words (only JJ)



(b) type 2) words (only VB)



(c) type 3) words (JJ and VB)



(d) type 4) words (neither JJ nor VB)

Figure 5. The found best SVD-feature pair separating the JJ (adjective) and the VB (verb) categories with Equation 9. (a)-(d) shows the words in the four types, listed in Section 2.3, plotted in the subspace created by the two features. The separation is not visually as clear as in Figure 4 showing the best separating ICA-based features.

Figure 6 shows the separation with Equation 10 for the ICA-based and the SVD-based features as a function of the number of extracted features. The ICA-based separation measure was calculated for five different feature sets extracted with different runs of the FastICA algorithm and the mean and one standard deviation of different runs is shown. The ICA-based features give clearly higher separation and keeps increasing with the number of extracted features.

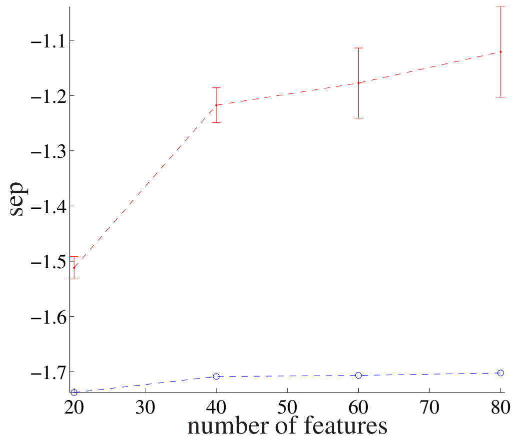


Figure 6. Comparison between ICA (upper curve) and SVD (lower curve) as a function of the number of extracted features. The y-axis shows the separation calculated with Equation 10. For the ICA-based features the mean and one standard deviation of five iterations is shown.

4. DISCUSSION

Statistical analysis of words, expressions and documents in their contexts have become commonplace. The philosophical, methodological and practical aspects related to this approach are wide. Using the ICA-based analysis of contexts we wish to show that the emergence of linguistic knowledge is possible without predetermined syntactic or semantic categories. Similar supporting evidence for the emergence has earlier been presented by a large number of researchers including but not limited to [3, 4, 5, 16]. The *Word ICA* method [1, 2, 9] has the characteristic that one can obtain a collection of emergent features that facilitates a distributed representation of words. In this representation each word may belong to several categories simultaneously. In this paper we have shown that these features not only characterize words in an intuitively appealing manner but that the features match with categories defined by linguists much more clearly than those obtained using Singular Value Decomposition. According to the results shown in this paper, the higher-order assumption of statistical independence in ICA gives more natural and intuitive results compared to the second-order decorrelation of SVD.

Both the Word ICA method and Latent Semantic Analysis produce vector representations for words. The emergent features still require further research on how they encode the structure of the language and on how the properties of individual features could be used to generate and further analyze language. Landauer and Dumais [18] have presented strong claims about the Latent Semantic Analysis as a means for the acquisition, induction and representation of knowledge. We are tempted to conclude that these processes can be modeled even more accurately when Independent Component Analysis is used. It is to be noted, though, that the word contexts do not need to be based only on text corpora but also multimodal contexts can be considered (see, e.g., [19, 20]).

5. ACKNOWLEDGEMENTS

We wish to thank Aapo Hyvärinen for his insightful guidance and the anonymous reviewers for their valuable comments on this paper.

6. REFERENCES

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

[2] T. Honkela and A. Hyvärinen, "Linguistic feature extraction using independent component analysis," in *Proceedings of IJCNN 2004, International Joint Conference on Neural Networks*, 2004, pp. 279–284.

[3] H. Ritter and T. Kohonen, "Self-organizing semantic maps," *Biological Cybernetics*, vol. 61, no. 4, pp. 241–254, 1989.

[4] K. W. Church and P. Hanks, "Word association norms, mutual information and lexicography.," *Computational Linguistics*, vol. 16, pp. 22–29, 1990.

[5] H. Schütze, "Dimensions of meaning," in *Proceedings of Supercomputing*, 1992, pp. 787–796.

[6] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, MA, 1999.

[7] T. Kolenda, L. K. Hansen, and J. Larsen, "Signal detection using ICA: application to chat room topic spotting," in *Proceedings of ICA2001, the Third International Conference on Independent Component Analysis and Signal Separation*, 2001, pp. 540–545.

[8] E. Bingham, J. Kuusisto, and K. Lagus, "ICA and SOM in text document analysis," in *Proceedings of the 25th ACM SIGIR Conference on Research and Development in Information Retrieval*, 2002, pp. 361–362.

[9] T. Honkela, A. Hyvärinen, and J. Väyrynen, "Emergence of linguistic features: Independent component analysis of contexts," in *Proceedings of NCPW9, Neural Computation and Psychology Workshop, Plymouth, England*, 2005, p. (in print).

[10] J. J. Väyrynen, T. Honkela, and A. Hyvärinen, "Independent component analysis of word contexts and comparison with traditional categories," in *Proceedings of NORSIG 2004, the 6th Nordic Signal Processing Symposium*, 2004, pp. 300–303.

[11] S. Dodel, J. M. Herrmann, and T. Geisel, "Comparison of temporal and spatial ICA in fMRI data analysis," in *Proceedings of ICA2000, the Second International Conference on Independent Component Analysis and Signal Separation*, Helsinki, Finland, 2000, pp. 543–547.

[12] P. Comon, "Independent component analysis—a new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.

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

[14] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, pp. 626–634, 1999.

[15] S. Haykin, *Neural Networks. A Comprehensive Foundation*, Prentice Hall, 1999.

[16] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, vol. 41, pp. 391–407, 1990.

[17] G. W. Stewart, "On the early history of the singular value decomposition," *SIAM Review*, vol. 35, pp. 551–566, 1993.

[18] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge," *Psychological Review*, vol. 104, pp. 211–240, 1997.

[19] L. K. Hansen, J. Larsen, and T. Kolenda, *Multimedia Image and Video Processing*, chapter On Independent Component Analysis for Multimedia Signals, pp. 175–199, CRC Press, 2000.

[20] V. Tuulos, J. Perkiö, and T. Honkela, *Adaptive and Statistical Approaches in Conceptual Modeling, Report A75*, chapter Modeling Multimodal Concepts, pp. 45–55, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland, 2005.