Comparisons of Self-Organized Word Category Maps

Timo Honkela

Helsinki University of Technology
Neural Networks Research Centre
P.O.Box 2200, FIN-02015 HUT, Finland
Timo.Honkela@hut.fi

Abstract

It has been shown earlier that the Self-Organizing Map (SOM) can be applied to the analysis and visualization of similarities of words in their usage in short contexts formed of adjacent words. These SOMs of words, called word category maps, have many potential applications. One of the application areas is information retrieval and data mining of textual document collections where the word category maps can be used in document encoding as has been demonstrated in the WEBSOM project. This paper concentrates on the question of how to create good word category maps, and specifically, how to compare different map instances. A general map comparison method that takes into account both the map topology and nonlinearity of the SOM is used. The paper presents results of comparisons in two experiments: first related to the number of words on a map, and second concerning the neighborhood type used in the SOM algorithm.

1 Introduction

Kohonen's Self-Organizing Map (SOM) [7, 8] algorithm can be used to create word category maps that describe relations of words based on their contexts. Interrelated words that have similar contexts appear close to each other on the map. In its simplest form the context consists of the neighboring words in a sentence ([2, 13, 14], see also [12]). The ith word in the sequence of words is represented by an n-dimensional real vector x_i (in the experiments reported here n was 50) with random-number components. Each unique word is encoded only once, i.e., when a word appears in the sequence again, the same random vector is used as its representation. The word to be mapped, "middle word", is encoded by εx_i where ε is a small scalar number (e.g., $\varepsilon = 0.2$). The scalar ε is used in order to enhance the contribution of the context in the organizing process. The context word vectors are calculated as the estimate of the expected value evaluated over the text corpus. In practice this means that the context of each distinct word is calculated as an average over all of its appearances in the text. This approach reduces the computational load significantly. The main disadvantage is that a word gets a single vector representation even if it happens to be severely ambiguous. Now the $X(i) \in \Re^{n\times(m+1)}$, the input vectors to the word category map, consist of concatenated "middle word" and context components (see, e.g., [2] for more details). The m refers to the width of the context. The SOM is calibrated after the training process by inputting the $X(i)$ once again to the word category map and labeling the best-matching nodes according to symbols corresponding to the x_i parts of the $X(i)$.

Previous experiments [2, 13, 14] have shown that the overall organization of a word category map reflects syntactical categories. On the other hand, the local organization seems to follow semantic features. Often a node becomes labeled by several symbols that are synonymic (see, e.g., [1, 5]). It is important to notice that the quality of a word category map cannot easily be evaluated by using traditional technical measures like the quantization error, or by comparing the resulting categories with traditional linguistic categories. In the following, some possibilities for comparing word category maps are outlined. Two specific parameters and their effects are considered in more detail: the number of words used as input and choice of the neighborhood type. The intent of this paper is to give insight on the task of comparing word category maps in order to help successful map creation.

2 Comparing Word Category Maps

There are many fundamental methodological alternatives and parameters that influence the quality of a word category map. One may, for instance, vary the width of the context, the method of how the context is encoded, the length of the word vector, the approach of how the words to be mapped are selected, and the parameters of the SOM algorithm including neighborhood width and type, learning coefficient, and length of the learning.

One may ask how significant the differences between the different maps are, and how one should compare the maps (see, e.g., the two maps in Fig. 1). The effects of parameter changes should also be compared with the natural variance that is present in the stochastic learning process. A traditional method for map evaluation is the use of quantization error measure. It is useful in measuring the goodness of a map in representing the input data.

One might compare two word category maps by computing the distance of each pair of words on the two maps. A map in which the words "algorithms" and "algorithm" are mapped into the same node is clearly different from the map in which those two words are, say, in the opposite corners. When the comparison is made for all of the pairs a general result is gained. Comparison of all the words pairs gives a full account of the similarity of the maps.

A straightforward comparison on the surface level of map outlined above may distort the result since the mapping found by the SOM is nonlinear. The distance between any two nodes could, however, be calculated in the original input space. In a visual inspection, one can use a gray level coding that illuminates the distances between neighboring nodes [15, 16]. A general map comparison method that takes into account both the map topology and nonlinearity of the SOM has been introduced by Kaski and Lagus [6]. In the comparison method, the dissimilarity of two maps is defined as the average (normalized) difference in how these two maps represent the distance between two data items that are here the word vectors. The distance between a pair of data samples is computed along the shortest path on the map in the original input space. The end points of the path are the best matching reference vectors, and the path passes through a series of reference vectors. In the series the nodes corresponding to each successive pair of reference vectors must be immediate neighbors (see [6] for details).

3 Some comparison results

In the following, examples of how much some specific word category map parameters affects the result are studied using the map comparison method of [6].

3.1 Number of words in the map

The decision of what is a suitable number of words in the word category map is influenced by multiple factors. This issue is partially related to the intended use of the map but also some general remarks can be made. The number of words and the map size determine how many words are mapped into a single map node on the average. If this average is high, it is most probable that the map nodes become "overloaded", i.e., they have to try to represent multiple features or categories. To consider applicational aspects one may take the WEBSOM method [3, 4, 5, 10, 11] as an example. The WEBSOM system creates maps of textual document collections using the word

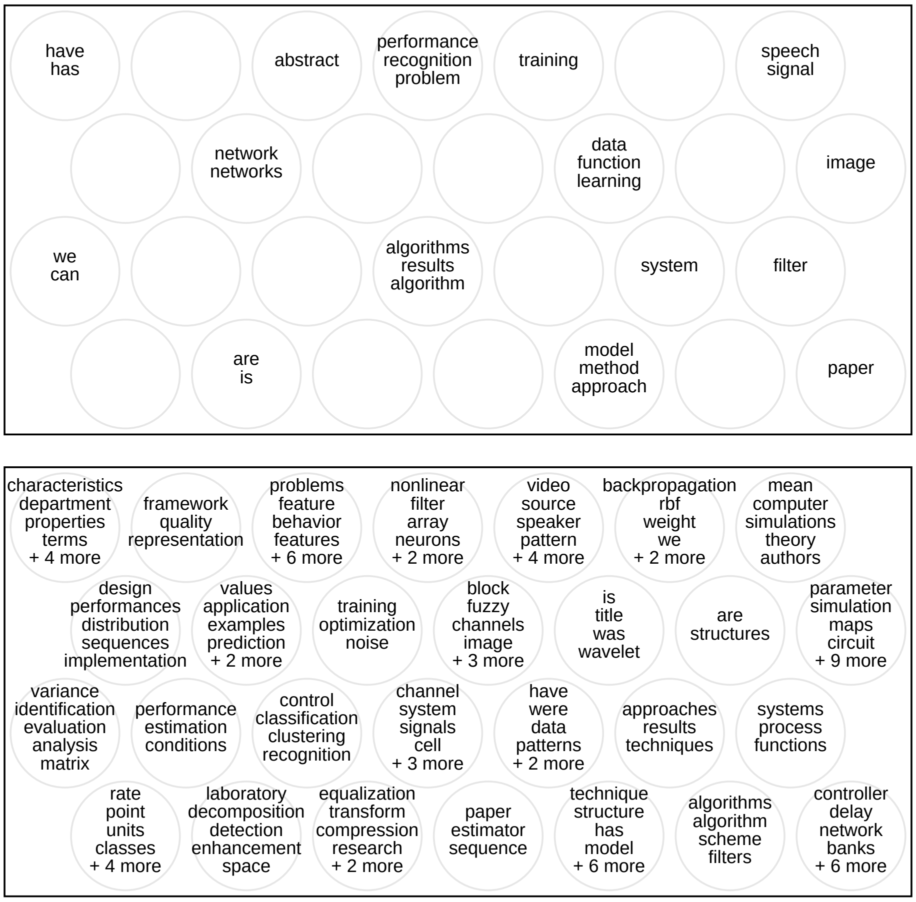

Figure 1: An example of two map instances. The upper word category map contains 28 manually selected words (mainly nouns and verbs), the lower one additional 135. The input material consisted of the abstracts of two conferences: ICNN'96 and EUSIPCO'96. The input had 42 421 separate trigrams, i.e., unique phrases of three words. The length of the overall text was 108 318 words. When the quality of the maps is considered one has to take into account the rather small corpus used that made possible the creation of hundreds of test maps.

category map in document encoding. Each word map node corresponds to a single element in the vector representation of a document. The words that "hit" the same word map node contribute similarly to the map histogram. Therefore the number of words in a node should be high enough, say 3 to 5, so that relevant synonyms are included. The limit depends on the "categorical or conceptual density" of the vocabulary being mapped. In the lower map of Fig.1 the words "is", "are" and "were" are located in the neighboring nodes but the map is clearly overloaded. One should note, however, that a rather low word per node ratio isn't as harmful in the WEBSOM system as one might suspect since the histogram is smoothed before it is used as a document "fingerprint". Too high a ratio would lead to overloading, which is destructive especially for short documents that contain little redundant information. The redundancy of long documents, the fact that they have multiple phrases revealing the topic, helps the system in creating a balanced representation.

The number of word instances in the text should naturally be large enough so that the sample of the typical contexts of the word would be representative. One issue of interest is how using less frequent words influences the organization of the more common words.

To investigate this issue, two collections of 20 word category maps were used to form a basis for the comparisons. In the first collection the maps simply contained 38 most frequent words in the corpus. The differences between the maps were caused by the stochastic learning process. For the second test, words were picked up manually so that only nouns and some of the most general verbs were included. This initial test set consisted of 20 word maps with 28 words.

The number of words was then gradually increased following the order of their decreasing frequency of occurrence. In the second test, only the nouns were included. Each new map (in the series of 20) was compared with the corresponding original map using the comparison method [6]. The set of word vectors of the original maps was used as the basis for the comparisons. The basic idea was to determine how much the inclusion of the less frequent words "disturbs" the organization of the more common words.

In Fig. 2 the (dis)similarity between a series of maps is presented. The first graph (left hand side) shows how a map degenerates in comparison with the original map (having least words) when less frequent words are added. In this first experiment where the words were used regardless of their category, the distortion effect approximately reaches its maximum as the number of words is doubled. In the second experiment (right hand side in the Fig 2), the distance to the reference maps with the smallest number of words is rather small in comparison with the first experiment.

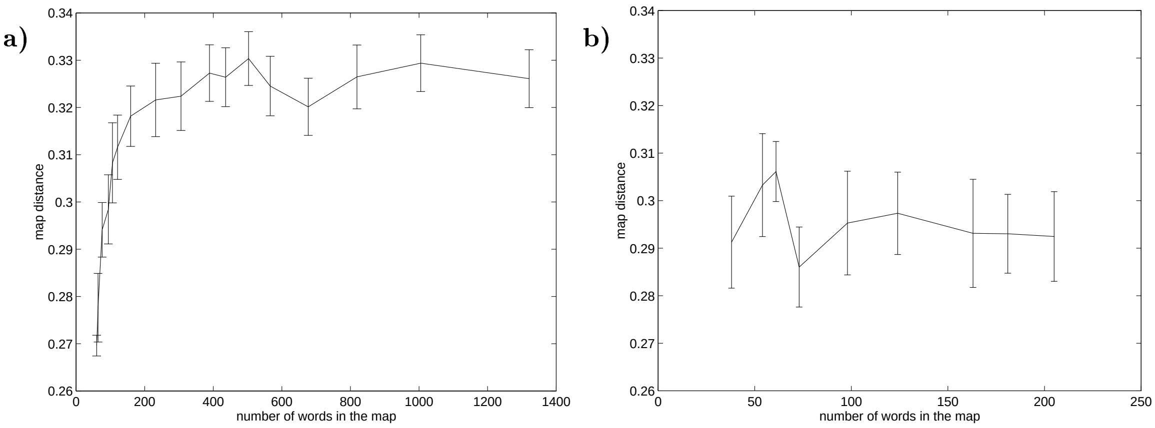

Figure 2: Relative dissimilarity of maps with different number of words mapped on them in the order of their decreasing frequency (a) when words of all categories are included, and (b) when only nouns are used. In (b), the text material was the same as in Fig. 1. The bars denote the standard error of the mean for each set of comparisons.

3.2 Neighborhood type

The SOM_PAK program package [9] has been used in the experiment reported in this paper. The package contains two options for the definition of the neighborhood function, $h_{ci}(t)$. The simpler of them, called "bubble", refers to a neighborhood of array points around node c. This index set can be denoted by N_c with $h_{ci} = \alpha(t)$ if $i \in N_c$ and $h_{ci} = 0$ if $i \notin N_c$, where $\alpha(t)$ is some monotonically decreasing function of time ($0 < \alpha(t) < 1$).

Another widely applied neighborhood kernel can be written in terms of the Gaussian function,

$$h_{ci} = \alpha(t) \cdot \exp\left(-\frac{||r_c - r_i||^2}{2\sigma^2(t)}\right) \quad , \tag{1}$$

where $\alpha(t)$ is a scalar-valued "learning rate", and the parameter $\sigma(t)$ defines the width of the kernel; the latter has the same purpose as the radius of N_c above. Both $\alpha(t)$ and $\sigma(t)$ are some monotonically decreasing functions of time. In the experiments, the learning was divided into two phases. In the first run, the $\alpha(0)$ was 0.2, the $\sigma(0)$ was 4, and the learning lasted for 50000 steps, and in the second run the figures were 0.02, 1, and 300000, correspondingly.

The effect of using Gaussian and bubble neighborhoods on word category map formation was studied by creating ten map instances of both types. Three kinds of comparisons were then made using the method of [6]. The distance between all the 45 different pairs of maps with (1) bubble neighborhood and (2) Gaussian neighborhood were measured. Also the mixed pairs were analyzed (3). The averaged results of the comparisons is shown in Fig. 3 where the bars denote the standard error of the mean for each set of comparisons. The results indicate that the Gaussian neighborhood is more stable: the distances between the map instances are smaller on the average.

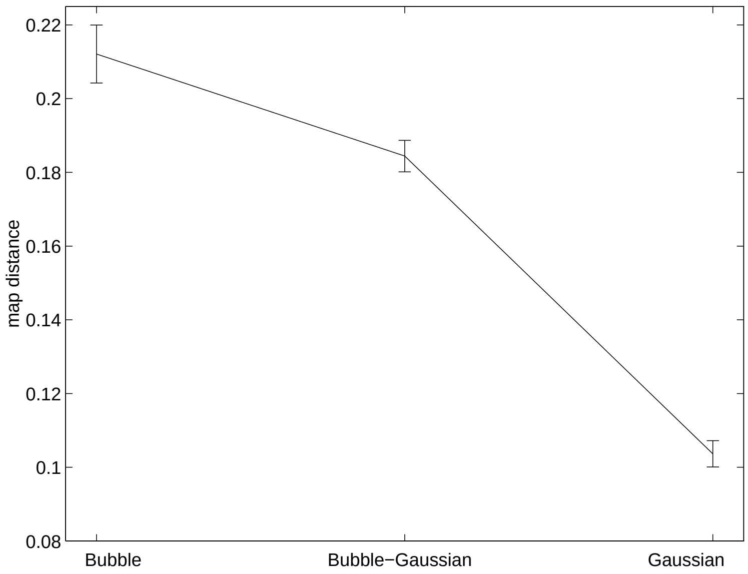

Figure 3: Comparison of map instances created using bubble and Gaussian neigborhoods. The bars denote the standard error of the mean for each set of comparisons.

4 Discussion

The paper has described some possibilities for quantitative analysis of word category maps. There are many fundamental methodological alternatives especially in input preparation and parameters that influence map formation. The comparison method used in the experiments proved to be useful in analyzing the advantages and disadvantages of the alternatives that were studied. The results indicated, for instance, that the inclusion of less frequent words may be fruitful area for further research. It would also be of interest to take the frequency of the words in the text into account, e.g. by weighting their contribution in the learning process using a logarithmic scale.

Acknowledgments

I wish to thank Sami Kaski, Teuvo Kohonen, and Krista Lagus for useful discussions and for their comments and advice.

References

[1] S. Finch and N. Chater. Unsupervised methods for finding linguistic categories. I. Aleksander and J. Taylor, eds., *Artificial Neural Networks, vol. 2*, pp. II-1365-1368, North-Holland, 1992.

[2] T. Honkela, V. Pulkki, and T. Kohonen. Contextual relations of words in Grimm tales analyzed by self-organizing map. In F. Fogelman-Soulié and P. Gallinari, eds., *ICANN-95, Proceedings of International Conference on Artificial Neural Networks, vol. 2*, pp. 3–7. EC2 et Cie, Paris, 1995.

[3] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen. Newsgroup exploration with WEBSOM method and browsing interface. Technical Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, January 1996. WEBSOM home page (1996) available at http://websom.hut.fi/websom/.

[4] T. Honkela, S. Kaski,K. Lagus, and T. Kohonen. Exploration of Full-Text Databases with Self-Organizing Maps. In *Proceedings of International Conference on Neural Networks (ICNN-96), vol. 1*, pp. 56-61, IEEE, Piscataway, NJ, 1996.

[5] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Creating an order in digital libraries with self-organizing maps. In *Proc of World Congress on Neural Networks (WCNN-96)*, 1996.

[6] S. Kaski, and K. Lagus. Comparing Self-Organizing Maps. *ICANN-96, Proc. of International Conference on Artificial Neural Networks*, 1996.

[7] Kohonen, T.: Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, pp. 59–69, 1982.

[8] T. Kohonen. *Self-Organizing Maps.* Springer-Verlag, 1995.

[9] Kohonen, T., Hynninen, J., Kangas, J., and Laaksonen, J. (1996). SOM_PAK: The self-organizing map program package. Technical Report A31, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland.

[10] T. Kohonen, S. Kaski, K. Lagus, and T. Honkela. Very large two-level SOM for the browsing of newsgroups. In *Proc. of International Conference on Artificial Neural Networks*, 1996.

[11] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. Self-organizing maps of document collections: A new approach to interactive exploration. In *Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD-96)*, AAAI Press, Menlo Park, California, 1996.

[12] R. Miikkulainen. *Subsymbolic Natural Language Processing: An Integrated Model of Scripts, Lexicon, and Memory.* MIT Press, Cambridge, MA, 1993.

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

[14] H. Ritter and T. Kohonen. Learning 'semantotopic maps' from context. In *Proc. IJCNN-90-WASH-DC, Int. Joint Conf. on Neural Networks*, vol. I, pp. 23–26. Lawrence Erlbaum, Hillsdale, NJ, 1990.

[15] Ultsch, A., Siemon, H. P.: Kohonen's self organizing feature maps for exploratory data analysis. In *Proc. INNC90, Int. Neural Network Conf.*, pp. 305–308, Kluwer, Dordrecht, 1990.

[16] Ultsch, A.: Self-organizing Neural Networks for Visualization and Classification. In O.Opitz, B. Lausen and R. Klar, eds. *Information and Classification*, pp. 307-313, Springer-Verlag, Berlin, 1993.