Using Growing Hierarchical Self-Organizing Maps for Document Classification

Michael Dittenbach, Dieter Merkl, Andreas Rauber

Institute of Software Technology, Vienna University of Technology
Favoritenstr. 9-11 / 188, A-1040 Vienna, Austria
{mbach, dieter, andi}@ifs.tuwien.ac.at

Abstract. The self-organizing map has shown to be a stable neural network model for high-dimensional data analysis. However, its applicability is limited by the fact that some knowledge about the data is required to define the size of the network. In this paper we present the *Growing Hierarchical SOM.* This dynamically growing architecture evolves into a hierarchical structure of self-organizing maps according to the characteristics of the input data. Furthermore, each map is expanded until it represents the correspondig subset of the data at a specific level of granularity. We demonstrate the benefits of this novel model using a real world example from the text classification domain.

1. Introduction

The self-organizing map (SOM) [3] is an artificial neural network model that is well suited for mapping high-dimensional data into a 2-dimensional representation space. The training process is based on weight vector adaption with respect to the input vectors. The SOM has shown to be a highly effective tool for data visualization in a broad spectrum of application domains [4]. Especially the utilization of the SOM for information retrieval purposes in large free-form document collections has gained wide interest in the last few years [5, 6, 8]. The general idea is to display the contents of a document library by representing similar documents in similar regions of the map. One of the disadvantages of the SOM in such an application arena is its fixed size in terms of the number of units and their particular arrangement, which has to be defined prior to the start of the training process. Without knowledge of the type and the organization of the documents it is difficult to get satisfying results without multiple training runs using different parameter settings, which obviously is extremely time consuming given the high-dimensional data representation.

Only recently a number of neural network models inspired by the training process of the SOM and having adaptive architectures were proposed [2]. The model being closest to the SOM is the so-called *Growing Grid* [1], where a

SOM-like neural network grows dynamically during training. The basic idea is to add rows or columns to the SOM in those areas where the input vectors are not yet represented sufficiently. More precisely, units are added to those regions of the map where large deviations between the input vectors and the weight vector of the unit representing these input data are observed. However, this method will produce very large maps which are difficult to survey and therefore are not that suitable for large document collections.

Another possibility is to use a hierarchical structure of independent SOMs [7], where for every unit of a map a SOM is added to the next layer. This means that on the first layer of the *Hierarchical Feature Map* (HFM) we obtain a rather rough representation of the input space but with descending the hierarchy the granularity increases. We believe that such an approach is especially well suited for the representation of the contents of a document collection. The reason is that document collections are inherently structured hierarchically with respect to different subject matters. In parentheses we should mention that this is essentially the way how conventional libraries are organized for centuries. However, like with the original SOM, the HFM uses a fixed architecture with a specified depth of the hierarchy and predefined size of the various SOMs on each layer. Again, we need profound knowledge of the data in order to define a suitable architecture.

In order to combine the benefits of the neural network models described above we introduce a *Growing Hierarchical SOM* (*GHSOM*). This model consists of a hierarchical architecture where each layer is composed of independent SOMs that adjust their size according to the requirements of the input data.

The remainder of the paper is organized as follows. In Section 2. we describe the architecture and the training process of the *GHSOM*. The results of experiments in document classification with the *GHSOM* are provided in Section 3. Finally, we present some conclusions in Section 4.

2. Growing Hierarchical SOM

The key idea of the *Growing Hierarchical SOM* (*GHSOM*) is to use a hierarchical structure of multiple layers where each layer consists of a number of independent self-organizing maps. One self-organizing map is used at the first layer of the hierarchy. For every unit in this map a self-organizing map might be added to the next layer of the hierarchy. This principle is repeated with the third and any further layers of the hierarchical feature map.

Since one of the shortcomings of SOM usage is its fixed network architecture we rather use an incrementally growing version of the SOM. This relieves us from the burden of predefining the network's size which is rather determined during the unsupervised training process. We start with a "virtual" layer 0, which only consists of one single unit. The weight vector of this unit is initialized as the average of all input data. The training process basically starts with a small map of, say, 2×2 units in layer 1, which are self-organized according to the standard SOM training algorithm.

Just to summarize the training algorithm, an input pattern is selected randomly and presented to the neural network. Each unit determines its activation according to the distance between its weight vector and the input vector. The unit showing the smallest distance, i.e. the *winner*, as well as a number of units in the vicinity of the *winner* are adapted. Adaptation is performed as a gradual reduction of the difference between the vector's components. Hence, after the adaptation the *winner* will be more similar to the input pattern.

This training process is repeated for a fixed number λ of training iterations. Ever after λ training iterations the unit with the largest deviation between its weight vector and the input vectors represented by this very unit is selected as the *error unit*. In between the *error unit* and its most dissimilar neighbor in terms of the input space either a new row or a new column of units is inserted. The weight vectors of these new units are initialized as the average of their neighbors. This training process is highly similar to the *Growing Grid* model. The difference so far is that we use a decreasing learning rate and a decreasing neighborhood range instead of fixed values as proposed in [1]. Especially the fixed neighborhood range is problematic when the network grows to be larger after a series of insertions.

An obvious criterion to guide the training process is the quantization error q_i. It is calculated as the sum of the distances between the weight vector of a unit i and the input vectors mapped onto this unit and may be used to evaluate the mapping quality of a SOM based on the mean quantization error (MQE) of all units in the map. The lower the value of the MQE, the better the map is trained. A map grows until its MQE is reduced to a certain fraction τ_1 of the q_i of the unit i in the preceding layer of the hierarchy. Thus, the map now represents the data mapped onto the higher layer unit i in more detail.

However, the most important difference to the *Growing Grid* is the following. *Growing Grid* is designed to build a single SOM to represent the input data. In case of a large number of input data the resulting map will be large, too. Just to illustrate the point, consider a geographical map of *Europe* containing all the information that we expect a map of *Belgium* should contain. This hypothetical map of *Europe* will be of a size making it tremendously difficult to find an orientation. A similar situation occurs if the contents of a large document library is represented by a single map. Thus, we are rather interested in building small maps where each unit represents a number of input data which are further expanded in separate maps further down the hierarchy.

As outlined above the initial architecture of the *GHSOM* consists of one self-organizing map. This architecture is expanded by another layer in case of dissimilar input data being mapped on a particular unit. These units are identified by a rather high quantization error q_i above a threshold τ_2. This threshold basically indicates the desired granularity level of data representation as a fraction of the initial quantization error at layer 0. In such a case, a new map will be added to the hierarchy and the input data mapped on the respective higher layer unit are self-organized in this new map, which again grows until its MQE is reduced to a fraction τ_1 of the respective higher layer unit's quantization error q_i. Note that this may not necessarily lead to a balanced hierarchy. The depth of the hierarchy will rather reflect the ununiformity which should be expected in real-world data collections.

Depending on the desired fraction τ_1 of MQE reduction we may end up with either a very deep hierarchy with small maps, a flat structure with large maps, or – in the most extreme case – only one large map, which is similar to the *Growing Grid*. The growth of the hierarchy is terminated when no further units require expansion, i.e. all units represent the respective data with a quantization error q_i below τ_2.

3. Experiments

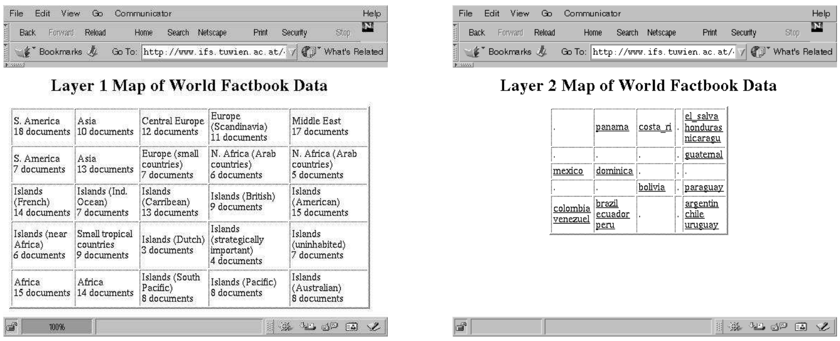
3.1. Input Data

For the experiments presented thereafter we use the 1990 edition of the *CIA World Factbook* (http://www.odci.gov/cia/publications/factbook) as a sample document archive. The *CIA World Factbook* represents a text collection containing information on countries and regions of the world. The information is split into different categories such as *Geography*, *People*, *Government*, *Economy*, *Communications*, and *Defense Forces*.

We use full-text indexing to represent the various documents. The complete information on each country is used for indexing. In total, the 1990 edition of the *CIA World Factbook* consists of 245 documents. During indexing we omit terms that appear in less than 15 documents or more than 196 documents, resulting in 959 content terms, i.e. terms used for document representation. These terms are weighted according to a simple $tf \times idf$ weighting scheme [9], i.e. term frequency times inverse document frequency. With this indexing vocabulary the documents are represented according to the vector-space model of information retrieval. The vectors representing the documents are further used for neural network training.

3.2. A *GHSOM* Atlas of Country Descriptions

Given the description of the regions political, economic and environmental features, the resulting *GHSOM* represents a rather intuitively interpretable geographically ordered mapping. The top-level map (cf. Figure 1(a)) evolved to a 5×5 map which already shows detailed clusters of the countries. For example, we can identify a cluster of predominantly Latin American countries in the upper left corner of the map, next to a cluster of Asian and European countries, with Middle East and Arab countries being located in the upper right corner of the map. We further find a cluster of African countries in the lower left corner of the map. The remainder of the SOM represents mostly islands, which in turn are clustered by political and economic ties as well as language or former colony status. For example, in the bottom right corner we have Australia and islands which are Australian territories.



(a) Top Layer

(b) Second Layer

Figure 1: Flat Hierarchy

If we want to have a more detailed view of a particular region, we have to take a look at the according map in the next layer. For example, the unit in the upper left corner, representing 18 Latin American countries, has been expanded in a second layer map, which in turn has grown to form another 5×5 SOM. The resulting sub-map is depicted in Figure 1(b), which now provides a more detailed representation of the respective countries. For example, we find Argentina, Chile and Uruguay, all of which are located in the southern half of South America and show strong economic ties, to be located on one unit in the lower right corner of the map, clearly separated for example from Middle American countries like El Salvador, Honduras and Nicaragua in the upper right corner of the second-layer SOM. As another cluster on this SOM we might mention the tropical countries Brazil, Ecuador, Peru, Columbia and Venezuela in the lower left corner of the map. Once again we should stress, that the *GHSOM* is not intended to cluster countries by their geographic location. However, surprisingly, the classification of the countries environmental, economic, and political descriptions results in a more or less geographically correct mapping. Still, this result is quite intuitive considering the fact that countries in neighboring geographic locations quite commonly are also very similar in terms of their climate, economical and political relationships etc.

Similarly, most other units of the first layer SOM have been expanded in second layer maps of between 4×3 and 6×4 units. Due to space considerations we cannot present these submaps in detail. However, the maps and country descriptions are available for convenient interactive exploration via our web server at http://www.ifs.tuwien.ac.at/ifs/research/ir/GHSOM.

Since the growth process of the *GHSOM* is guided by the fraction of the remaining mean quantization error at each layer, we can also choose to have the *GHSOM* evolve into a deep hierarchy of small maps, which proves to be specifically useful for large data sets. Requiring a less refined representation at each subsequent layer in favor of a more rigid splitting of clusters into separate maps results in a structure of 6 layers of maps of size 2×2 to 2×3 units. However, the document clusters remain more or less the same as for the flat *GHSOM*. For example, a layer 5 map in the Latin American countries branch represents exactly the country descriptions of the countries found in the lower half of Figure 1(b). Again, the resulting deep hierarchy *GHSOM* is available at the same URL for convenient comparison.

4. Conclusion

We presented the *GHSOM*, a novel neural network model based on the self-organizing map. The main feature of this model is its capability of dynamically adapting its architecture to the requirements of the input space. Instead of having to specify the precise number and arrangement of units in advance, the network determines the number of units required for representing the data at a certain accuracy level at training time. This growth process is guided solely by the desired granularity of data representation. As opposed to other growing network architectures, the *GHSOM* does not grow into a single large map, but rather dynamically evolves into a hierarchical structure of growing maps in order to represent the data at each level in the hierarchy at a certain granularity. This enables the creation of smaller maps, resulting in better cluster separation due to the existence of separated maps. It further allows easier navigation and interpretation by providing a better overview of huge data sets. The benefits of the proposed approach have been demonstrated by a real world application from the text classification domain.

References

[1] B. Fritzke. Growing grid – a self-organizing network with constant neighborhood range and adaption strength. *Neural Processing Letters*, 2(5), 1995.

[2] B. Fritzke. Growing self-organizing networks — Why? In *Proc Europ Symp on Artificial Neural Networks (ESANN'96)*, Bruges, Belgium, 1996.

[3] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 1982.

[4] T. Kohonen. *Self-organizing maps.* Springer-Verlag, Berlin, 1995.

[5] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. Self-organizing maps of document collections: A new approach to interactive exploration. In *Proc Int'l Conf on Knowledge Discovery and Data Mining (KDD-96)*, Portland, OR, 1996.

[6] D. Merkl. Exploration of text collections with hierarchical feature maps. In *Proc Int'l ACM SIGIR Conf on R&D in Information Retrieval (SIGIR'97)*, Philadelphia, PA, 1997.

[7] R. Miikkulainen. Script recognition with hierarchical feature maps. *Connection Science*, 2, 1990.

[8] A. Rauber and D. Merkl. Finding structure in text archives. In *Proc. European Symp. on Artificial Neural Networks (ESANN98)*, Bruges, Belgium, 1998.

[9] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley, Reading, MA, 1989.

.
M. Dittenbach, D. Merkl and A. Rauber: **Using Growing Hierarchical Self-Organizing Maps for Document Classification**
In: Proceedings of the 8. European Symposium on Artificial Neural Networks (ESANN'2000) April 26-28, 2000, Bruges, Belgium.