Neural Networks 105 (2018) 166–184

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

High-resolution Self-Organizing Maps for advanced visualization and dimension reduction

Ayu Saraswati, Van Tuc Nguyen, Markus Hagenbuchner [*], Ah Chung Tsoi

School of Computing and Information Technology, University of Wollongong, Australia

ARTICLE INFO

Article history:
Received 11 October 2017
Received in revised form 24 March 2018
Accepted 15 April 2018
Available online 4 May 2018

Keywords:
Self-Organizing Map
Artificial neural network
Clustering and classification

ABSTRACT

Kohonen's Self Organizing feature Map (SOM) provides an effective way to project high dimensional input features onto a low dimensional display space while preserving the topological relationships among the input features. Recent advances in algorithms that take advantages of modern computing hardware introduced the concept of high resolution SOMs (HRSOMs). This paper investigates the capabilities and applicability of the HRSOM as a visualization tool for cluster analysis and its suitabilities to serve as a pre-processor in ensemble learning models. The evaluation is conducted on a number of established benchmarks and real-world learning problems, namely, the policeman benchmark, two web spam detection problems, a network intrusion detection problem, and a malware detection problem. It is found that the visualization resulted from an HRSOM provides new insights concerning these learning problems. It is furthermore shown empirically that broad benefits from the use of HRSOMs in both clustering and classification problems can be expected.

1. Introduction

The SOM (Kohonen, 1982) is widely used for data visualization purposes and for the exploration stage of data mining applications. A key characteristic of SOM is its ability in providing a topology-preserving mapping of a high dimensional input (feature) space onto a low dimension grid (Kohonen, 1982). SOM is especially suitable for data visualization and analysis because it conveniently facilitates the user to use the unique insights of humans in being able to visualize from a two dimensional display the relationships among the input vectors in high dimensional space which is otherwise impossible. Humans can mentally connect the dots, or blobs in the two dimensional display, and then interpret them as groupings even though there are no visual boundaries drawn among the dots or blobs in the display. Therefore the SOM projection abilities help users to understand intricate relationships among the input vectors by exploring its mapping results on the display space. Such visualization would often act as a prelude to further processing of the input data (Kohonen, 1982).

The mapping quality depends on the granularity of the grid since the mapping space of the SOM is discrete. An HRSOM is a SOM consisting of a very large number of neurons and hence, the display space of HRSOMs is finely grained. The benefit of creating HRSOMs intuitively is to enhance the quality of the projected vectors or to improve the visualization of the macro as well as the micro structures, indicating relationships existing among the input vectors (Forti & Foresti, 2006; Skupin & Esperb, 2008); were these displayed on a low granularity grid, they would have become indistinguishable from one another. HRSOM increases the mapping space supporting the separation of dissimilar input patterns and hence can be more suitable for learning problems that exhibit complex relations among input vectors. In contrast, low resolution SOMs (LRSOMs) are unable to be used for the visualization of intricate and complex relationships among input vectors since they are forced into simpler structures. SOMs are also often used in cluster analysis. The use of low resolution SOMs can assist the formation of clusters by forcing the compression of information in the display space. However, for visualization and as a pre-processor in an ensemble model (Noi, Hagenbuchner, Scarselli, & Tsoi, 2013), it is often desirable to have a finely grained display space in order to reduce information loss.

The HRSOM's capability to maintain and show intricate data relationships is evaluated on the policemen dataset (Hagenbuchner, Gori, Bunke, Tsoi, & Irniger, 2003) and the KDD99 dataset (Hettich & Bay, 1999). This paper also presents an approach that exploits the discovered data relationships by using the HRSOM as a pre-processor in an ensemble learning architecture (Noi et al., 2013). This provides a pre-disposition of the classifier towards those classes which were implicitly formed by the HRSOM clusters, as no such cluster information was explicitly provided to the classifier, only the locations of the projected vectors onto the display space

[*] Corresponding author.
E-mail addresses: sa783@uow.edu.au (A. Saraswati), vtn966@uow.edu.au (V.T. Nguyen), markus@uow.edu.au (M. Hagenbuchner), act@uow.edu.au (A.C. Tsoi).

https://doi.org/10.1016/j.neunet.2018.04.011
0893-6080/© 2018 Elsevier Ltd. All rights reserved.

were provided to the classifier as augmented inputs. This allows a faster convergence in the training of the classifier, and resulting in a higher generalization accuracy when evaluated on some practical datasets, e.g., the intrusion detection dataset KDD99 dataset, and a malware detection dataset.

The SOM and MLP as a classifier are relatively "old" machine learning algorithms (Haykin, 2009). Newer methods such as those found in Deep Learning have shown superior performance on a range of learning problems. Nevertheless, the SOM remains indispensable in the data exploration stage of data mining and MLPs remain best suited for many learning problems for the following reasons:

• Deep Learning (Bengio, 2009) tends to require more training samples which would "cover" the input space well. The methods in this paper are thus more suited for learning problems with a relatively sparse coverage of the feature space (i.e. limited number of samples).

• The SOMs and MLP (Haykin, 2009) used in this paper are more scalable because the SOM has only one codebook layer which can be processed in parallel and the number of hidden layers in the MLP is much smaller than the number of hidden layers typically found in Deep Learning (Bengio, 2009). While it is possible to process neurons in a hidden layer in parallel it is necessary to process the various hidden layer one at a time (in sequence). The MLP and SOM are hence better suited for applications that require an implementation on devices with limited computational capability.

• The SOM and MLP algorithms (Haykin, 2009) are simpler than those typically found in Deep Learning (Bengio, 2009). This can be of relevance i.e. in data mining because users often prefer to understand the methods used. It is very hard to describe Deep Learning in a comprehensible fashion to users that have a limited technical background.

• The design of a SOM and MLP architecture is simpler than designing a Deep Learning architecture. It is much simpler to find, given a new learning problem, an optimal network architecture with SOMs, MLPs, because the number of unknown training parameters is smaller. Deep Learning architectures consist of numerous hidden layers, possibly with different widths in different hidden layers, and therefore many more parameters to adjust. The optimal number of layers and the optimal number of neurons for each hidden layer is hard to determine due to the many hidden layers. It is for this reason that Deep Learning architectures (Bengio, 2009) sometimes either have the same number of neurons in each layer or a trapezoidal architecture.

The main contributions of this paper can be summarized as follows:

Contribution 1: Recent work in Van Nguyen, Hagenbuchner, and Tsoi (2016) and Saraswati, Hagenbuchner, and Zhou (2016) provided the motivation, and described an approach which allows HRSOMs to be trained in a time efficient manner and reported that HRSOMs can show intricate details contained in the training dataset of learning problems. This paper is to give a better understanding of HRSOMs by extending the work presented in Van Nguyen et al. (2016) and Saraswati et al. (2016) through a qualitative and quantitative analysis of the results, their comparisons with those obtained using LRSOMs, and applications to a range of learning problems.

Contribution 2: The literature on SOM, see e.g., Kohonen (1982) suggests that SOMs should be small in size to limit the number of empty cells and to encourage the formation of clusters in the projection (see more specifically Vesanto & Alhoniemi, 2000, Wendel & Buttenfield, 2010). The recommendations go as far as to suggest $5\sqrt{N}$ as the maximum size of a SOM for learning problems with N samples (i.e. as is suggested in an influential paper by Vesanto in Vesanto & Alhoniemi, 2000). It was conjectured (Vesanto & Alhoniemi, 2000) that the relationships among training samples are lost when training very large SOMs due to the effects of entropy. These guidelines are widely used by the scientific community. Contrary to this belief this paper shows that HRSOMs not only retain relationships among the input data but also are able to show intrinsic details of these relationships. This paper together with (Saraswati et al., 2016; Van Nguyen et al., 2016) are among the first to show that, contrary to common beliefs, using small SOMs can have a negative effect on the quality of results.

Contribution 3: This paper presents an approach by which the intrinsic details shown in a HRSOM can be exploited. Towards this end, this paper presents an ensemble model (Noi et al., 2013) which uses the HRSOM as a preprocessor for augmenting samples before processing them by a supervised classifier, e.g., MLP (Haykin, 2009). It is shown that the results of the classifier improves by using this data augmentation technique and that the improvement grows with the size of the SOM thus demonstrating that mappings of HRSOMs are more informative. While the data augmentation idea was first introduced in Noi et al. (2013) with a LRSOM, this paper is the first to apply such an idea with a HRSOM, and showed that the HRSOM is more effective in pre-disposing the classification results towards good implicit clusters formed by the SOM than the less well-formed implicit clusters formed by a LRSOM.

Contribution 4: The paper presents an analysis of results on a range of learning problems. The results produced by the ensemble system (Noi et al., 2013) together with an HRSOM as a preprocessor are competitive with highly specialized (handcrafted) state-of-the-art approaches. The paper thus shows that the HRSOM used as a data augmentation scheme in the ensemble system (Noi et al., 2013) can be used as a general framework for a wide range of applications and that the results are expected to be competitive with approaches that specifically target a given learning problem.

The rest of the paper is organized as follows: Section 2 describes the model architecture of both the HRSOM as well as ensemble systems. Section 3 describes the learning problems that will be utilized. Section 4 presents the visualization capability of the HRSOM. Section 5 compares clustering performances of different resolutions of SOM. Section 6 discusses the effects of HRSOM in ensemble models. Section 7 provides concluding remarks and gives some future research directions. The analysis of results for a large malware detection learning problem contained in the Appendix confirms the general findings made in the main body of the paper.

2. Model architectures

This section describes the HRSOM (Kohonen, 1982) and the corresponding ensemble models (Noi et al., 2013) that will be studied in this paper.[1]

2.1. High resolution self-organizing map

The HRSOM training algorithm maintains the general properties of Kohonen's original SOM algorithm (Kohonen, 1982) as described in the following: The SOM algorithm performs a nonlinear and topology preserving projection of the n-dimensional input feature vectors onto a q-dimensional grid of *neurons*, and $n \gg q$ (Kohonen, 1982). The location of a neuron i in the q-dimensional grid is defined by its q-dimensional $\mathbf{l}_i$ location vector. Other neurons adjacent to a given neuron are considered belonging to its neighborhood denoted as $\mathcal{N}$ (Kohonen, 1982); the geometry of the neighborhood $\mathcal{N}$ could be square or hexagonal (Kohonen, 1982). Each neuron is associated with an n-dimensional codebook vector $\mathbf{m}$, where n is the same value as the dimension of the input vectors. The aim of the SOM training algorithm is

[1] The software used in this paper to train the LRSOMs, HRSOMs, and MLPs can be obtained from http://teaching.cs.uow.edu.au/~markus/data/Neurocomp2018.tgz.