Pattern Recognition using the TTOCONROT

César A. Astudillo[1]* and B. John Oommen[2]**

[1] Department of Computer Science, Universidad de Talca
Km 1. Camino a los Niches, Curicó, Chile
castudillo@utalca.cl

[2] School of Computer Science, Carleton University,
Ottawa, Canada: K1S 5B6.
oommen@scs.carleton.ca

Abstract. We present a method that employs a tree-based Neural Network (NN) for performing classification. The novel mechanism, apart from incorporating the information provided by unlabeled and labeled instances, re-arranges the nodes of the tree as per the laws of Adaptive Data Structures (ADSs). Particularly, we investigate the Pattern Recognition (PR) capabilities of the Tree-Based Topology-Oriented SOM (TTOSOM) when Conditional Rotations (CONROT) [8] are incorporated into the learning scheme. The learning methodology inherits all the properties of the TTOSOM-based classifier designed in [4]. However, we now augment it with the property that frequently accessed nodes are moved closer to the root of the tree.

Our experimental results show that on average, the classification capabilities of our proposed strategy are reasonably comparable to those obtained by some of the state-of-the-art classification schemes that only use labeled instances during the training phase. The experiments also show that improved levels of accuracy can be obtained by imposing trees with a larger number of nodes.

Keywords: Tree-Based SOMs, TTOSOM, CONROT, Pattern Recognition.

1 Introduction

In a previous work [1, 6], we presented a clustering algorithm that combined the philosophies defined by the tree-structured families of Self Organizing Maps (SOMs) and the field of Adaptive Data Structures (ADSs). The pioneering manner in which we perceived clustering, attempted to generate asymptotically optimal trees based on the access probabilities of the neurons[3]. In this paper we

* Assistant Professor. IEEE Member. The work of this author is partially supported by the FONDECYT grant 11121350, Chile.

** *Chancellor's Professor* ; *Fellow: IEEE* and *Fellow: IAPR*. This author is also an *Adjunct Professor* with the University of Agder in Grimstad, Norway. The work of this author was partially supported by NSERC, the Natural Sciences and Engineering Research Council of Canada.

[3] A paper that reported the preliminary results of the TTOCONROT won the best paper award in a well-known AI conference [2].

design a classifier based on these principles, and show the effect of the classification accuracies obtained on different real-world domains.

To report the contribution of this paper in the context of the work done in [4], we mention that in [4], we designed a classifier based solely on the Tree-Based Topology-Oriented SOM (TTOSOM) algorithm, and showed that it was able to learn the decision boundaries based on labeled and unlabeled samples simultaneously. This so-called "semi-supervised" learning classifier utilized the strategy presented by Zhu [12], which identified clusters by using a possibly large number of unlabeled samples, and subsequently, associating each neuron with a label by utilizing a small number of labeled instances. We showed that this approach, indeed, can produce accuracies that are reasonably comparable to the ones achieved by state-of-the-art classifiers.

A natural extension to our investigation is to develop a study analogous to the one performed in [4], but now considering the effect of the rotations in the tree. In this sense, our proposed methodology consists of deriving a classifier similar to the TTOSOM-based classifier, but using the TTOCONROT [6] as a foundation.

The remainder of the paper is organized as follows: The next section describes the necessary background regarding the SOM. Section 3 explains how the TTOCONROT is used to perform classification. Subsequently, Section 3 focuses on the experimental results, and finally, Section 4 contains the conclusions.

1.1 Literature Review

The Self Organizing Maps (SOMs) is a family of Neural Networks (NN) suitable for visualization and data clustering. The model uses a network of neurons arranged in a grid which are trained using a concept called *competitive learning*. In each training step, a new instance (also called input vector) is presented to the network and the most similar neuron is declared as the winner or *best matching unit* (BMU). To achieve this, each neuron is associated to a weight vector that possess the same dimensionality as the input vectors and a dissimilarity function (such as the Euclidean distance) is used to compare them. The SOM introduces the concept of neighborhood function, that identify a subset of neurons in the vicinity of the BMU. A central process of the training mechanism of the SOM is the migration phase, in which the BMU and its neighboring neurons are moved closer to the input vector. This movement is controlled by the so-called update rule:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \phi_{ci}(t)(\mathbf{x}(t) - \mathbf{w}_i(t)) \tag{1}$$

where $\mathbf{x(t)}$ is a d-dimensional vector that represents the input vector at time t, c is the index of the BMU, $\mathbf{w_i}$ is the weight vector associated to the i-th neuron and ϕ_{ci} is the neighborhood function.

As a result of the training process the weight vectors of the neurons absorb the properties of the original data distribution and its topological structure.

There are hundreds of papers reporting the applicability of the SOM in almost all branches of engineering (if not all) [7]. However, in spite of all these benefits, the SOM has known handicaps, some of which are discussed in [5, 7]. As a result, numerous variants of SOM has been designed in an effort to render the topology more flexible or to accelerate the learning process.

One of this variants is the Tree-based Topology Oriented SOM (TTOSOM) [3], which uses a neural tree instead of a grid. The SOM trains a user-defined tree in a similar manner compared to the SOM but using a neighborhood function defined over the tree which produces a completely different mapping. The TTOSOM has shown holographic properties and also reduces to the 1D SOM, when the tree is a "linear" sequence of neurons.

2 The TTOCONROT-Based Classifier

In [4], the authors designed a classifier based on the TTOSOM algorithm that was able to learn from labeled and unlabeled samples. This so-called "semi-supervised" learning classifier utilized the strategy presented by Zhu [12], and consisted of clustering the instances using a "massive" number of unlabeled samples, and subsequently, identifying the label of each neuron based on a possibly small number of labeled instances. The authors of [4] showed that this approach can, indeed, produce accuracies that are comparable to the ones achieved by state-of-the-art classifiers.

Our goal in this paper is to devise a classifier analogous to the one presented in [4], but this time based on the foundation of the TTOCONROT instead of the TTOSOM. In order to clarify the way in which this classifier is built, we will first summarize the main properties of the above-mentioned clustering technique.

The reader will recall that in [6], we had merged the fields of SOMs and ADSs. The adaptive nature of the strategy presented, namely the Tree-Based Topology-Oriented Topology Using Conditional Rotations (TTOCONROT), is unique because adaptation is perceived in two forms: The migration of the codebook vectors in the feature space is a consequence of the SOM update rule, and the rearrangement of the neurons *within* the tree is a result of the ADS-related rotations. This reorganization can be perceived to be both automatic and adaptive, such that on convergence, the DS tends towards an optimal configuration with a minimum average access time. In most cases, the most probable element will be positioned at the root (head) of the tree (DS), while the rest of the tree is recursively positioned in the same manner.

The TTOCONROT [6] is a further enhancement of the TTOSOM [3] which considers how the underlying tree itself can be rendered dynamic and adaptively transformed. To do this, we presented a method by which a SOM with an underlying BST structure can be adaptively re-structured using Conditional Rotations [8]. These rotations on the nodes of the tree are local, can be done in constant time, and performed so as to decrease the WPL of the entire tree. In [6], we also introduced the concept referred to as *Neural Promotion*, where neurons gain prominence in the NN as their significances increase. The advantages of such a scheme is that the leaned tree learns the topological peculiarities of the stochastic data distribution, and at the same time recursively positions neurons accessed more often close to the root. As a result, the TTOCONROT, converges in such a manner that the neurons are ultimately placed in the input space so as to represent its stochastic distribution, and additionally, the neighborhood properties of the neurons suit the best BST that represents the data.

Even though, the advantages of the CONROT algorithm are explained in [6], the proposed architecture allows the inclusion of alternative restructuring modules other than the CONROT. Potential candidates which can be used to perform the adaptation are the Splay and the Mehlhorn's D-Tree algorithms, among others [8].

Analogously to the classifier devised in [4], our aim is to design a classifier that works in two stages. First of all, the data distribution and its structure is learned in an unsupervised manner using the TTOCONROT scheme. In a second phase, we utilize some labeled samples to categorize the decision regions that have been previously created.

The TTOCONROT-based classifier uses the cluster-then-label paradigm [11], leading to an algorithm similar to the one described in [4], with the difference that the TTOCONROT is used as the unsupervised learning algorithm.

3 Experimental Results

3.1 Experimental Setup

In order to verify the capabilities of the TTOCONROT for classifying items belonging to the real-world domain, and for making the results comparable to the ones obtained by the TTOSOM-based classifier, we have chosen the same datasets described in [4]. These six datasets are Iris, Wisconsin Diagnostic Breast Cancer (WDBC), Wine, Yeast, Wine Quality and Glass datasets, all publicly available from the UCI Machine Learning repository [9]. For an explanation of each of these datasets, we refer the reader to Section 4.3 included in [4].

Analogous to the experiments performed in [4], the classifiers considered in this comparison include five supervised classifiers, namely, Bayes Networks (BN), Naive Bayes (NB), C4.5, k-Nearest Neighbors (KNN) and Learning Vector Quantization 1 (LVQ1), and three "semi-supervised" classifiers, namely the TTOCONROT, the TTOSOM and the SOM. The sampling method utilized to measure the accuracy was the stratified 10-fold cross validation.

3.2 Comparison to Other Classifiers

We started our experimental analysis by comparing the accuracies of the TTOCONROT with the rest of the classifiers mentioned above, using the parameter settings specified in [4]. The results obtained are presented in Table 1, which shows the accuracy of the classifiers obtained for the various datasets. For example, Table 1 shows that the TTOCONROT classifier, using *only* 15 neurons, accurately predicts, with an accuracy of 96.07%, the correct label of the instances belonging the *wine* dataset, which is outperformed only by the BN and the NB schemes. On the other hand, the SOM classifies correctly the same dataset with an *accuracy* of only 67.98%!

Dataset	**ROT15**	**TTO15**	**BN**	**NB**	**C4.5**	**KNN**	**LVQ1**	**SOM**
iris	94.00	92.00	92.67	96.00	96.00	95.33	96.00	84.67
wdbc	93.32	92.09	95.08	93.15	93.15	96.66	92.09	90.51
glass	53.74	52.34	71.96	49.07	67.76	67.76	61.22	63.08
wine	96.07	95.51	98.88	97.19	93.82	94.94	74.16	67.98
yeast	51.08	51.82	56.74	57.61	55.86	54.78	24.33	46.16
winequality	53.60	53.41	57.72	55.03	62.91	57.79	44.15	49.59

Table 1: General classification results of the TTOCONROT and other methods investigated, reported in terms of the accuracy.

We have also compared the TTOCONROT-based classifier with other VQ-based methods. One such strategy that belongs to the supervised family is the LVQ1, while the SOM, the TTOSOM and the TTOCONROT primarily learn the distributions using an unsupervised learning paradigm. All four classifiers used the same values for their parameters (the ones specified in [4]). In addition, the LVQ1 and the SOM used 128 neurons, and the results shown for the TTOSOM and the TTOCONROT include *only* 15 neurons, respectively. As per our results, the TTOCONROT, using only a small percentage of the neurons used in the SOM and LVQ1 (almost 10%), outperforms their recognition capabilities in almost *all* six datasets[4]. The differences with respect to the TTOSOM are more subtle and are analyzed in a subsequent section.

Similar to the TTOSOM, we observe that the TTOCONROT offers accuracies comparable to the ones obtained by certain supervised classifiers. For instance, the results are similar to the one obtained using the KNN. However, as stated in [4], even though the KNN is internally used for labeling the neurons, this is done only once, and applies to only a small subset of the neurons, which represent a small fraction of the total number of instances, i.e., those which are involved in the computations of the KNN every time a query is performed.

Another advantage that the scheme presents is its "semi-supervised" nature, that allows it to associate the neurons with a class label using only a minimum number of tagged instances. In cases when these samples are scarce (but when the unlabeled samples are abundant), it has been shown that other schemes that belong to the same "semi-supervised" family, yield competitive results as pointed out in [10] and as our results in [4] demonstrate.

[4] The table show the results for the case when we have only 15 neurons. But if the number of neurons is increased to 127, the accuracy is superior in all the VQ-based algorithms.

3.3 Effect of the Number of Neurons

We now consider the effect of varying the number of neurons involved in the TTOCONROT tree. To test this, we trained the TTOCONROT with the configuration presented in [4], and steadily augmented the size of the respective tree. Analogous to the experimental settings used in [4], we permitted the starting value for the radius to be twice the depth of the tree, so as to ensure that all the neurons are initially considered as part as the BoA.

Table 2 shows the accuracies obtained by the TTOCONROT, where in the respective column, the specific tree size is systematically increased. The respective graphical curves are illustrated in Figure 1. To cite an example, Table 2 shows that the TTOCONROT classifier, using 127 neurons, predicts with an accuracy of 55.60%, the actual category of the instances belonging to the *winequality* dataset. The experiments use an analogous parameter configuration in which we systematically increase the size of a full binary tree with depths ranging from 3 to 12. Observe that the tree was restricted to be binary, even though it could have been of an arbitrary size. The reason for this was the TTOCONROT nature which is constrained by a BST structure[5]. Trees with small size were used to test the capabilities of the classifiers with a very condensed representation of the feature space. On the other hand, trees with a larger size were utilized so as to observe the effect of adding artificial data points which, in some cases was even greater than the number of sample points themselves. These artificial points attempted to preserve the original properties of the feature space.

Dataset ↓ Neurons →	**7**	**15**	**31**	**63**	**127**	**255**	**511**	**1023**
glass	51.87	53.74	63.08	67.76	68.22	66.36	66.36	67.29
iris	92.00	94.00	92.67	94.67	93.33	94.67	94.00	94.00
wdbc	88.23	93.32	94.55	95.96	94.55	95.08	96.13	95.43
wine	91.01	96.07	97.19	94.38	97.19	97.19	96.07	95.51
winequality	53.85	53.60	53.28	54.72	55.60	56.85	56.66	58.79
yeast	50.74	51.08	53.23	55.73	55.32	50.27	51.21	52.29

Table 2: The accuracy of the TTOCONROT as the number of neurons increases.

[5] We are currently investigating the generalization of the CONROT, which will allow rotations on trees with an arbitrary number of children per node.

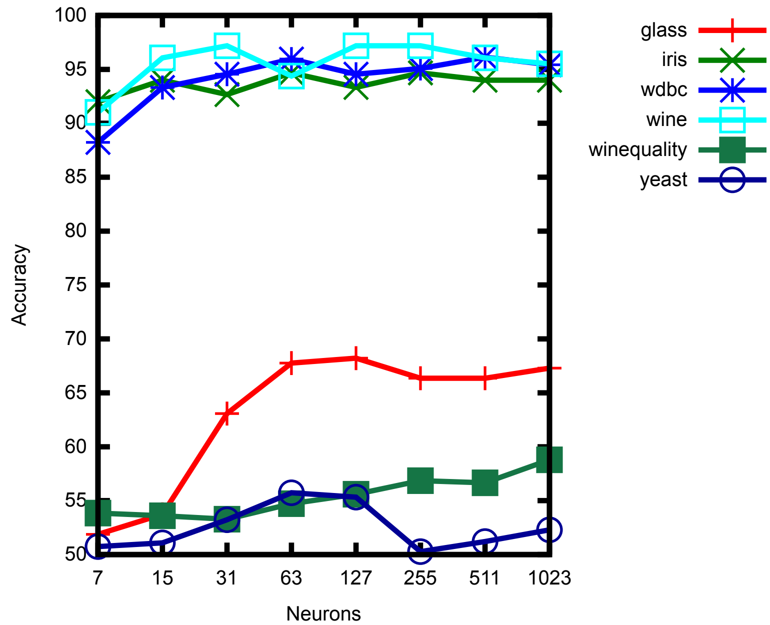

Fig. 1: The accuracies for the different datasets as obtained by using the TTOCONROT-based classifier and an increasing number of neurons.

3.4 Difference of classifying with and without Conditional Rotations

We have also investigated the effect of the rotations and studied how the accuracies vary as the number of neurons is increased.

The results for the *wine* dataset when the number of neurons is increased are shown in Table 3. In the table, each row presents the accuracies obtained by using a specific tree size, and each column indicates the accuracies obtained by the TTOCONROT and the TTOSOM, respectively. In order to verify if one strategy performs better than the other, we have computed the average accuracy and ranking indices. As per our observations, both algorithms possess a similar pattern. The maximum accuracy obtained in both cases is 97.19%. However this "peak" is reached by the TTOSOM only once (when using 127 neurons), while the TTOCONROT-based classifier achieve this in 3 instances, i.e., when using 31, 127 and 255 neurons respectively. In this regard, it is worth mentioning that based on this results, only the BN, which belongs to the "supervised" family, could outperform this accuracy, obtaining in that case, 98.88%, i.e., only a fraction better than the results obtained by our proposed methods. It is remarkable that the TTOCONROT was able to provide almost the highest accuracy possible, in comparison to the state-of-the-art classifiers included in our study, using only a limited number of 31 neurons. The fact that this result can be replicated by using a larger tree, further demonstrates the consistency of the method. From our perspective, this evidence suggests that the user does not need to know *a priori* the exact number of neurons required to train the tree effectively.

Neurons	TTOCONROT	TTOSOM
7	91.01	94.94
15	96.07	95.51
31	97.19	95.51
63	94.38	96.07
127	97.19	97.19
255	97.19	96.63
511	96.07	96.07
1023	95.51	96.63
2047	96.07	96.63
4095	96.07	96.07

Table 3: *Wine* dataset – Accuracy rate in % obtained by using the TTOCONROT and the TTOSOM as the size of the tree is increased.

Another dataset that we have considered belongs to the same problem domain, i.e., the *winequality* dataset. However, the latter presents a harder classification problem, in which the state-of-the-art supervised classifiers provide accuracy rates which are roughly between 50% and 60%. Figure 2 illustrates the differences in accuracy obtained by the TTOCONROT and the TTOSOM as the size of the tree is increased.

Observe that in both cases, the classifiers have a tendency to increase their recognition capabilities as the number of neurons is increased. However, we observe that the TTOCONROT presents an almost monotonic non-decreasing behavior. From this behavior we believe that when solving practical problems, it is worth training the classifier with trees that possess even more neurons than the number of training samples. From our experiments, we infer that it is possible to improve the accuracy rates, if additional computational power, time and/or space are available. We believe that this occurs because the TTOCONROT tree also effectively covers those regions where no samples lie, and this is used by the classifier to accurately predict the class labels of those regions when labeled instances become available.

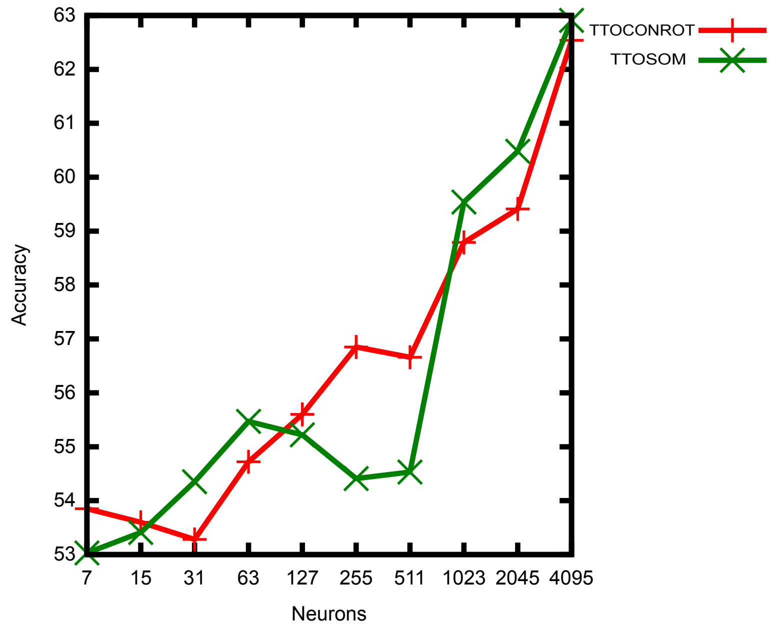

Fig. 2: The *winequality* dataset is learned using the TTOCONROT-based classifier and the TTOSOM-based classifier. In each case the number of neurons is increased systematically.

4 Conclusions

This paper has presented the design and experimental analysis of a Patter Recognition (PR) scheme based on the Tree-Based Topology Oriented SOM using Conditional Rotations (TTOCONROT). The approach utilizes the tree-based neural network to learn the distribution of *all* the samples available, regardless of the fact that their labels are known, and then utilize a set of labeled samples (expected to be scarce), to categorize the regions of the feature space. In particular, the proposed scheme constrains the neural tree as per the laws of the field of Adaptive Data Structures (ADS).

Our experiments demonstrated that, the TTOCONROT-based classifier is able to sometimes outperform state-of-the-art classifiers that use the supervised learning paradigm, i.e., those which are unable to learn from unlabeled samples. This concurs with the results of other researchers who observe that under certain scenarios, semi-supervised schemes like the one presented in this paper, can lead to performance levels that are comparable to the ones obtained by true supervised methods [10]. Particularly, in most of our experiments, trees whose sizes are only a small fraction of the cardinality of the dataset, are sufficient to obtain accuracies comparable to the ones provided by the best supervised classifiers.

Additionally, we have performed a meticulous analysis to identify the advantages of incorporating the neural rotations provided by the TTOCONROT. To do this, we compared the results with the TTOSOM-based classifier (presented in [6]), using analogous parameter settings and using different tree sizes. Our results showed that regardless of the inclusion of the rotations, competitive accuracy rates can be obtained. Moreover, our experiments also suggest that in certain cases, the rotations lead to accuracy rates that increase in a smoother manner, in comparison to the ones obtained by the TTOSOM-based classifier, as more neurons are incorporated in the tree.

References

1. C. A. Astudillo and B. J. Oommen. A novel self organizing map which utilizes imposed tree-based topologies. In *6th International Conference on Computer Recognition Systems*, volume 57, pages 169–178, 2009.
2. C. A. Astudillo and B. J. Oommen. On using adaptive binary search trees to enhance self organizing maps. In A. Nicholson and X. Li, editors, *22nd Australasian Joint Conference on Artificial Intelligence (AI 2009)*, pages 199–209, 2009.
3. C. A. Astudillo and B. J. Oommen. Imposing tree-based topologies onto self organizing maps. *Information Sciences*, 181(18):3798 – 3815, 2011.
4. C. A. Astudillo and B. J. Oommen. On achieving semi-supervised pattern recognition by utilizing tree-based SOMs. *Pattern Recognition*, 46(1):293 – 304, 2013.
5. C. A. Astudillo and B. J. Oommen. Fast BMU search in SOMs using random hyperplane trees. In Duc-Nghia Pham and Seong-Bae Park, editors, *PRICAI 2014: Trends in Artificial Intelligence*, volume 8862 of *Lecture Notes in Computer Science*, pages 39–51. Springer International Publishing, 2014.
6. C. A. Astudillo and B. J. Oommen. Self-organizing maps whose topologies can be learned with adaptive binary search trees using conditional rotations. *Pattern Recognition*, 47(1):96–113, 2014.
7. C. A. Astudillo and B. J. Oommen. Topology-oriented self-organizing maps: A survey. *Accepted in Pattern Analysis and Applications*, 2014.
8. R. P. Cheetham, B. J. Oommen, and D. T. H. Ng. Adaptive structuring of binary search trees using conditional rotations. *IEEE Trans. on Knowl. and Data Eng.*, 5(4):695–704, 1993.
9. A. Frank and A. Asuncion. UCI machine learning repository. `http://archive.ics.uci.edu/ml`, 2010.
10. B. Gabrys and L. Petrakieva. Combining labelled and unlabelled data in the design of pattern classification systems. *International Journal of Approximate Reasoning*, 35(3):251 – 273, 2004. Integration of Methods and Hybrid Systems.
11. X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.
12. X. Zhu and A. B. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan & Claypool Publishers, January 2009.