ISSN 2278-3091
Volume 8, No.1.2, 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse0681.22019.pdf
https://doi.org/10.30534/ijatcse/2019/0681.22019

Kohonen Network with Parallel Training: Operation Structure and Algorithm

Vladyslav Diachenko[1], Oleksii Liashenko[1], Bnar Fareed Ibrahim[2], Oleg Mikhal[1], Yurii Koltun[3],
[1]Department of Electronic Computers, Kharkiv National University of Radio Electronics, Ukraine,
vladyslav.diachenko@nure.ua
[2]Department of Computer Network, Lebanese French University, Erbil, KR-Iraq,
bnar.fareed@lfu.edu.krd
[3]Department of Information and Network Engineering, Kharkiv National University of Radio Electronics, Ukraine,
yurii.koltun@nure.ua

ABSTRACT

The article proposes the modified Kohonen network with search algorithm of winning neuron under condition of parallel presentation of several pattern unit. The ongoing productivity gains are promising in the light of multi-core multiprocessor computer system. The training set size is determined by use of elements of static modeling.

Key words: Kohonen maps, parallel training, distributed data processing, neural network.

1. INTRODUCTION

There are a number of technical applications, in particular in connection with environmental issues, in which information from a sensor system, each of which is not attainable or extremely limited in serviceability. For such systems, reconfigurable systems for collecting and preprocessing information are appropriate [1]. In them, individual sensors individually develop their resources and are decommissioned. When the critical level of losses is exceeded, the system is supplemented with new lots of sensors. To ensure reliability during the operational period, it is advisable to pre-process information directly in the sensor, such as clustering over a discrete set of states. This gaves the new impetus of distributed data processing (DDP) development – a principle that served as the foundation of the neural networks (NN). The relevant applied direction are said to be related to information support globalization and to intensive use of processing parallelism [2, 3, 9]. The very important elements of these directions are tasks such as pattern recognition, situations' identification, decision-making, etc., undertaken in dynamic environment. The impact of external influencing factors (that may be uncorrelated and different-scale according to time and degree of influence) determines the advisability of developing of self-adaptive and self-organizing systems. However, at the same time, the adaptability, as paradigm of technical system interaction with each other and environment, acquires qualitatively different nature and fundamentally different feasibility in connection with DDP advanced features. Its' very promising the further development of self-organizing NN with lateral connections; particularly – Kohonen networks (KN) [4]. They are the very effective clustering mechanism: basic data regionalization for several pairwise disjoint areas according to the inspected specifics' level. The cluster approach effectiveness is based on obtained data interpretability and visualization. In this way the attractiveness of clustering method with the use of KN lies in new information content implementing about system or phenomenon under analysis. Clustering, as a research method, is most effective at data processing initial stages. Drilling down the databases' process particularly allows locating ill-founded data, in order to identify additional contributing factors at a later stage. Of particular interest is the clustering in the absence of a priori requirements for localized clusters' quantity and configuration. The results obtained automatically become the phenomenon under analysis model framework.

In [5] there are considered simplistically certain aspects of adaptive parallel procedure of SN training, ensuring a number of advantages to the training productivity and efficiency compared to the classic case. The present article deals with structural principles of SN with parallelized training, easy to the implementation in computer system with parallel processing elements, particularly on personal computer with multi-core microprocessor. Such NN is very promising to the use as the parsing structure, especially with clustering tasks reliance.

2. KN CLASSICAL CASE

NN class - unsupervised machine learning – provides a means of solving problems such as large data volumes classification, organization and visualization. KN are one of this class NN representative. SN belongs to self-organizing networks, that doesn't receive any information about intended output during the signal input receipt. During the training process the self-organizing network splits producing signal input into classes, building relevant topological maps. Thus, KN potentially implements the automated methods of new structures determinations in data store: exploratory data analysis (cluster neighborhood recognition and determination) and new phenomena detection (data recognition and assigning to the specific clusters; if observation that differs from other known samples comes up, it isn't classified, in other words, they identify its novelty).

Due to the unsupervised learning KN, as data analysis instrument, is particularly effective in lack of a priory requirements for clusters' quantity. In case of two-dimensional (m×n) receptive field A: $\{a_{11}, a_{12},\ldots, a_{ij},\ldots, a_{mn}\}$ the map learning includes the next cyclic-repeated operations:

– Vectors presentation to the pattern units (PU), that is parallel to each neuron from receptive field;

– Formation for each (further) PU each neuron reaction;

– winning neuron recognition $x \in A$, which personal vector is component-wise minimally distant from presented PU vector;

– Vector correction active of neurons of nearest neighbor x, based on their distance and learning process step number (k), according to the next expression statement:

$$a_{ij}(k+1) = a_{ij}(k) + \alpha(k)\, f_{ij}(k)\, (x(k) - a_{ij}(k)). \qquad (1)$$

The amplifier gain α(k) is equal to 1 in the beginning of learning, and is setting to 0 during the learning process (with the growth of k). The monotone decreasing function f(k) describes neighborhood neurons' membership degree to the term "winning neuron neighborhood". In (1) fij(k) – this function value is for aij neuron on step k. For the purpose of constancy of system performance during the learning process, α(k) decreases with growth of k, and fij(k) "constrict" around x(k). If these conditions are not fulfilled, every next PU presentation would be "getting out of form" the receptive field, trained by preceding presentations. If conditions are met – there will be grouping (clustering) of receptive field neurons' vectors, as a result of training (learning), around some sets of values, most unique to training set. Looking ahead, in the standard operating mode (in operating mode of trained KN) undefined given vector (unit) will be relevant to one of the clusters.

3. ADAPTIVE PARALLEL PROCEDURE

The described sequential procedure of KN training differs by its awkwardness. The winning neuron that was detected on the step k is "relevant" only for his neighborhood. If for PU (k+1) the winning neuron is far from the previous one, procedure of its neighborhood correction (1) won't affect the previous training step results. Similarly, the training step (k+2) may have the winning neuron at large distances from the previous two neurons. In consequence, relevant PU, in the context of their real equal relevancy for this KN learning process, could be presented randomly. In consequence, the procedure with α(k) and "constrict" fij(k) according to (1) wrenches PU real relevancy for training process, by introducing additional ("parasitical", unmotivated) dependence from PU order of presentation.

It's worth noting, that the parallelism of input and the KN receptive field's inception data processing fundamentally contradicts to the sequenced character of unit presentation and KN learning. The controversy removal by communication structure modifying and KN learning procedure, potentially provides for positive effect: picks up training speed and, by extension, the learning process effectiveness and KN post exposure. In an appropriate manner, data organization structure, gained and stored by KN, also changes. In consequence, KN field-performance data also changes after learning process.

The [6] offer the option of KN learning procedure duration reduction, containing processing parallelism. The PU presentation, neurons' reaction organization of receptive field and winner neuron identifying – these are the fundamentally sequence actions, because each step of learning procedure can contain by definition only one winner neuron. Neurons vectors correction has somewhat another purpose. After the k PU presentation the winner neuron is:

$x(k) \in \{a_{11}, a_{12},\ldots, a_{ij},\ldots, a_{mn}\}$; after (k+1) unit presentation – the winner is x(k+1).

The legal situation of correction areas' disjointness is also permitted:

$$F(k) \cap F(k+1) = \varnothing; \quad F(k), F(k+1) \subset A; \quad f(k), f(k+1) > 0. \qquad (2)$$

In such case, at larger n and m a non-void intersection is hardly possible. Suppose g units are presented sequent, and each of these units is governed by its winning neuron. Suppose, with regard for (2), their disposition range A is that the correction areas F(1), F(2),…,F(g) are mutually disjoint:

$$F(p) \cap F(q) = \varnothing; \quad p, q \in (1,2,...,g); \quad p \neq q. \qquad (3)$$

Such situation is accessible by sequence undefined PU presentation with withdrawal, if recurrent presented PU fails to satisfy (3). In this case, PU picks out units from every g cycle, which are potentially related with different clusters. In other words, the clustering procedure is additionally supported by mentioned withdrawal in the moment of PU presentation, implemented similarly to the "time rollback".

Looking ahead, for the formed g-elemental winning neurons set the summarized (combined, compositional) correction function is designed

$$\begin{aligned} f_g &= f(1) \cup f(2) \cup \ldots \cup f(h) \ldots \cup f(g) = \\ &= \min(f(1), f(2),\ldots,f(h),\ldots, f(g)), \end{aligned} \qquad (4)$$

Describing neighbor neurons' membership degree to relevant winning neurons, having g local upper limits in winner neurons' points' location. With consideration of (4) the neurons' range A (receptive field) is corrected at the time, in other words, the correction procedure is parallel for winner neurons. This procedure repeats until the PU set exhaustion, as indicated in algorithm scheme 1.

4. ADVANTAGE OF ADAPTIVE PROCEDURE

This approach adaptively lies in the fact that the number g is set initially, according to results of preliminary probabilistic estimation that is individual for every concrete objective situation, with consideration of (2) and (3). However, during the next PU set organization, those Pus that fail to satisfy concrete situation, are delayed and presented in next learning circles. It's easy to see, that every cycle has g presentation of PU amplitude variation (coefficient α(k) in (1)) and neighborhood dimensional changes, surrounded fij(k), is not necessary, because the separate winner neurons don't compete with each other for the neighborhood within one group.

As a result, KN training isn't such long-duration and becomes more efficient: it takes inferior training PU set, or on condition of training set fixed volume the training result is more stable at the level of clusters' configurations product ability (figure 1).

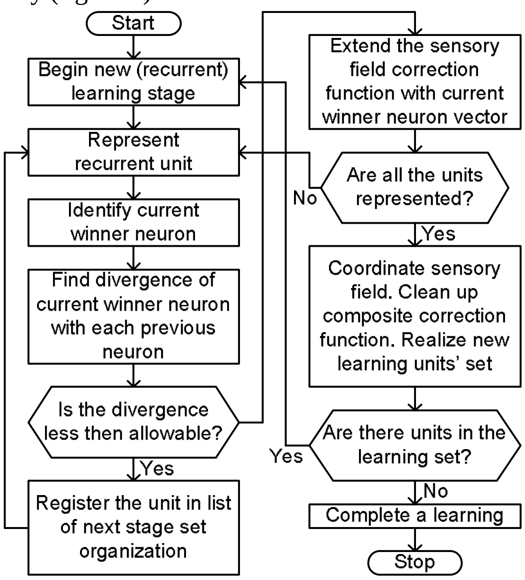

Figure 1: Training algorithm scheme of modified Kohonen network

4.1 PROBABILISTIC EVALUATION

For probabilistic evaluation of PU g-set volume, it's appropriate to carry out the statistical modeling. Let's consider the case of two-dimensional sensory field m×n in size, with permanent configuration of winner neuron neighborhood (coefficient fij in (1)), that includes only 4 closest (neighbor) neurons. In this case tree structurally different neurons types:

A. Corner neurons (if such neuron becomes the winner one, only two neighbor neurons change their weight values). The number of such neurons is k1 = 4;

B. Edge neurons (three neighbor neurons change their weight values). The number of these neurons linearly depends on network size: k2 = 2 * (m + n – 4);

C. Central neurons (four neighbor neurons change their weight value). Their number quadratically depends on network size: k3 = m * n – 2 * (m + n – 2).

Simplifying model situation: suppose m = n. Then k1 = 4; k2 = 4 (n – 2); k3 = n2 – 4 (n – 1). With the help of direct calculation we find that at n ~ 40 the weight (k1+k2) comes to approximately 10% from k3; at n ~ 400 – approximately 1%; at n ~ 4000 – approximately 0,1%; which is to say that edge effects flatten only in case of essential extension of sensory field. That's why probabilistic evaluation simplification possibilities are determinate. It's required to calculate probabilities for nonintersection areas, which primarily depend on winner neuron location area, because the firing neurons' area can have different sizes. After that it's required to calculate the quantity of all possibilities with the first winner neuron. Such neuron can be not only the corner neuron, but also edge and central neurons. Let's consider possibilities for pointed cases:

A. corner neuron: P = 4 / (m * n);

B. edge neuron: P = 2 * (m + n – 4) / (m * n);

C. central neuron:

$$P = (m * n - 2 * (m + n - 2)) / (m * n) = 1 - (2 * (m + n - 2) / (m * n)). \quad (5)$$

Further the probability of non itersection can be calculated on the premise that it's necessary to take account of already firing neurons area locations, i.e. that the next winner neuron and its neighbor neurons won't get into such areas. Relevant analytical expressions stay simple only within the statement of nonempty intersection probability smallness, i.e. for the large-sized sensory fields. Maximum permissible weights estimation of KN parallel learning is easier to interpret by statistical modeling methods that allow to take account of receptive fields configuration (in case $m \neq n$).

5. STATISTICAL MODELING APPLICATION

The goal of running modelling – is the origination of statistical estimates sets for probabilities of given percent of using primer learning materials within sustainable two-dimensional statistical arrangement of winner neurons fallout on the KN sensory field. Knowing mentioned probabilities and having data about performance improvements of modified KN learning procedure, in comparison with traditional case, could be pointed out efficient volumes of PU learning sets. The minimum KN learning time could particularly serve as a criterion of optimality.

The modeling is realized with use of random number generator, reproducing winner neuron identification process. Within structural relationship the model is similar to learning algorithm (Fig.1), and is supplemented by standard blocks of statistical analysis and by results documentation. In simplified form, the modeling contains next stages:

The matrix A input of receptive field (sizes m, n and elements $a_{i,j}$, where $i \in (1, 2,\dots, m)$, $j \in (1, 2,\dots, n)$).

The quantity calculation of edge k2 and central k3 neurons according to the matrix given dimensions.

At the first stage (after first unit presentation) the quantity of possible winner neurons equals to S=m×n.

The winner neuron origination $a_{i,j}$ after PU presentation. Definition of its type: edge ($i \in (1, m), j \notin (1, n)$ or $i \notin (1, m), j \in (1, n)$); central ($i, j \notin (1, n, m)$); corner ($i \in (1, m), j \in (1, n)$). In model simplified version the activation function changes only the weights of winner neuron's neighbor neurons ($a_{i-1,j-1}$, $a_{i-1,j}$, $a_{i-1,j+1}$, $a_{i,j-1}$, $a_{i,j+1}$, $a_{i+1,j-1}$, $a_{i+1,j}$, $a_{i+1,j+1}$). Moreover the number of modified neurons is 3 for corner neurons, 5 for edge neurons and 8 for central ones. By so doing, after each unit presentation and winner neuron identification it's deducted from S the number of modified neurons, including winner neuron.

After each unit presentation, S is compared with Q threshold value. If S > Q, the unit presentation runs still; if S = Q, - the units' presentation cycle is over.
The modeling-building alternative version is also developed: repeated PU generation (pseudorandom new winner neuron coordinate dimensioning), checking its inequality with already allocated winner neurons, statistical estimation of successful placement possibility depending on presented PU number.
The graphs (Figure 2) show the simulation results for the receptor fields of neurons. Charts 1, 2, ..., 5 correspond to the allowable distances between elements 1, 2, ..., 5 units. The probability P of overlapping elements increases with an increase in the number N of displayed elements and the permissible distance of their location. The probability significantly depends on the size of the receptor field: taking into account the scale, the entire array of graphs (b) can be compressed and placed in the bottom row of the cells of the graph (a). The dependence of the probability on the asymmetry of the field (the case of m × n, m ≠ n) was not considered at this stage of the simulation.

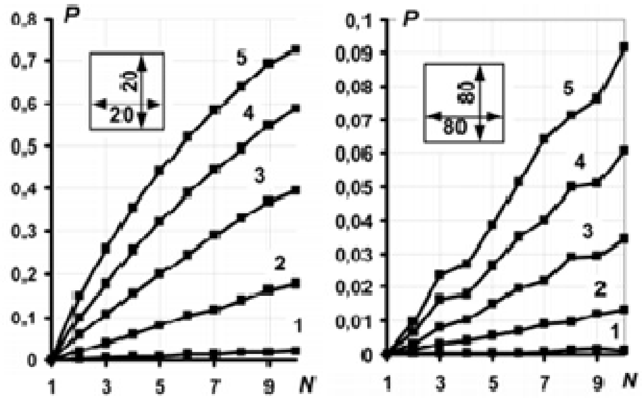

Figure 2: The probability P of overlapping objects from the number N of the object being shown for receptor fields of size
a - 20 × 20 and b - 80 × 80

The results can be interpreted as follows. According to, for example, schedule 3 (Figure 2, b) with eight consecutive presentations of objects, the probability of an overlap does not exceed 3%. For a specific software implementation with a certain gain in the performance of the learning process (see Fig. 1) in comparison with the traditional version of the Kohonen network, based on the simulation results presented in Figure 2, specific recommendations can be made to optimize the size of the training sample. The optimality criterion for a given configuration of Kohonen network parameters (receptor field format, allowable distance between elements, etc.) can be formulated, in particular, as maximizing the fragment of a training sample (minimizing the fragmentation of the training

6. CONCLUSION

Data capacity of Kohonen network is defined by sensory field volume. Furthermore, the learning set essentially goes up. The perspective direction of network improving productivity at the stage of learning is the offered algorithm of parallel training. The winner neuron identification procedure is carried out parallely for several pattern units, particularly with use of multi-core and multiprocessor computer systems.
The research is at an advanced stage: the key elements of Kohonen network modified learning algorithm are brassboarded, the backbone nodes of statistical node are developed. The realization of compos able (combined) function of sensory field correction needed further consideration.

REFERENCES

1. Churyumov, G., Tkachov, V., Tokariev, V., Diachenko, V, **Method for Ensuring Survivability of Flying Ad-hoc Network Based on Structural and Functional Reconfiguration**, IEEE International Conference of Information Technologies and Security (ITS 2018, Kyiv, Ukraine), pp. 64-76, November 2018. https://doi.org/10.1007/s10559-011-9285-7
2. Rudenko, O., Bezsonov, A., Liashenko, A., Sunna, R, **Approximation of Gaussian basis functions in the problem of adaptive control of nonlinear objects,** Journal "Cybernetics and Systems Analysis", vol.47 №1, 2011, pp. 3-13.
3. Andrews J.R., **Foundations of multithreaded parallel and distributed programming,** Publishing house «Williams", pp.512, 2003.
4. Kohonen, T, **Self–organization and associative memory**, Series in Information Sciences, vol. 8. Berlin: Springer verlag, 1984.
5. Michal, O., Rudenko, O, **Modeling on Petri net of virtual computer device for study of local-parallel algorithms' effectiveness**, Journal "Control systems and machines", № 3, pp. 18-28, 2003.
6. Diachenko, V., Mikhal, O., **Modeling the process of interactive learning intelligent self-organizing systems,** Journal "Systems of control, navigation and communication", №2, pp.102-107, 2017.
7. Amin Salih Mohammed, D Yuvaraj, M Sivaram, V Porkodi, **Detection And Removal Of Black Hole Attack In Mobile Ad Hoc Networks Using GRP Protocol,** "International Journal of Advanced Research in Computer Science", Vol. 10, Issue-6, 2018. https://doi.org/10.26483/ijarcs.v9i6.6335
8. Priyanka Thakur and Dr. Rajiv Shrivastava, **"A Review on Text Based Emotion Recognition System"**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 7. Issue 5, 2018. https://doi.org/10.30534/ijatcse/2018/01752018
9. Revathy. S, Saravana Balaji. B, N.K.Karthikeyan, **"From Data Warehouses to Streaming Warehouses: A Survey on the Challenges for Real-Time Data Warehousing and Available Solutions"**, International Journal of Computer Applications, Volume 81, No 2, pp.15-18, November 2013. https://doi.org/10.5120/13984-1990