International Journal of Electrical and Computer Engineering (IJECE)
Vol. 7, No. 6, December 2017, pp. 3570~3577
ISSN: 2088-8708, DOI: 10.11591/ijece.v7i6.pp3570-3577

Comparative Study of Neural Networks Algorithms for Cloud Computing CPU Scheduling

Gibet Tani Hicham, El Amrani Chaker, Elaachak Lotfi
Laboratory of Informatics Systems and Telecommunications (LIST), Abdelmalek Essaadi University, Morocco

Article Info

Article history:

Received May 24, 2017
Revised Jun 22, 2017
Accepted Jul 10, 2017

Keyword:

Cloud computing
CPU scheduling
Neural networks

ABSTRACT

Cloud Computing is the most powerful computing model of our time. While the major IT providers and consumers are competing to exploit the benefits of this computing model in order to thrive their profits, most of the cloud computing platforms are still built on operating systems that uses basic CPU (Core Processing Unit) scheduling algorithms that lacks the intelligence needed for such innovative computing model. Correspdondingly, this paper presents the benefits of applying Artificial Neural Networks algorithms in regards to enhancing CPU scheduling for Cloud Computing model. Furthermore, a set of characteristics and theoretical metrics are proposed for the sake of comparing the different Artificial Neural Networks algorithms and finding the most accurate algorithm for Cloud Computing CPU Scheduling.

Corresponding Author:

Gibet Tani Hicham,
Laboratory of Informatics Systems and Telecommunications (LIST),
Abdelmalek Essaadi University,
Route Ziaten, B.P. 416 Tangier, Morocco
Email: gibet.tani.hicham@gmail.com

1. INTRODUCTION

CPU (Core Processing Unit) scheduling is the process of assigning compute resources to a particular job or task submitted for execution on a specific system. The allocation of compute resources follows a predefined scheduling algorithm on the system (generally declared on the operating system kernel). Up to this moment, the majority of computer systems are using simple scheduling algorithms that were defined on the past 20 years (such as First Come First Served, Round Robin or Priority scheduling) and that still give remarkable results for daily use bases with some minor modifications. However, with the extraordinary advance of computer engineering, the major shift of the world into the internet and with the birth of Cloud Computing, these basic CPU scheduling algorithms are starting to become deprecated.

The major problem with the existing CPU scheduling algorithms is the low performance related to the time-consuming jobs that comes with the Cloud model of computing (Offering IT resources as services: Infrastructures, platforms and Applications), therefore they produce a poor response time that is not suitable for large-scale environments. On the same context, many investigators are promoting Artificial Neural Networks (ANN) as a solution to optimize the existing algorithms, thus assisting Cloud Computing providers and users make intelligent decisions regarding their investments on this outstanding technology. Several neural networks algorithms are available and comparing them in the aim of choosing the best algorithm for CPU scheduling is a complicated mission giving the vast application fields of neural networks. In this paper, a set of practical features has been considered to assess and evaluate the existing neural networks algorithms and foremost choosing the most appropriate algorithm for Cloud Computing CPU scheduling.

2. CLOUD COMPUTING PRESENTATION

CloudComputing is a model for enabling ubiquitous, on-demand network access to a shared pool of configurable computing resources [4] [5]. By studying this new service delivery model originates the challenge of managing hundreds of thousands of users and applications requests. Therefore, a Cloud Computing provider should consider intelligent infrastructure deployment in order to establish a Cloud Computing offer, which insures transparency, scalability, security and foremost celerity. A Cloud Computing offer range from offering an end user a specific IT infrastructure (storage, servers, network…), to proposing complicated application and software solutions (CRM, ERP…) and all of this is organized on a layered architecture (Figure 1).

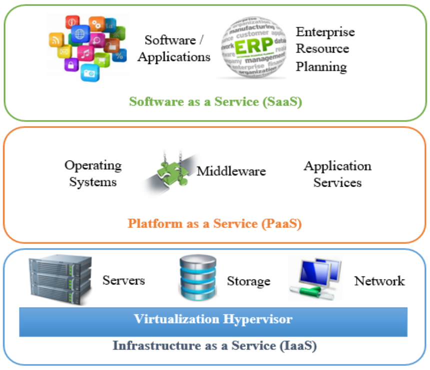

Figure 1. Cloud Computing Layered Architecture and Delivery Model

One of the central cloud providers' objectives is the provisioning of physical resources for users or a specific application. Thus, a cloud provider should select and control the allocation of the correct resource whether a cloud user request it as a service (IaaS) or a cloud application of the higher layers needs it (PaaS or SaaS).

3. NEURAL NETWORKS AND ARTIFICIAL INTELLIGENCE

3.1. Overview

Artificial Neural Networks (ANN) is an information-processing paradigm that simulates the human brain. It was designed to mimic the way the human brain executes a specific task or function [6] [7]. This kind of networks "Figure 2" is composed of several calculations unites called neurons, which are combined in layers and operating in parallel. The information will be propagated layer to layer, from the input layer to the output layer. The ANNs have the ability to store empirical knowledge and make it available for the users. The knowledge of the network will be stored in synaptic weights, obtained by the process of adaptation or learning.

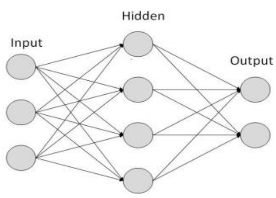

Figure 2. Artificial neural network

Based on the weights and transfer functions [7], the activation value is passed from node to node. Each node sums the activation values it receives, and then modifies the value based on its transfer function. The activation procedure follows a feed forward process and the difference between the predicted value and the actual value (error) will be propagated backward by apportioning them to each node's weights according to the amount of the error the node is responsible for (e.g., gradient descent algorithm [8]), as shown in Figure 3.

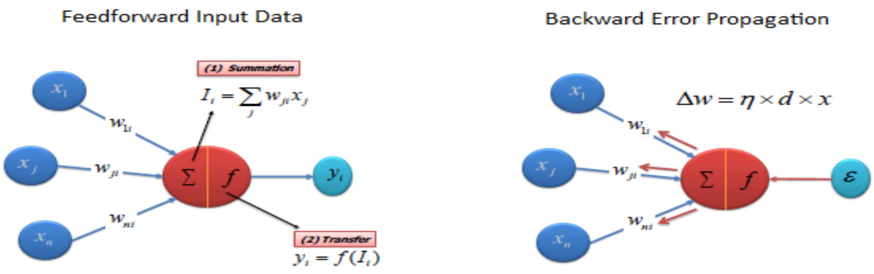

Figure 3. Feed forward input data and backward error propagation

3.2. Activation Function

The Activation function [8] translates the input signals to output signal. There are several kinds of activation functions: Unit step, Sigmoid, Gaussian, etc. (Figure 4).

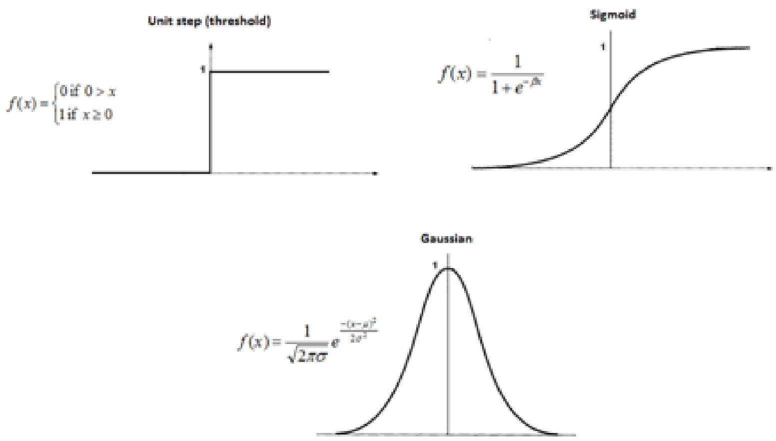

Figure 4. Activation functions Unit step, Sigmoid, and Gaussian

3.3. Types of Artificial Neural Networks

Artificial Neural Networks [7] [8] are generally classified into feed-forward and feedback networks. The Feed-forward [7] network is a non-recurrent network, which contains inputs, outputs, and hidden layers; the signals can only travel in one direction. Input data is passed onto a layer of processing elements where it performs calculations. It includes Perceptron and Radial Basis Function networks. Feed-forward networks are used often in data mining. Multi-layer [7] Perceptron “Figure 5” is one of the feed-forward networks; it has the same structure of a single layer Perceptron with one or more hidden layers. The learning algorithm used in this network is the back propagation [9]. It consists of two phases: the forward phase where the activations are propagated from the input to the output layer, and the backward phase, where the error between the observed actual and the requested nominal value in the output layer is propagated backwards in order to modify the weights and bias values.

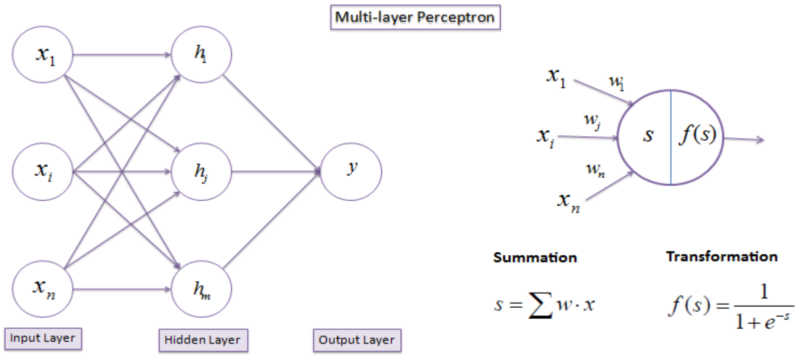

Figure 5. Multi –layer Perceptron

Backward propagation: Propagates the errors backward by apportioning them to each unit according to the amount of the error each unit is responsible for, see Figure 6.

1. Error in any ***output*** neuron

$$d_o = y \times (1-y) \times (t-y)$$

2. Error in any ***hidden*** neuron

$$d_i = y_i \times (1-y_i) \times (w_i \times d_o)$$

3. Change the ***weights***

$$\Delta w = \eta \times d \times x$$

Figure 6. Error propagation

The Feed-back [10] network has feed-back paths, meaning they can have signals traveling in both directions using loops. All possible connections between neurons are allowed. Since loops are present in this type of networks, it becomes a non-linear dynamic system, which changes continuously until it reaches a state of equilibrium. Feed-back networks are often used in associative memories and optimization problems where the network looks for the best arrangement of interconnected factors.

3.4. Training Techniques

Training techniques or learning algorithms have a significant impact on the performance of the neural network. The choice of a suitable learning algorithm is therefore application and infrastructure dependent. There are varieties of learning algorithms that can be used to train a neural network, below is the description of some algorithms that will be used in this comparative study.

Back-propagation: an abbreviation of backward propagation of error algorithm [12] was originally introduced in the 1970s. It is a method of training artificial neural networks based on the gradient descent [13], one of the optimization methods. It calculates the gradient of a loss function with respect to all the weights in the current network. The algorithm is described below:

Table 1. Back-Propagation Training algorithm

1.	Initialize weights to small random values
2.	Choose input pattern
3.	Propagate signal forward through network
4.	Determine Error (E) and propagate it backwards through network to assign credit to each unit
5.	Update weight by means gradient descent :

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}}$$

Backpropagation action can cause changes in the weight of the presynaptic connections, there is no simple mechanism for an error signal to propagate through multiple layers network, and it is among the disadvantages of this learning method.

Resilient Propagation: Heinrich Braun created resilient propagation "Rprop", an abbreviation of resilient back-propagation, in 1992 [14]. It is a learning heuristic for supervised learning in feed-forward artificial neural networks. "Rprop" is considered the best algorithm, measured in terms of convergence speed, accuracy and robustness with respect to training parameters [16].

"Rprop" is similar to the back-propagation algorithm. However, it has two main advantages over back propagation:

- Training with "Rprop" is often faster than training with back propagation.
- "Rprop" does not require the specification of any free parameter values, as opposed to back propagation that needs values for the learning rate.

The main disadvantage of "Rprop" is that it is a more complex algorithm to implement than back propagation.

Genetic algorithm training: The Genetic algorithms [16] are algorithms for optimization and learning based on several features of natural selection. They can also be used for training of artificial neural network. The design of the algorithm was inspired by observation of natural evolution process. The genetic algorithm performs several operations including [17]:

Table 2. Genetic training algorithm

1.	Random initialization of the preliminary population.
2.	In-loop evaluation of every chromosome by measuring its fitness.
3.	Comparison with the minimal desired fitness.
4.	Selection of the fittest subset of chromosomes.
5.	Perform crossing-over, which is exchange of features from the selected subset of chromosomes.
6.	Introduce mutations, which are random changes applied to randomly chosen features of the chromosomes.
7.	Return to the 2nd point.

During training process, every chromosome on the genetic algorithm evolves from all the connection weights from the artificial neural network.

Other training methods: There are other training methods that can be used to train several artificial neural networks, e.g. "Scaled Conjugate Gradient [18], Competitive Learning [19], Levenberg-Marquardt [20], Hopfield learning [21], etc.", most of those algorithms belong to the supervised learning family, and each of them has specific features, advantages, and disadvantages that mostly can't be adapted to CPU scheduling problematic.

4. NEURAL NETWORKS AND CLOUD COMPUTING CPU SCHEDULING

CPU scheduling is involved in each of the Cloud Computing layers (Figure 1), whereas it will affect significantly the platforms performance (Operating System), middleware and software responses. Hence, choosing the accurate algorithm for CPU scheduling will have a massive impact on the Cloud delivery response time and presents a finer alternative to expanding the infrastructures in order to promote celerity, thus reducing costs relative to acquiring the new infrastructures, management, provisioning, monitoring and troubleshooting.The finest CPU scheduling algorithm on a Cloud Computing model should predict the amount of time (Time Quantum) that is essential for each task submitted for execution in respect to the following directions:

- ✓ Reduce the number of context switches (the amount of times the CPU switches from a task to another)
- ✓ Reduce the average amount of time that a task spent on the waiting list.
- ✓ Reduce the average amount of time necessary to carry out the execution of a task.

By studying these guidelines and the existing CPU Scheduling algorithms, we were able to emphasize the following ANN key criteria that will affect the Cloud Computing service delivery model:

- ✓ Response Time (S_1): The amount of time necessary to produce a result.
- ✓ Training methods (S_2): Support of ANN existing training methods
- ✓ Training duration (S_3): The amount of time required to coach the algorithm before it can start taking decision.
- ✓ Integration (S_4): Simplicity of coding and integration with existing platforms (Operation systems, Hypervisors, Cloud provisioning platforms).

A theoretic weight that varies from 0 to 1 has been given to each one of the criteria mentioned above that represent its importance to solving the scheduling problematic:
Response Time: $w_1 = 0.35$, Training methods: $w_2 = 0.25$, Training duration: $w_3 = 0.3$, Integration: $w_4 = 0.1$

$$\sum_{n=1}^{4} wn = 1$$

5. RESULTS AND DISCUSSION

The evaluation considered in this paper consists of evaluating the type of artificial neural networks based on the criteria described on the previous section. According to literature, there are a variety of ANN Types and each one of them has proven its capacity in one or multiple fields. The challenge is to find the ANN type that can be adapted the most to CPU scheduling for cloud computing and this by reviewing the Artificial Neural Network algorithms applications on the field:

Table 3. ANN Applications

Type OF ANN	Application	Adapted for CPU Scheduling / System resources management
Multi-layer Perceptron [22]	Supervised learning[23] Pattern recognition [24] Speech recognition [24] Image recognition [24] Machine translation [24]	"Multi-layer Perceptron" has been used to optimize job scheduling results [3].
RBF network [25]	Mac-Key Glass Chaotic time series [26] Logistic Map [27] Prediction Non Linear system [26] [27] Forecasting [28]	RBF neural network is used in the prediction of the time and resources consumed by applications [40]
Kohonen self-organizing network [29]	Meteorology, Oceanography [30] Project prioritization and selection [31]	--
Recurrent neural network [32]	Hand writing and speech Recognition [33] Computer Vision [34] Language Processing [35]	Recurrent Neural Network has been used to optimize the number of queues and quantum to decrease the response time of processes and increase the performance of scheduling. [41].
Modular neural networks [36]	Predication [37] Pattern recognition [38] Classification [39]	--

Table 4. ANN Scoring

Type OF ANN	Response Time	Training methods		Training duration	Integration
Multi-layer Perceptron	0.8	- Back-propagation - Resilient back-propagation - Genetic algorithmic	0.3	0.6	0.8
RBF network	0.7	- Gradient Descent - Kalman Filtering - Genetic Algorithmic	0.3	0.5	0.7
Kohonen self-organizing network	0.1	- Self-Organizing Map	0.1	0.1	0.1
Recurrent neural network	0.5	- Recurrent learning - Extended Kalman - Gradient descent - Global optimization	0.4	0.4	0.5
Modular neural networks	0.1	- Modular neural network training algorithm	0.1	0.1	0.1

The overall score for each algorithm is calculated as follow: S = $\sum_{n=1}^{4} wn * Sn$

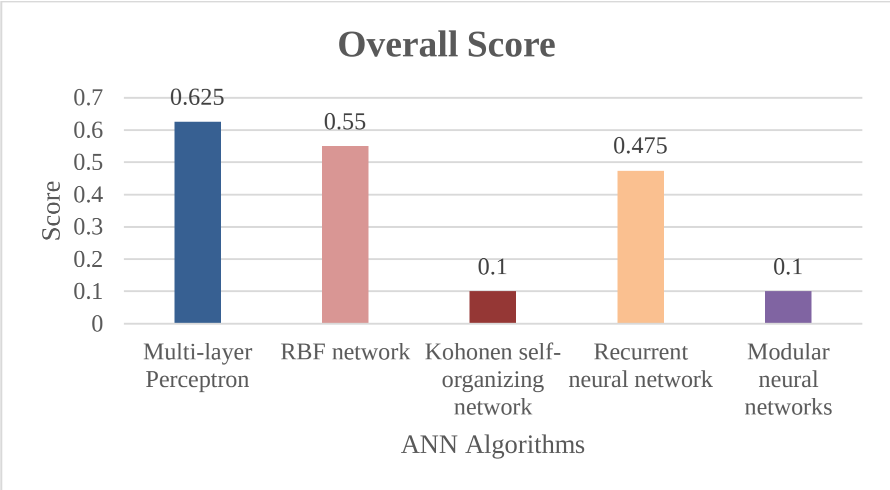

Figure 6. ANN algorithms Overall Score

According to figure 6, the Multi-layer Perceptron ANN attained the finest score, followed by RBF network and Recurrent Neural Network respectively. Therefore, Multi-layer Perceptron is the ANN type that can better answer to the problematic of CPU scheduling on Cloud Computing.

6. CONCLUSION

The study engaged on this paper is a theoretical evaluation of Artificial Neural Networks and their abilities to solve the problem related to CPU scheduling on Cloud Computing. A set of conceptual metrics have been considered to score each ANN type and training techniques and that is in regards to specific criteria used to evaluate the performance of the scheduling algorithms in the Cloud that can be resumed on reducing the average waiting time of tasks on the execution queue and stimulating the response time. In spite of the difficulties encountered in order to spot the accurate ANN type suited for the CPU scheduling challenge on the Cloud, the Multi-layer Perceptron ANN radiates as the best candidate to answer to each of the criteria considered during the evaluation and assessment.

This accomplishment will be expanded by conducting more studies and testing on the Multi-layer Perceptron ANN algorithm using specific simulators. Furthermore, an implementation of the algorithm in one of cloud computing platforms in order to assess the performance of the algorithm on real based situations.

REFERENCES

[1] F. DaríoBaptista, S. Rodrigues, F. Morgado-Dias. "Performance comparison of ANN training algorithms for classification", *Intelligent Signal Processing (WISP)*, 2013 IEEE 8th International Symposium.

[2] R. Caruana, A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms", *Proceedings of the 23rd international conference on Machine learning. ACM*, 2006.

[3] M. Maqableh, H. Karajeh, R. Masa'deh,"Job Scheduling for Cloud Computing Using Neural Networks", *Communications and Network*, 2014, 6, 191-200.

[4] C. El Amrani, K.B. Filali, K. Ben Ahmed, A.T. Diallo, S. Telolahy, "A Compartive Study of Cloud Computing middleware", *IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, 2012.

[5] G.T. Hicham, E. Chaker, "Cloud Computing CPU Allocation and Scheduling Algorithms Using CloudSim Simulator", *International Journal of Electrical and Computer Engineering*, Vol. 6, No. 4, August 2016.

[6] M. Abdella, T. Marwala, "The Use of Genetic Algorithms and Neural Networks to Approximate Missing Data in Database", *Computing and Informatics*, Vol 24, 2005, 577–589.

[7] SaedSayad: http://www.saedsayad.com/artificial_neural_network.htm.

[8] J. Skorin-Kapov, K.W. Tang, "Training Artificial Neural Networks: Backpropagation via Nonlinear Optimization", *Journal of Computing and Information Technology*, CIT 9, 2001, 1, 001–014.

[9] D.Kaul, N. Anam, S. Gaikwad, S. Tiwari, "Domain Based Categorisation Using Adaptive Preprocessing", *International Journal for Research in Applied Science & Engineering Technology*, Volume 4, February 2016.

[10] S.E. Fahlman, "An Empirical Study of Learning Speed in Backpropagation Networks", *Carnegie Mellon Report*, No CMU- Cs, pp. 88-162.

[11] S. Padmapriya, S. Andavan, "A Study on Algorithmic Approaches and Mining Methodologies In Data Mining", *International Journal of Computer Science Trends and Technology*, Volume 3, Issue 1, Jan-Feb 2015.

[12] E.R. David, E.H. Geoffrey, J.W. Ronald. “Learning representations by back-propagating errors”, *Nature International Weekly Journal of Science*, 1986.
[13] A. Snyman,“Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms”, *Springer Publishing*, ISBN 0-387-24348-8, 2005
[14] M. Riedmiller, H. Braun, “Rprop - A Fast Adaptive Learning Algorithm”, *Proceedings of the International Symposium on Computer and Information Science VII*, 1992.
[15] M. Riedmiller, H. Braun, “A direct adaptive method for faster backpropagation learning: The RPROP algorithm”, *Proc. IEEE Int. Conf. On Neural Network*, pp. 586-591, 1993.
[16] A.E. Eiben, “Genetic algorithms with multi-parent recombination”, *PPSN III: Proceedings of the International Conference on Evolutionary Computation*, 78–87, ISBN 3-540-58484-6.
[17] B.P. Jiang, “Neural Network Algorithm Optimized by Genetic Algorithm and Its Simulation”, *International Journal of Computer Science*, Issues 10, 2013, 516-519.
[18] M. Fodslette, “A scaled conjugate gradient algorithm for fast-supervised learning. Neural Networks”, *DAIMI PB 339*, 1993.
[19] R. David, D. Zipser, J.L. McClelland, “Parallel Distributed Processing”, *MIT Press*, Vol. 1, pp. 151–193.
[20] D. Marquardt, “An Algorithm for Least-Squares Estimation of Nonlinear Parameters”, *SIAM Journal on Applied Mathematics*, Vol. 11, No. 2, June 1963, pp. 431–441.
[21] Mac Kay, J.C. David, “Hopfield Networks”, *Information Theory, Inference and Learning Algorithms. Cambridge University Press*. p. 508. ISBN 0521642981.
[22] Rosenblatt, Frank, “Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms”. *Spartan Books, Washington DC*, 1961
[23] G. Cybenko, “Approximation by superpositions of a sigmoidal function”, *Mathematics of Control, Signals, and Systems*, 2(4), 303–314
[24] P.D Wasserman, T. Schwartz, “Neural networks. II. What are they and why is everybody so interested in them now?”, *IEEE Expert*, 1988, Volume 3, Issue 1
[25] H. Haviluddin, I.Tahyudin, “Time Series Prediction Using Radial Basis Function Neural Network”,*International Journal of Electrical and Computer Engineering*,Vol. 5, No. 4, August 2015.
[26] E.S. Chong, S. Chen, B. Mulgrew, “Gradient Radial Basis Function Networks for Nonlinear and Nonstationary Time Series Prediction”, *IEEE Transactions on Neural Networks*, Vol. 7, No. 1
[27] R. Zamora, D.Racoceanu, N.Zerhouni, “Recurrent radial basis function network for time-series prediction”, *Engineering Applications of Artificial Intelligence*, Elsevier, 2003.
[28] L. Yu, K. Keung Lai, S. Wang, “Multistage RBF neural network ensemble learning for exchange rates forecasting”, Neurocomputing.
[29] Kohonen, Teuvo, “Self-Organized Formation of Topologically Correct Feature Maps”, *Biological Cybernetics*. 43 (1): 59–69. Doi:10.1007/bf00337288.
[30] Y.Liu, R.H. Weisberg, “A review of self-organizing map applications in meteorology and oceanography”, *Self-Organizing Maps-Applications and Novel Algorithm Design*, 253-272.
[31] G. Zheng, V. Vaishnavi, “A Multidimensional Perceptual Map Approach to Project Prioritization and Selection”, *AIS Transactions on Human-Computer Interaction*, (3) 2, pp. 82-103.
[32] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, J. Schmidhuber, “A Novel Connectionist System for Improved Unconstrained Handwriting Recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, 2009.
[33] H. Sak, A.W. Senior, F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling”, *Proc. Interspeech*, pp 338-342, Singapore, Sept. 201
[34] Tripathi, Subarna, “Context Matters: Refining Object Detection in Video with Recurrent Neural Networks”, *arXiv preprint arXiv*: 1607.04648 (2016).
[35] Socher, Richard, Lin, Ng. Cliff, Y.Andrew,Manning, D. Christopher, “Parsing Natural Scenes and Natural Language with Recursive Neural Network”, *The 28th International Conference on Machine Learning* (ICML 2011).
[36] A. Farooq,“Biologically Inspired Modular Neural Networks”, *PhD Dissertation, Virginia Tech.* 2000 [Internet Acess] http://scholar.lib.vt.edu/theses/available/etd-06092000-12150028/unrestricted/etd.pd
[37] T. Kimoto, K. Asakawa, M. Yoda, M. Takeoka, “Stock market prediction system with modular neural networks”, *International Joint Conference on Neural Networks*, Pages 1-6. Piscataway, NJ, USA 1990.
[38] L. Mui, A. Agarwal, A. Gupta, P.W. Shen-Pei, “An Adaptive Modular Neural Network with Application to Unconstrained Character Recognition”, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, No. 5, Pages 1189-1204. October 1994.
[39] P. Blonda, V. Laforgiva, G. Pasquariello, G. Satalino, “Multispectral classification by modular neural network architecture”, *International Geoscience and Remote Sensing Technologies, Data Analysis and Interpretation*, Vol. 4. Pages 1873-1876. New York, 1993.
[40] Matsunaga, Andréa, A.B.F. José, “On the use of machine learning to predict the time and resources consumed by applications”, *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing. IEEE Computer Society*, 2010.
[41] L. Becchetti, S. Leonardi, S.A Marchetti, “Average-Case and Smoothed Competitive Analysis of the Multilevel Feedback Algorithm”, *Mathematics of Operation Research*, Vol. 31, 2006.