International Journal of
Intelligent Systems and
Applications in Engineering
ISSN:2147-6799 www.atscience.org/IJISAE
Advanced Technology and Science
Original Research Paper

Training Of Artificial Neural Network Using Metaheuristic Algorithm

Shaimaa Alwaisi*[1], Omer Kaan Baykan[1]

Accepted : 11/07/2017 Published: 31/07/2017

Abstract: This article clarify enhancing classification accuracy of Artificial Neural Network (ANN) by using metaheuristic optimization algorithm. Classification accuracy of ANN depends on the well-designed ANN model. Well-designed ANN model Based on the structure, activation function that are utilized for ANN nodes, and the training algorithm which are used to detect the correct weight for each node. In our paper we are focused on improving the set of synaptic weights by using Shuffled Frog Leaping metaheuristic optimization algorithm which are determine the correct weight for each node in ANN model. We used 10 well known datasets from UCI machine learning repository. In order to investigate the performance of ANN model we used datasets with different properties. These datasets have categorical, numerical and mixed properties. Then we compared the classification accuracy of proposed method with the classification accuracy of back propagation training algorithm. The results showed that the proposed algorithm performed better performance in the most used datasets.

Keywords: *Artificial Neural Network, Metaheuristic Optimization algorithm, Back propagation Algorithm, Shuffled Frog Leaping algorithm*

1. Introduction

The successes that are achieved by ANNs in various domains such as engineering technology and cognitive science cannot be ignored because of their worthy capabilities in cognitive complicated nonlinear mapping relationship and generating trainer models [1]. Gradient descent-based learning algorithm such as back-propagation are utilized in order to training ANNs. Unfortunately, Utilizing this earliest approach for training ANNs lead to get some drawbacks which include easy to fall into local minimum value, low precision and slow convergence speed [2]. To remedy these problems, the nature inspired algorithms which are represent metaheuristic algorithms are utilized to training ANNs. For instance, The researchers used developed second order stochastic learning algorithm in order to training a ANN model [3]; Krill Herd Algorithm has been used for training ANNs as in [4]. Other researchers used modified bat-inspired algorithm for optimizing ANNs as in [5]. They used modified versions of bat algorithm for enhancing both the weights and structure of ANNs. And these modification versions of bat algorithm led to enhancing the quality of convergence. Metaheurstic algorithms are classified into single-based and population-based methods. In training ANNs with single based metaheurstic, it is start with one solution and incorporate with its neighborhood to find the best solution, for instance [6] and simulated annealing [7]. Population based algorithms launched with many solutions and search for its neighborhood at the same time to create series of solution; they are stopping until condition is met. Population based algorithms also classify to evolutionary algorithm and swarm intelligence algorithm. Evolutionary algorithm EA involve genetic algorithm which are used in recent year for optimizing ANNs performance. The researchers adopted a new automatic method for enhancing ANNs structure. They adopted genetic approach, crossover method and enforcement of stochastic model. They used adopted hybrid model in order to choose best structure for ANN. Their proposed algorithm allowed to recombine feasible set of layers in the adopted model, feasible set of nodes in each layer, links between nodes and activation function for each node to generate feasible search space of possible architectures for the ANN model [8]. Other researchers proposed an approach that is combine the idea of Physics inspired gravitational search method and biology inspired flower pollination method. The experimental result for a proposed algorithm showed improving of classification accuracy [9]. Whereas swarm intelligence algorithm that are inspired from social behavior of animals which have a common task includes many studies. For example, the researchers used ant colony algorithm for training ANNs. They are adopted modified ant colony algorithm for learn the structure of feed-forward neural networks [10].the researchers used Multiverse model which is inspired from nature-animals algorithm to global optimization [11]. PSO algorithm that are proposed by Kennedy and Eberhard is one of the successful algorithms that have vital successes for training ANNs [12]. Modified PSO presented high performance for solving simple problem which did not produce good performance with other approach [13]. Other researchers adopted algorithm depend on Gaussian method and fuzzy concept to enhance the optimal weight for FFNN and improve FFNN architecture and layers. Utilizing of Gaussian distribution improved PSO convergence. As a consequence, the primary adjusts of the variables did not impact on the average and speed of PSO algorithm. After that, fuzzy reasoning rule utilized to eliminate redundancy in weights in Ann's architecture. They used iris dataset problem to analyze the achievement of adopted algorithm [14]. In addition to the updates that are happened on the original PSO algorithm, other PSO versions have also been modified to fix different real-life problems; in 2016 Sankhadeep Chatterjee et al. proposed (NN-PSO) classifier in order to expect the collapse potentially in RC

[1] *Selcuk University, Dept.of Comp. Eng., Konya,Turkey*
** Corresponding Author: Email: shaima.safa@yahoo.com*
Note: This paper has been presented at the 5th International Conference on Advanced Technology & Sciences (ICAT'17) held in Istanbul (Turkey), May 09-12, 2017

building body. They proposed particle swarm optimization-based approach in order to determine a set synaptic weights with less RMSE value in neural network model [15]. Tianhua Liu and Shoulin Yin proposed a developed inspired-nature PSO algorithm utilized to Back Propagation-ANN model. PSO utilized enhanced modified acceleration operator and developed modified inertia weight in order to optimize the weights and back propagation threshold for ANN model. These modification led to enhance ratio of convergence speed. In addition to enhancing a prediction accuracy for Back Propagation-ANN model. And then they are utilized multimedia evaluation approach in order to evaluate approach's efficiency [16].

In our work, Shuffled Leaping Frog Algorithm (SFLA) is utilized on ten datasets for training ANN. The datasets from UCI website. The performance of the SFLA is compared with the performance of the BP algorithm in WEKA data mining program.

2. Material and Methods

2.1. Artificial Neural Network

Artificial Neural Network (ANN) is a famous artificial intelligent model which is simulate the work of human mind mechanism. ANN is an artificial intelligent model that are handle information coming from different nodes in this model known as neurons. These nodes arranged in different layer which are work together to solve complicated task. An ANN model are shown Fig. 1.

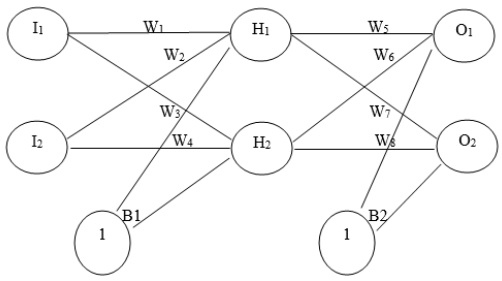


Figure 1. Artificial neural network model

The number of input layers to ANN represent by n, wij is the weights that are connect between input layer with hidden layer and j represent the number of hidden layer nodes. B is the bais for hidden layer. The relation between these parameters defined as follow:

$$Net\ (i) = \sum_{i=1}^{n} (Ii\ Wij) + Bj \qquad (1)$$

The following equation can be used for one or both hidden and output layer to find the output:

$$\text{Out (j)} = \frac{1}{1 \quad + \quad \exp(-net(i))} \qquad (2)$$

By using the above equals the output for output layers can be calculated. Error value for each output neuron can be calculates by the following equation:

$$\text{E (q)} \ = \ \frac{1}{2} \sum (T - O)^2 \qquad (3)$$

The main purpose for ANN is to learn the machine a particular task and update its knowledge about this task with respect to a given dataset. These properties made ANN optimal model for using in different fields such as pattern recognition and data classification problems. There are various structures for ANN each structure designed to solve a specific task. The basic architecture for ANN consist of three different layer. First layer is input layer which is consist of a set of input nodes that are represent input element for the ANN model. Second layer is hidden layer that is consist of hidden neurons. Hidden layer connected completely with output layer. Third layer is output layer which is represent the response layer for the model. Neurons in output layer represent activation functions that are calculate the final output for ANN. The nodes in ANN model connected together by synaptic weights. Firstly, these weights assigned randomly and updated during the learning phase by utilizing one of training algorithm.

For classification problem, ANN model require to be learned in order to be capable of expect the demand output value. Therefore, a set of sample dataset feed to ANN model in order to learn ANN [17].

2.2. Back Propagation Algorithm

Back Propagation algorithm is the most famous supervised learning algorithms that are used for training ANN models. The structure of back propagation algorithm consists of single input layer, one or more hidden layer and single output layer. These layers connected together by weights. The main purpose of back propagation algorithm is to decrease the error rate in the output leisurely through training phase. The back propagation algorithm achieves the back process in order to decrease the error rate. To compute the new value of weights for the output layer we use the following equation

$$\delta q \ = \ (Oq \ - \ Tq) * (1 \ - \ Oq) * Oq \qquad (4)$$

To compute the error value for the hidden layer we use the following equation:

$$\sigma_j \ = \ Hj \ * \ (\ 1 \ - \ Hj\ ) \ * \ \sum_{i=1}^{q} (\delta i\ Wi\ ) \qquad (5)$$

Where Hj represent the output of hidden layer.j represent the number of hidden layers.

In order to edit the weights for output layer using the following equation:

$$W_{jq} \ = \ W_{jq} \ + \ a * \ \delta_q \ * \ H_j \qquad (6)$$

In order to edit the weights for bais vector of output layer using the following equation:

$$W_q = \ W_q \ + \ a \ * \ \delta_q \qquad (7)$$

Where *a* represent adjustment parameter. In order to edit the weights for input layer using the following equation:

$$Wji \ = Wji \ + \ \eta \ * \sigma j \ * \ Ii \qquad (8)$$

Where η represent learning rate. In order to edit the weights for bais vector of input layer using the following equation:

$$W_i = \ W_i \ + \ \eta \ * \ \sigma j \qquad (9)$$

The activation function that we are used in our work is sigmoid function. Mean square error (MSE) can be used to find the rate of error between the calculated output and the target output [16][18].

2.3. Brief Introduction of SFLA

The shuffled frog leaping algorithm (SFLA) is a memetic meta heuristic optimization approach. SFLA algorithm developed by Eusuff and Lansey to solve complicated engineering problems[19]. For instance, vehicle routing problem [20] and channel equalization problem [21]. SFLA algorithm integrate the features of the GA and the nature-inspired PSO algorithm such

as few parameter, well efficiency in global search and swap global information through the individuals. SFLA compose of number of frogs grouped in population. Each frog in the population represent potential solution for the problem. Frog population divided into subgroups named memplexes according to a particular basis. Each frog have goal and preform local exploration and the worst's frog solution therefore they going on optimal solution. Each memplex develop with a specific number of steps. The frogs with their solutions in one memplex will be exchange with the frogs of other memplex through shuffling operation. The algorithm will repeat the operation of shuffling and sharing solution until the condition is met.

The steps of the algorithm as follow:

A. Create the population randomly

Generate the population F frog within a space solution

B. Organization and Sorting Operation

Compute the fitness function value for each frog in the population depending on the given optimization problem .then, detect the best and worst frog and sorting the frogs from the best to the worst.

C. Division Operation

Distribute the organized frog into m groups called memplex. Each memplex have n frogs so that F = m x n. The best frog go to the first memplx. The following one go to second memplex. The n^{th} frog go to the m^{th} memplex. While, $(m+1)^{th}$ go backward to the first group.

D. Evolution process

Detect the best and worst frog in each memplex as X_b and X_w respectively. Identify the best frog for the whole population as X_g. The worst frog will be modified as follow:

$$Si = rand\,(0,1) \text{ x } (Xb - Xw) \quad (10)$$

$$Xw = Xw + Si \;\; , -S\max \leq Si \leq S\max \quad (11)$$

Where S_{max} is the maximum possible movement in the frog location. i =1, 2,…,M_{gn} is the passing in the frog location. M_{gn} the number of iteration in each memplex. Submemplex is created to avoid falling in local optima, the frogs in supmemplex chosen according to their fitness value. The better frog have good chance to be chosen. The old frog`s fitness value is superseded when detect a better position. Otherwise X_b is changed by the global optimal solution.If there is no enhancement in the position then the position of the old frog will be created randomly and the fitness valu is computed. This operation is continue until the condition is met [22][23].

3. Proposed Method

Most important elements to design and improve the accuracy of the ANN is the collection of weights. These weights should be codified into the individual that represents the solution of our problem. The solutions generated by the bio inspired algorithm will be measured by the fitness function with the aim to select the best individual which represents the best ANN. SFLA is going to lead the evolutionary learning process until finding the best ANN. It is important to remark that only pattern classification problems will be solved by the proposed methodology. We supposed that the input for the ANN is equal for the number of attributes of the chosen problem, while the number of nodes in the hidden layer is chosen depending on repetitive examination and recording to detect the best hidden layer number for a chosen dataset. The number of output layer is equal to the number of classes for the chosen problem. We used sigmoid function for both hidden layers and output layers. SFLA algorithm led the global and local process for the ANN´s weights to detect the optimal weights set for the chosen problem.

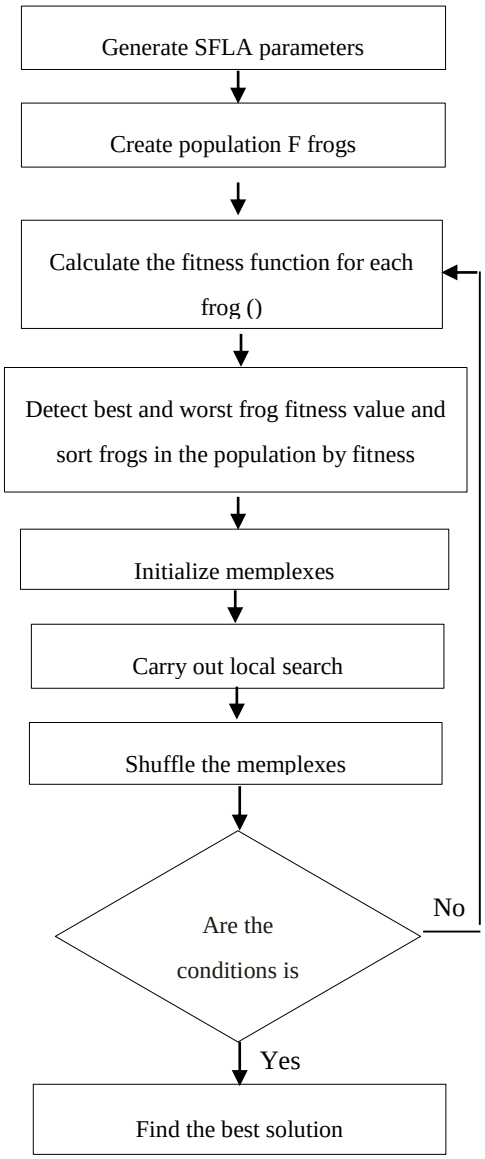


Figure. 2 Flowchart for Proposed Algorithm

4. Experimental Study

SFLA-ANN algorithm is appropriate for finding solution for the important multi dimension problems which require high classification accuracy. SFLA-ANN algorithm programmed by c# visual studio 2013. To demonstrate the efficiency of SFLA -ANN algorithm, we took ten well-known datasets from UCI website. Properties of the datasets are showed in Table 1. MLP-BP results were obtained using WEKA data mining program.

We used MSE equation in order to calculate the error rate. To demonstrate the efficiency of SFLA algorithm to learning ANN, we compared SFLA-ANN with BP algorithm in terms of accuracy ratio for classification. The parameters of BP

algorithm are learning rate equal to 0.3, momentum is equal to 0.2 and iteration number equal to 1000 .We chose datasets that are different in features, properties and number of samples .The results are shown in Table 3.

Table 1. Properties of Datasets

No	Name	Attributes	Samples	Class
1	Pima_Indians_Diabetes	8	768	2
2	Ecoli	8	336	8
3	İris	4	150	3
4	Thyroid	5	215	3
5	Wine	13	178	3
6	Haberman's Survival	3	306	2
7	Liver disorders	6	345	2
8	blood-transfusion	4	748	2
9	Heart	13	270	2
10	Balance	4	625	3

Table 2 SLFA parameters

Parameter	Value
Memplex number	m = 5
Number of frog in each memplex	n = 10
Total number of frog in the population	F = 50
The maximum allowed change in the frog position	S_{max} = 1.5

Table 3. Average of accuracy of classification

Dataset	BP algorithm accuracy%	SFLA algorithm accuracy%	Hidden layer
Pima Indians Diabetes	75	76,9	6
Haberman's Survival	68,6	78	4
Liver disorders	71,01	68,73	5
blood-transfusion	78,74	78,85	5
Iris	97,33	97,70	5
Wine	97,19	96,66	7
Thyroid	96,74	99,36	6
E.coli	85,11	82,89	6
Heart	78,51	83,58	7
Balance	90,88	91,48	5

We used preprocessing techniques to manipulate datasets before using the datasets in the algorithm. We made normalization for the datasets between [-1, 1] interval. We calculated the accuracy of classification ten time for each data set. At each time we made cross validation for the datasets. Fold number =10. Then, we calculated the average for the accuracy. of classification. When we analyzed the results we noticed for diabetes and Liver disorders dataset the accuracy of classification was between 76,9 % and 68,73%. This value is little comparing with the other datasets because of the limited number of samples in a particular class and bad preprocessing applied on dataset. In general, we noticed from the above table enhancing in training process for ANN. When we used ten different datasets the best testing results were for Thyroid dataset. This data set when it used to test with SLFA algorithm the accuracy of test was 99,36 % . The second good result were for iris data set with test accuracy reached to 97,7 %. In general, SFLA algorithm for training artificial neural network showed good performance in term of the accuracy of classification in comparing with BP algorithm.

5. Conclusion

In our study, ANN synaptic weights were detected by SFLA algorithm. We assigned the detected weights for the designed ANN model to implement classification process on ten datasets. In order to indicating the performance of the proposed algorithm we Chose datasets have different characteristics. Applying SFLA-ANN method on the datasets led to enhancing in the accuracy rate for the selected problem; hence, SFLA-ANN algorithm can be appropriate to rating crucial datasets that demand high classification accuracy, we can apply SFLA algorithm to learning complicated ANN models which consist of multi layers to enhance the accuracy rate.

References

[1] L. Wang and X. Fu, "Data mining with computational intelligence", 1th ed Berlin, Germany: Springer-Verlag, 2005.pp 276

[2] M. Alhamdoosh and D. Wang, "Fast decorrelated neural network ensembles with random weights", *INS Information Sciences*, vol. 264, pp. 104-117, 2014.

[3] S. S. Liew, M. Khalil-Hani, and R. Bakhteri, "An optimized second order stochastic learning algorithm for neural network training", *Neurocomputing*, vol. 186, no. 12, pp. 74-89, 2016.

[4] P. A. Kowalski and S. Lukasik, "Training Neural Networks with Krill Herd Algorithm", *Neural Process Lett*, vol. 44, no. 1, pp. 5-17, 2016.

[5] N. S. Jaddi, S. Abdullah, and A. R. Hamdan, "Optimization of neural network model using modified bat-inspired algorithm", *ASOC Applied Soft Computing*, vol. 37, pp. 71-86, 2015.

[6] S. Chalup, F. Maire, and C. E. C. "A study on hill climbing algorithms for neural network training ", (in No Linguistic Content), vol. 3, Washington D.C., 2014–2021, 1999, vol. 3 .

[7] R. S. Sexton, B. Alidaee, R. E. Dorsey, and J. D. Johnson, "Global optimization for artificial neural networks: A tabu search application", *European Journal of Operational Research*, vol. 106, no. 2, pp. 570-584, 1998.

[8] K. G. Kapanova, I. Dimov, and J. M. Sellier, "A genetic approach to automatic neural network architecture optimization,", *Neural Computing and Applications*, no. 4, 2016.

[9] D. Chakraborty, S. Saha, S. Maity, A., "Training feedforward neural networks using hybrid flower pollination-gravitational search algorithm", International Conference on Futuristic Trends on Computational, and M. Knowledge, pp. 261-266, 2015.

[10] K. M. Salama and A. M. Abdelbar, "Learning neural network structures with ant colony algorithms", *Swarm Intell Swarm Intelligence*, vol. 9, no. 4, pp. 229-265, 2015.

[11] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-Verse

Optimizer: a nature-inspired algorithm for global optimization,", *Neural Comput & Applic Neural Computing and Applications*, vol. 27, no. 2, pp. 495-513, 2016.

[12] J. Kennedy, R. Eberhart, and I. I. C. o. N. N. *"Particle swarm optimization "*, Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942-1948 vol.4, 1995.

[13] V. G. Gudise, G. K. Venayagamoorthy, and I. S. I. S., "Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks," Proceedings of the IEEE, pp. 110-117, 2003.

[14] H. Melo and J. Watada, "Gaussian-PSO with fuzzy reasoning based on structural learning for training a Neural Network", *NEUCOM Neurocomputing*, vol. 172, pp. 405-412, 2016.

[15] S. Chatterjee, S. Sarkar, S. Hore, N. Dey, A. S. Ashour, and V. E. Balas, "Particle swarm optimization trained neural network for structural failure prediction of multistoried RC buildings", *Neural Comput & Applic Neural Computing and Applications*, no. 9, 2016.

[16] T. Liu and S. Yin, "An improved particle swarm optimization algorithm used for BP neural network and multimedia course-ware evaluation", *Multimed Tools Appl Multimedia Tools and Applications*, no. 5, 2016.

[17] Sudip Mandal, G. Saha, and R. K. Pal, "Neural Network Training Using Firefly Algorithm," *Global Journal on Advancement in Engineering and Science (GJAES)*, vol. 1, no. 1, 2015.

[18] Hameed, A. A., Karlik, B. and Salman, M. S., "Back-propagation algorithm with variable adaptive momentum ", *Knowledge-Based Systems*, vol 114, pp 79-87,2016.

[19] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization", *Engineering Optimization*, vol. 38, no. 2, pp. 129-154, 2006.

[20] J. Luo, X. Li, M.-R. Chen, and H. Liu, "A novel hybrid shuffled frog leaping algorithm for vehicle routing problem with time windows", *Information Sciences Information Sciences*, vol. 316, no. 1, pp. 266-292, 2015.

[21] S. Panda, A. Sarangi, and S. P. Panigrahi, "A new training strategy for neural network using shuffled frog-leaping algorithm and application to channel equalization", *AEU - International Journal of Electronics and Communications AEU - International Journal of Electronics and Communications*, vol. 68, no. 11, pp. 1031-1036, 2014.

[22] K. K. Bhattacharjee and S. P. Sarmah, "Shuffled frog leaping algorithm and its application to 0/1 knapsack problem", *ASOC Applied Soft Computing Journal*, vol. 19, pp. 252-263, 2014.

[23] A. Abraham, J. L. Mauri, J. Buford, J. Suzuki, S. M. Thampi, " Advances in computing and communications " , Berlin, Germany: Springer-Verlag , 2011.