VOL. 11, NO. 24, DECEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

BAT-BP: A NEW BAT BASED BACK-PROPAGATION ALGORITHM FOR EFFICIENT DATA CLASSIFICATION

Nazri Mohd. Nawi[1], M. Z. Rehman[1], Nurfarian Hafifie[1], Abdullah Khan[1] and Insaf Ali Siming[2]
[1]Software and Multimedia Centre, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Johor, Malaysia
[2]Faculty of Science, Computer and Human Development, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Johor, Malaysia
E-Mail: nazri@uthm.edu.my

ABSTRACT

Training neural networks particularly back propagation algorithm is a complex task of great importance in the field of supervised learning. One of the nature inspired meta-heuristic Bat algorithm is becoming a popular method in solving many complex optimization problems. Thus, this study investigates the use of Bat algorithm along with back-propagation neural network (BPNN) algorithm in-order to gain optimal weights to solve the local minima problem and also to enhance the convergence rate. This study intends to show the superiority (time performance and quality of solution) of the proposed meta-heuristic Bat-BP algorithm over other more standard neural network training algorithms. The performance of the proposed Bat-BP algorithm is then compared with Artificial Bee Colony using BPNN (ABC-BP), Artificial Bee Colony using Levenberg-Marquardt (ABC-LM) and BPNN algorithm. Classification datasets from UCI machine learning repository are used to train the network. The simulation results show that the efficiency of BPNN training process is highly enhanced when combined with BAT algorithm.

Keywords: back propagation neural network, bat search, local minima, meta-heuristic optimization, hybrid neural networks.

INTRODUCTION

Artificial Neural Networks (ANN) are analytical techniques modelled on the neurological functions of the brain. It is also learn about processes of human cognitive system. ANNs consists of Artificial Neurons which is a small processing unit and are able to do works by processing information like biological neurons in the brain. The ANN is capable of machine learning, pattern recognition, optimization and others. Due to that, it can be trained to perform complex calculations [1].

ANN has the ability that it can be trained to store, recognize, estimate and adapt to new patterns without having the prior information of the function it receives has made ANN superior to the conventional methods used in the past. Due to its learning and adaptation ability, it has been widely used in the engineering fields such as biological modelling, decision modelling, control system, health and medicine etc [2-5]. In Artificial Neural Network (ANN), each node in a layer is connected to each other node in the adjacent layer. It is due to the behavior of ANN that consists of an input layer, hidden layers and also output layer of neurons. Meanwhile, Back-Propagation Neural Network (BPNN) algorithm known as the oldest supervised learning multilayer feed-forward ANN proposed by Rumelhart, Hinton, and Williams [6]. BPNN is highly suitable for problems where there are no relationships found between the output and inputs since its ability back-propagate. Thus, the flexibility and learning capabilities has made BPNN to be used in various range of applications [7]. Despite its high flexibility and learning capabilities, BPNN also has some limitations. Careful selection of parameters such as network topology, initial weights and biases, value of learning rate, activation function, and value for the gain in the activation function is required since it implement gradient descent learning rule. An improper choice of these parameters can lead to slow network convergence or network stagnancy. Regarding of these limitations problems, previous researchers have suggested some modifications to enhance and improve the training time of the network. Some of the variations suggested are the use of learning rate and also momentum to stop the network stagnancy and to speed up the network convergence. These two parameters are frequently used in order to control the adjustments of the weight along the steepest descent and control the oscillations [8-10]. In recent years, various new techniques have been proposed to train ANN and also to improve the weakness of gradient-based techniques. Some of the algorithms include global search techniques such as artificial bee colony back-propagation (ABC-BP) algorithm [11-12], genetic algorithm (GA) [13], hybrid PSO-BP [14], and evolutionary artificial neural networks algorithm (EA) [15]. However, these algorithms still not overcome the problem of local minima. Hence, in order to improve the local minima in BPNN convergence and also to gain precision, this paper proposes a new Bat-Based back-propagation (BAT-BP) algorithm which apply Bat algorithm [16] with BPNN in order to find the optimal weights in BPNN training process [6-10]. The proposed Bat-BP algorithm is used to train on datasets from the UCI machine learning repository.

The remaining paper is organized as follows: Section 2 explains Bat Search algorithm. Section 3 discusses the weight update process in the proposed Bat-BP algorithm. Simulation results are discussed in Section 4 and finally the paper is concluded in the Section 5.

THE PROPOSED BAT-BP ALGORITHM

Bat is a metaheuristic optimization algorithm developed by Xin-She Yang in 2010 [18]. Bat algorithm is based on the echolocation behavior of microbats with varying pulse rates of emission and loudness. In the

VOL. 11, NO. 24, DECEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

proposed BAT-BP algorithm [10], each position represents a possible solution (i.e., the weight space and the corresponding biases for BPNN optimization in this paper). The weight optimization problem and the position of a food source represent the quality of the solution. In the first epoch, the best weights and biases are initialized with BAT and then those weights are passed on to the BPNN where momentum coefficient, α is appended. The weights in BPNN are calculated and compared in the reverse cycle. The BAT will continue searching the best weights until the last cycle/ epoch of the network is reached or either the MSE is achieved. The pseudo code for the Bat-BP is given as;

1. Initialize Bat population size and BPNN Structure
2. Load the training data
3. While MSE<stopping criteria
4. Pass the cuckoo nests as weights to network
5. Feed forward network runs using the weights initialized with Bat
6. The sensitivity of one layer is calculated from its previous one

   and the calculation of the sensitivity starts from the last layer of the network and move backwards.

7. Update weights and bias in the network
8. Calculate the error in the network
9. Minimize the error by adjusting network parameters using Bat
10. Generate Bat Preys (x_j) by selecting random target preys $i.e. X_i = X_j$
11. Evaluate the fitness of the prey, Chose a random prey i

If

a. $X_j > X_i$ *Then*

b. $x_i \leftarrow x_j$

c. $X_i \leftarrow V_j$

End if

12. Bat keeps on calculating the best possible weight at each epoch until the network is converged.
13. End While

RESULTS AND DISCUSSIONS

The workstation used for experimentation was equipped with a 2.5GHz Core-i5 processor and 4-GB of RAM. The simulations are carried-out using MATLAB 2010 software on three classification datasets such as breast cancer [17], diabetes [18] and glass [19]. The following algorithms are analyzed and simulated on the datasets;

1. Back Propagation Neural Network (BPNN) algorithm
2. Artificial Bee Colony with Back-Propagation (ABC-BP) algorithm
3. Artificial Bee Colony with Levenberg Marquardt (ABC-LM) algorithm and
4. The proposed Bat based Back-Propagation (BAT-BP) algorithm

Breast cancer classification dataset

This problem tries to diagnosis breast cancer by trying to classify a tumor as either benign or malignant based from continuous clinical trials [17]. The selected network architecture is used for the breast cancer classification problem is consists of 9 inputs nodes, 5 hidden nodes and 2 output nodes.

Table-1. CPU time, Epochs, MSE, accuracy for breast cancer dataset.

Algorithm	BPNN	ABC-BP	ABC-LM	Bat-BP
CPU Time	95.46	1482.9	1880.64	345.42
Epochs	1000	1000	1000	1000
MSE	0.271	0.184	0.0139	0.0219
Accuracy (%)	90.71	92.02	93.831	97.81

In the Table-1, it is clearly seen that proposed BAT-BP algorithm converged within 100 epoch and 345.42 CPU cycles. Meanwhile, ABC-BP is seen converging within 1482.9 CPU cycles and in 150 epochs. Whereas, ABC-LM converges within 1880.64 CPU cycles and 1000 epochs. The proposed BAT-BP has the higher accuracy than all algorithms except ABC-LM which is giving slightly better mean squared error (MSE) than Bat-BP.

Pima Indian diabetes dataset

Taken from UCI Machine learning Repository, this dataset consists of 768 examples, 8 inputs and 2 outputs and contains all the information of the chemical change in a female body whose disparity can cause diabetes [18]. The feed forward network topology for this network is set to 8-5-2.

Table-2. CPU time, Epochs, MSE, accuracy for diabetes dataset.

Algorithm	BPNN	ABC-BP	ABC-LM	Bat-BP
CPU Time	57.05	4257	2805.09	532.17
Epochs	1000	1000	1000	1000
MSE	0.269	0.201	0.141	0.0093
Accuracy (%)	84.94	91.468	65.098	99.07

For the Bat-BP, ABC-LM, ABC-BP and BPNN, Table-2 shows the CPU time, epochs and the MSE for the Diabetes Classification dataset test problem. As can be seen clearly from the Table-2, the proposed Bat-BP has the highest accuracy and converges within 1000 epoch. Bat-BP converged to global minima with an MSE of 0.0093 and within 532.17 CPU cycles showing the best performance among the other three algorithms.

Glass classification dataset

The Glass dataset is also taken from UCI repository. This dataset has 214 instances [19]. The selected feed forward network architecture for this network is set to 9 inputs, 5 hidden and 6 outputs layers.

Table-3. CPU time, Epochs, MSE, accuracy for glass dataset.

Algorithm	BPNN	ABC-BP	ABC-LM	Bat-BP
CPU Time	32.73	1715.9	1336.19	268.60
Epochs	1000	1000	1000	1000
MSE	0.364	0.0258	0.005	0.0048
Accuracy (%)	94.04	91.365	93.96	99.52

It can also be noted from the Table-3 that ABC-BP has the least accuracy and also take 1715.9 CPU cycles to complete in the 1000 epochs. While, ABC-LM took the highest CPU time which is 1336.19 in order to complete in 1000 epochs with an average accuracy of 93.96 and 0.005 MSE. BPNN shows the fastest CPU cycle that is 32.73 but a bigger MSE, i.e. 0.364. In Table-3, we can see that Bat-BP is converging with a smaller 0.0048 MSE and 50 epochs along with a mere 268.60 CPU cycles.

CONCLUSIONS

As a popular and most widely used algorithm, BPNN is known to be able to train Artificial Neural Networks (ANN) successfully. However, BPNN algorithms have some limitations; such as getting stuck in local minima and slow convergence rate. In this paper, Bat-BP algorithm is proposed to train BPNN in order to achieve fast convergence rate and enhanced accuracy. The performance of Bat-BP algorithm is then compared with the ABC-CP, ABC-LM, and also BPNN algorithms. The performance of the proposed Bat-BP is verified by means of simulations on Breast Cancer (Wisconsin) Classification, Diabetes, and Glass datasets are used respectively. The simulation results show that the proposed Bat-BP shows high accuracy for all dataset.

ACKNOWLEDGEMENTS

The Authors would like to thank Office of Research, Innovation, Commercialization and Consultancy (ORICC), Universiti Tun Hussein Onn Malaysia (UTHM) and Ministry of Higher Education (MOHE) Malaysia for financially supporting this Research under Fundamental Research Grant Scheme (FRGS) vote no. 1236. This research is also supported by Gates IT Solution Sdn. Bhd under its publication scheme.

REFERENCES

[1] Deng W. J., Chen W. C., and PeiW. 2008. Back-propagation neural network based importance-performance for determining critical service attributes, Journal of Expert Systems and Applications. 2. pp. 1-26.

[2] Zheng H., Meng W. and Gong B. 1992. Neural Network and its Application on Machine fault Diagnosis. ICSYSE 1992. pp. 576-579.

[3] KoskoB. 1992. Neural Network and Fuzzy Systems, 1st Edition, Prentice Hall, India.

[4] Basheer I. A. and Hajmeer M. 2000. Artificial Neural Networks: fundamentals, computing, design and application, Journal of Microbiological Methods. 43 (1). pp.03-31.

[5] Krasnopolsky V. M., and Chevallier F. 2003. Some Neural Network applications in environmental sciences. Part II: advancing computational efficiency of environmental numerical models, Journal of Neural Networks. 16 (3). pp.335-348.

[6] Rumelhart D. E., HintonG. E. and Williams R. J. 1986. Learning Internal Representations by error Propagation. Parallel Distributed Processing: Explorations in the Microstructure of Cognition.

[7] Lee K., Booth D. and Alam P. A. 2005. Comparison of Supervised and Unsupervised Neural Networks in Predicting Bankruptcy of Korean Firms. Expert Systems with Applications. 29.

[8] RehmanM. Z. and Nawi N. M. 2011. The Effect of Adaptive Momentum in Improving the Accuracy of Gradient Descent Back Propagation Algorithm on Classification Problems. CCIS Journal of Software Engineering and Computer Systems, Springer Heidelberg. 179 (6). pp.380-390.

[9] Rehman M. Z.,Nawi N. M. 2012. Studying the effect of adaptive momentum in improving the accuracy of gradient descent back propagation algorithm on classification problems, International Journal of Modern Physics (IJMPCS). 1 (1).

[10] Nawi N. M., Rehman M. Z. and Khan A. 2014. A New Bat based Back-Propagation (Bat-BP) algorithm. Advances in Systems Science: Advances in Intelligent

systems and Computing. Springer International Publishing.

[11] Nandy S., Sarkar P. P. and DasA. 2012. Training a Feed-forward Neural Network with Artificial Bee Colony Based Back propagation Method. International Journal of Computer Science and Information Technology (IJCSIT). 4 (4). pp.33-46.

[12] KarabogaD., AkayB., and OzturkC. 2007. Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks. 4th International Conference on Modeling Decisions for Artificial Intelligence (MDAI 2007). Kitakyushu, Japan. August 16-18.

[13] MontanaD. J. and DavisL. 1989. Training feed forward neural networks using genetic algorithms. Proceedings of the 11thInternational Joint Conference on Artificial Intelligence. 1. pp.762-767.

[14] Mendes R., Cortez P., Rocha M., Neves J. 2002. Particle swarm for feed forward neural network training. Proceedings of the International Joint Conference on Neural Networks. 2. pp.1895-1899.

[15] YaoX. 1993. Evolutionary artificial neural networks. International Journal of Neural Systems. 4(3). pp.203-222.

[16] Yang X. S. 2010. A new metaheuristic bat-inspired algorithm. Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). pp. 65-74.

[17] Wolberg W. H. and Mangasarian O. L. 1990. Multisurface method of pattern separation for medical diagnosis applied to breast cytology, National Academy of Sciences. 87. pp. 9193-9196.

[18] Smith J. W., Everhart J. E., Dickson W. C., Knowler W. C. and Johannes R. S. 1988. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. Symposium on Computer Applications and Medical Care. pp.261-265.

[19] Evett I. W. and Spiehler E. J. 1987. Rule induction in forensic science. KBS in Government, Online Publications. pp. 107-118.