

Dynamic Hidden Layers Selection of ANN Architecture Using Particle Swarm Optimization

Qamar Abbas, Jamil Ahmad, and Waqas Haider Bangyal

Abstract —The main purpose of this paper is that how to make Artificial Neural Networks (ANN) dynamic in the sense that it can decide that which architecture from given set of architecture has the optimal results. For this we need to use some optimization technique to get the optimum architecture of the ANN. In this paper Particle Swarm Optimization Technique is used. Particle Swarm Optimization (PSO) has applied to variety of optimization problems and it provides good results. In this research different architecture optimization techniques for ANN applications are reviewed. Researchers have proposed different techniques of ANN optimizations but still there required efficient ANN architecture optimization technique. In this paper the survey of ANN optimization techniques and PSO is presented and then the solution to make the ANN dynamic using PSO is proposed.

Index Terms — PSO, particle, layer.

I. INTRODUCTION

An Artificial Neural Network (ANN) is a branch of Artificial Intelligence. ANN is applied to a variety of engineering, medical, and scientific applications successfully. The main part of ANN is the architecture of the designed ANN for any application. So for it is a trial and error based. Trial and error based is a good technique but it takes lot of time to find the optimum architecture of the ANN [1].

Lot of peoples have worked on many techniques of ANN architecture optimization [2], [3], [4]. But the techniques used for optimization are not dynamic. The optimization techniques on which people have worked are discussed in detail in this paper.

PSO is motivated by social behavior of bird flocking or fish schooling. Particle Swarm Optimization (PSO) is a population based stochastic optimization technique solving continuous and discrete optimization problems developed by Dr. Kennedy and Dr. Eberhart in 1995[5].

The solution sets that are used in PSO are known as a particle that moves in the solution space and finds the particle with optimum solution [6]. The system is initialized with a population of random solutions. Each particle keep track of the best solution found so far called fitness. Two types of fitness values are store one for the particle normally known as p_{best} and one the best fitness of the whole system known as g_{best} . Each particle has some position and velocity and position and velocity are updated towards the particle best position and best velocity [2], [5], [6]. The pseudo code of PSO algorithm is as given.

The steps of PSO algorithms [19] are as given follows

- In the first step velocities and positions of the particles of the population are initialized.
- In second step fitness of each particle is calculated.
- In third step local best position is changes if the current value of fitness is larger than the local best fitness.
- In fourth step change the global best position if the current value of the fitness is greater than the global best fitness.
- In fifth step the position and the velocities are updated according to equations (1, 2).
- In the sixth step we repeat the steps second to fifth until we reach the stopping criteria.

The rest of the paper is structured as follows: in Section II, literature survey is discussed. Section III illustrates proposed technique. Section IV presents the conclusion of the paper and Section V presents the future work.

The equation to update the velocity is as given follows

$$V = V_{old} + c_1 * r_1(p_{best} - current) + c_2 * r_2(g_{best} - current) \quad (1)$$

where V is velocity, V_{old} is previous velocity, c_1, c_2 are social factors and r_1, r_2 are random numbers, p_{best} is the current particle best position, $current$ is the current particle position, g_{best} is the best of best particle position.

The equation to update the position of the particle is as given follows

$$X = X_{old} + V \quad (2)$$

where X is the new position of the particle, X_{old} is the previous position of the considered particle and V is the velocity calculated using equation (1).

II. LITERATURE SURVEY

Optimization of ANN architecture is very important. It involves many of the unique features that are quite different from the common practices of ANN. The models of the applications of the ANN are expensive to create and applicability is limited as well. For these reasons architecture of ANN should be optimized. But in past few years, efforts have been made for some applications. These different optimization techniques for ANN architecture optimization are given in this section in detail.

Carvalho [8] has used PSO with two variants is discussed with two variants for the training of neural networks. The medical field bench mark classification problem is used. Local search operator is used with PSO to find the simulation

Manuscript received November 23, 2012; revised February 20, 2013.

The authors are with Iqra University, Islamabad, Pakistan (email: qamar.bhk@gmail.com, jamil.ahmad@abasyn.edu.pk, waqas_bangyal@hotmail.com).

results. Simulations results show that the hybrid PSO proves itself to be better for neural network training.

Zhong [9] proposed the PSO for the radial base function neural networks. The weights and the structure of the RBF neural networks are optimized. The simulations results using PSO base neural networks on the gearbox fault diagnosis application shows that it has better classification effect as compared to RBF based neural networks.

Park [7] used new selection strategy of the adaptive inertial weight and weight optimization of ANN is proposed. This technique has based on the training error of ANN. the simulation results shows that using this strategy the results are more accurate and speedy.

Su [10] has trained ANN ridgelet neural network with improved particle swarm optimization. Improved PSO with mutation operator is applied to ridgelet neural network and the simulation results shows that Improved PSO with mutation operator has better performance as compared to the traditional PSO.

Guang [2] has implemented the particle swarm optimization based Backpropagation (PSOBP) neural network on the engineering projects to improve their risk evaluation quality. The simulation results obtained through PSOBP are better as compared to the Backpropagation neural network applied on risk evaluation engineering project.

Ling [11] has used PSO based on fuzzy logic with cross mutated operation (FPSOCM). Inertia weight is determined by fuzzy inference system using FPSOCM. For tuning parameters of ANN, local optimum solution is easily found by using variable inertial weight and the cross-mutated operation.

Zhitao [12] has trained neural network PSO with adaptive inertia weight is presented. The inertia weight is improved on the adaptive base for optimum weights and threshold. Improved PSO is used to extract the features of classification from the Chinese handwritten characters. The simulation results of the improved PSO are compared with the traditional Backpropagation algorithm of ANN which clearly shows that performance of improved PSO is better than the Backpropagation.

Tang [13] identified the problem of local optima in traditional Backpropagation algorithm of ANN and PSO algorithm is suggested. In this paper the algorithm that is used is guaranteed convergence particle swarm optimizer (GCPSO) algorithm Backpropagation. The simulation results of GCPSO are compared with the results of Backpropagation neural network for pattern recognition application which shows that GCPSO has better results.

Wu [14] proposed and implemented fuzzy neural network with PSO (PSO-FNN). This algorithm can take the advantage of the PSO algorithm. The simulation results of PSO-FNN are précised and better than Backpropagation neural network.

Zhang [15] proposed BackPropagation Neural Network (BPNN) based on PSO and robust error function for image restoration application. Different image restoration parameters are tested during the simulation results. These results provide good quality in terms of visual quantities and qualitative performance.

Zhang [16] has implemented fuzzy neural network based on the PSO. Some problems of BP neural networks like long

training time, fall into local minimum and initial conditions are considered for fuzzy neural networks based on PSO and the simulation results shows that results of fuzzy based system are effective and feasible.

Zhao [6] has proposed and implemented RBF neural network using PSO with mutation operator. All the concerned parameters are coded with the proposed algorithm and the simulation results are compared with the standard PSO algorithm. Simulation results of the proposed algorithm are more effective.

Nuzly [17] has proposed and implemented Quantum-inspired Particle Swarm Optimization (QiPSO) based Evolving Spiking Neural Network (ESNN). To test the performance of this proposed algorithm a synthetic data set application is used. The simulation results of this proposed methods are promising.

In [3] PSO based neural network is proposed for the power forecast application. This algorithm is culture PSO that overcomes the problems of the normal neural networks. To achieve the global optimization the population space knowledge is directly transferred to knowledge space. The simulation results show that neural network based culture PSO has better results.

Ma [17] has used new evaluation method PSO and BP based. Evaluation index system application is used. This system can help management departments and government supervision to select the best route. The simulation results show that this new method is feasible and has scientific results.

Kiranyaz [4] proposed and implemented multidimensional Particle Swarm Optimization (MDPSO). This method used multidimensional search space in the PSO technique for optimum dimension. Dimension optimum and error apace can be found using this method in the architecture space. The simulation results of MD PSO are efficient.

III. PROPOSED TECHNIQUE

The proposed technique is described here. PSO will have some dimensions. These dimensions will have the same number as the output nodes only. Each output node will be represented by one dimension of PSO. And each particle will represent one node in the hidden layer. Then the technique of the PSO as specified above will be applied on the solution space. Solution space has some dynamic solutions of the ANN architecture. The PSO will dynamically predict the best ANN architecture for the any application. It is important to note that the number of hidden neurons in the hidden layer shall not be specified by the user. The PSO will predict the optimum and the suitable architecture of the ANN. The winner particle will mean that this particle represent the optimum solution of the considered ANN application. The winner particle of the PSO will be the winner output node.

IV. CONCLUSION

Many researchers have applied PSO to ANN applications as an alternate of Backpropagation algorithm. But no one have so for used PSO on hidden layer for the dynamic ANN

architecture selection from the solution space of the ANN architectures. Since it is very common to use trial and error method to select the ANN architecture. It is a time consuming and difficult task. To remove this deficiency, dynamic ANN architecture selection using PSO is proposed. It is suggested that this proposed technique shall proved to be better when this proposed technique will be implemented, it will prove to be better solution for the dynamic ANN architecture selection for all applications.

V. FUTURE WORK

The future work of this paper will be that this proposed technique will be applied on some application like digit recognition, character recognition, symbol recognition and vice versa. The results will then be compared with manual designed architectures to check the accuracy of results.

REFERENCES

- [1] M. Carvalho and T. B. Ludermir, "An analysis of PSO hybrid algorithms for feed-forward neural networks training," *Ninth Brazilian Symposium on Neural Networks*, pp. 6-11, 2006.
- [2] B. Zhong, D. Wang, and T. Li, "Application of optimized neural network based on particle swarm optimization algorithm in fault diagnosis," in *Proc. IEEE International Conference on Cognitive Informatics*, pp. 476-480, 2009.
- [3] T. Park, J. Lee, and B. Choi, "Optimization for artificial neural network with adaptive inertial weight of particle swarm optimization," in *Proc. IEEE International Conference on Cognitive Informatics*, pp. 481 – 485, 2009.
- [4] R. Su, *et al.* "A New ridgelet neural network training algorithm based on improved particle swarm optimization," in *Proc. Third International IEEE Conference on Natural Computation*, pp. 411-415, 2007.
- [5] C. Chang, *et al.* "Application of Particle swarm optimization based BP neural network on engineering project risk evaluating," in *Proc. Third International Conference on Natural Computation*, pp. 750 – 754, 2007.
- [6] S. H. Ling, H. T. Nguyen, and K. Y. Chan, "A new particle swarm optimization algorithm for neural network optimization," in *Proc. Third International Conference on Network and System Security*, pp. 516 – 521, 2009.
- [7] G. Zhitao, *et al.* "Handwritten Chinese characters recognition based on PSO neural networks," in *Proc. Second International IEEE Conference on Intelligent Networks and Intelligent Systems*, pp. 350-353, 2009.
- [8] P. Tang and Z. Xi, "The research on BP neural network model based on guaranteed convergence particle swarm optimization," in *Proc. Second International Symposium on Intelligent Information Technology Application*, pp. 13-16, 2008.
- [9] X. Wu, M. Yuan, and H. Yu, "Soft-sensor modeling of cement raw material blending process based on fuzzy neural networks with particle swarm optimization," in *Proc. International Conference on Computational Intelligence and Natural Computing*, pp. 158 – 161, 2009.
- [10] Y. Zhang, *et al.* "Image restoration based on robust error function and particle swarm optimization bp neural network," in *Proc. Fourth International Conference on Natural Computation*, pp. 640-644, 2008.
- [11] X. Zhang, *et al.* "A fuzzy neural network based on particle swarm optimization applied in the speech recognition system," in *Proc. Eighth International Conference on Intelligent Systems Design and Applications*, pp. 693-697, 2008.
- [12] J. Zhao, W. Chen, and W. Xu, "Nonlinear system identification based on radial basis function neural network using improved particle swarm optimization," in *Proc. Fifth International Conference on Natural Computation*, pp. 409 - 413, 2009.
- [13] H. N. A. Hamed, N. Kasabov, and S. M. Shamsuddin, "Integrated feature selection and parameter optimization for evolving spiking neural networks using quantum inspired particle swarm optimization," in *Proc. International Conference of Soft Computing and Pattern Recognition*, pp. 611-619, 2009.
- [14] D. Niu, Z. Gu, and M. Xing, "Research on neural networks based on culture particle swarm optimization and its application in power load forecasting," in *Proc. Third International Conference on Natural Computation*, pp. 270 – 274, 2007.
- [15] C. Ma, F. Wu, and R. Lu, "Decision-making method of hazardous material transportation route based on particle swarm optimization algorithm and neural network," *Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, pp. 1023 - 1027, 2008.
- [16] S. Kiranyaz, *et al.* "Unsupervised design of artificial neural networks via multi-dimensional particle swarm optimization," in *Proc. International Conference on Pattern Recognition*, pp. 1 – 4, 2008.
- [17] L. Fausset, *Fundamental of Neural Networks*, Third Edition, 2008.
- [18] M. A. Otair and W. A. Salameh, "Speeding up back-propagation neural networks," in *Proc. Proceedings of the 2005 Informing Science and IT Education Joint Conference*, pp. 167-173, 2005.
- [19] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization," in *Proc. IEEE International Conference on Neural Networks*, pp. 1942 – 1948, 1995.