Journal of Computers Vol. 29 No. 4, 2018, pp. 269-276
doi:10.3966/199115992018082904022

Classification of Speech Based on BP Neural Network Optimized by PSO

Hong-wei Ye[1], Xiao-jun Wen[2*]

[1] School of Electronic and Information Engineering, Heyuan Polytechnic, Heyuan, 517000, China
aboyhw@163.com

[2] School of Computer Engineering, Shenzhen Polytechnic, Shenzhen, 518055, China
szwxjun@sina.com

Received 1 December 2017; Revised 30 April 2018; Accepted 31 May 2018

Abstract. A back-propagation (BP) neural network consists of an input layer, one or more hidden layers and an output layer. An input vector is presented to the network, it is propagated forward through the network, layer by layer, until it reaches the output layer. The output of the network is then compared to the desired output, using a loss function, The BP neural network easily falls into a local extreme values and the slow convergence, during the Classification of Speech using it. This experiment selected four types of speech: Guzheng, folk songs, rock and pop, the categories of speech were converted to the matrix $Z_{1\times4}$. A BP neural network structure of 24-7-4 was established. The BP neural network with 24 dimensional feature data as input and four categories as output was used. A new method is put forward to optimize weights and threshold of BP neural network using PSO. results of which were analysed and compared with that BP neural network. The PSO-BP neural network not only enhances the accuracy, but also reduces the computation.

Keywords: BP neural network, Classification, optimization, PSO

1 Introduction

The optimization of systems and processes is very important to the efficiency and engineering domains. Optimization problems are solved by using rigorous or approximate mathematical search techniques. The BP Neural Network is the very important method in the domains. Unfortunately, the BP neural network easily falls into a local extreme values and the slow convergence. In order to improve the shortcomings of BP neural networks, many methods for optimizing BP neural networks are proposed. Taking advantage of genetic algorithm to optimize the weights and threshold of BP neural network [1]. A model is based on Glowworm Swarm Optimization optimized BP network [2], An optimized BP neural network was proposed based on modified artificial fish algorithm, it is proved that the proposed optimization method improves the generalization performance of BP neural network according to function fitting simulation experiments [3]. Repeated the optimization of the BP network's weight combination with the aid of a clonal selection particle swarm algorithm, and then adopted the weight optimized as the initial value of the BP neural network [4]. BP neural network improved by genetic algorithm (GABP) is established to model the relation between welding appearance and the characteristics of the molten-pool-shadows. [5] The above mentioned optimization method is more complex to implement, Particle swarm is based on the algorithm described in Kennedy and Eberhart [6]. Particle swarm optimization (PSO) is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space [7]. The advantages of PSO are that it is easy to implement and there are few parameters to adjust [8-10]. We use the advantages of Particle swarm optimization to

* Corresponding Author

optimize the weights and threshold of BP neural network. And then Use the obtained optimal value to train the network and apply it to speech feature signal classification.

2 Research Method

2.1 BP Neural Network

The architecture of a BP network as shown in Fig. 1, consists of an input layer, one or more hidden layers and an output layer. There are i input nodes, j hidden nodes and k output nodes. All input nodes are connected to all hidden nodes through weighted connections w_{ji}, and all hidden nodes are connected to all output node through weighted connection w_{kj}, During supervised training, input patterns supplied to the BP network are processed in two stages. In the first stage, the training patterns are passed forward through the network architecture to provide a predicted value for each output variable of the BP model. Any error associated with the prediction is then passed back through the model to update the hidden and output layer weights, with the objective of reducing the associated prediction error [11-15]. This network training procedure continues until the network weights are updated sufficiently so that further error reduction is not possible.

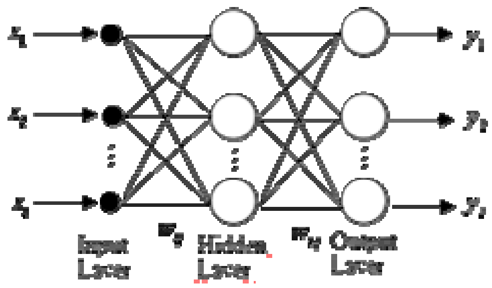

Fig. 1. Basic structure of a BP neural network

The specific steps of BP neural network training as follows:

Step 1. Initialize, In order to have some numbers to work with, here are the initial n input nodes, p hidden nodes and q output nodes. All input nodes are connected to all hidden nodes through weighted connections w_{ij}, and all hidden nodes are connected to all output node through weighted connection w_{kj}. The thresholds of the hidden and output layers are h and s . Determining the activation function and learning rate of a neuron's activation function.

Step 2. Calculate the output of the hidden nodes, where H_j is the output of the hidden layer. P is the number of the hidden nodes, Where f *is* the activation function, The input data is represented as a vector X=(x_1, x_2, …… x_i). h_j *is* the threshold of the hidden layer.

$$H_j = f\left(\sum_{i=1}^{n} w_{\mathrm{ij}} x_i - h_j\right) \qquad j = 1, 2, \cdots, p \tag{1}$$

Step 3. Calculate the output of the output nodes, where s_k *is* the threshold of the output layer.

$$O_{\mathrm{k}} = \sum_{j}^{p} H_j \, w_{\mathrm{jk}} - s_k \qquad k = 1, 2, \cdots, q \tag{2}$$

Step 4. Calculate the error of the output nodes. where T_k is the real result.

$$\mathrm{e}_{\mathrm{k}} = T_K - O_k \qquad k = 1, 2, \cdots, q \tag{3}$$

Step 5. Weight updates

$$w_{ij} = w_{ij} + \eta H_j (1 - H_j) x(i) \sum_{k=1}^{p} w_{jk} e_k \qquad i = 1, 2, \cdots, n \qquad j = 1, 2, \cdots p \tag{4}$$

$$w_{jk} = w_{jk} + \eta H_j e_k \quad j = 1,2,\cdots p \quad k = 1,2,\cdots q \tag{5}$$

Step 6. Threshold updates

$$h_j = h_j + \eta H_j (1 - H_j) x(i) \sum_{k=1}^{p} w_{jk} e_k \quad j = 1,2,\cdots p \tag{6}$$

$$s_k = s_k + e_k \qquad k = 1,2,\cdots q \tag{7}$$

Step 7. If the convergence criteria are satisfied, stop. Otherwise, return to step 2.

2.2 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a technique used to explore the search space of a domain problem to find the settings or parameters required to maximize or minimize a particular objective. The particle swarm algorithm begins by creating the initial particles, and assigning them initial velocities. It evaluates the objective function at each particle location, and determines the best (lowest) function value and the best location. It chooses new velocities, based on the current velocity, the particles' individual best locations, and the best locations of their neighbors. It then iteratively updates the particle locations, velocities, and neighbors. Iterations proceed until the algorithm reaches a stopping criterion. Supposing in a S dimension search space, a swarm formed by M={M_1, M_2, …… M_i} particles through a randomly initializes, then the position M_i={m_{i1}, m_{i2}, …… m_{iS}} is the ith particle in a swarm. Substitute them into the object function and adaptive value will be come out, which can be used to evaluate the solution. the velocity can be represented as V_i={v_{i1}, v_{i2}, …… v_{iS}}, individual extremum P_i={p_{i1}, p_{i2}, …… p_{iS}} represents the ith particles' individual best position, the global extremum P_z={p_{z1}, p_{z2}, …… p_{zS}} represents the best locations of their neighbors. Velocity and position are updated each time according to formulas below.

$$\begin{cases} v_{id}(t+1) = \omega v_{id}(t) + \varphi_1\ rand\ (\,)\,[p_{id} - m_{id}(t)] + \varphi_2\ rand\ ()[p_{zd} - m_{id}(t)] \\ z_{id}(t+1) = z_{id}(t) + v_{id}(t+1) \\ if\ v_{id}(t+1) > v_{max}, v_{id}(t+1) = v_{max} \\ if\ v_{id}(t+1) < -v_{max}, v_{id}(t+1) = -v_{max} \end{cases} \tag{8}$$

In the formula, t represents iteration number, ω is inertia weight, φ_1, φ_2 is learning factor, $rand\ (\,)$ is a random numbers in the range [0,1].

2.3 Dataset and Preprocessing

The speech signal classification process is:After preprocess the acquired speech signal, a suitable algorithm is used to extract the speech feature signal from the speech signal. The speech feature signal can be seen as a pattern of speech, and then through the comparison of existing reference patterns. Get the best matching reference pattern to get the classification of the speech signal. This experiment selected four types of speech: Guzheng, folk songs, rock and pop, Extracting 500 groups of 24 dimensional speech feature signals using Cepstrum coefficient. The speech feature signals of the four types of music as shown Fig. 2. The Part of the speech feature as shown Table 1.

Before the training of BP network, In order to reduce the impact of data on BP network prediction errors, Map matrix row minimum and maximum values to [-1,1]. The features in each dimension is normalized by

$$X_n = \frac{x_n - x_{\min}}{x_{\max} - x_{\min}} \tag{9}$$

Where x_n is the nth values of the speech feature, $x_{\min}$ is the minimum values of the speech feature and $x_{\max}$ is the maximum values. In order to use BP neural network to identify four types of speech, the categories of speech were converted to the matrix $Z_{1\times4}$ as shown in Table 2. For example, the BP neural network output is [0.0442 0.9549 -0.0461 0.0470], this type of speech belongs to category 2.

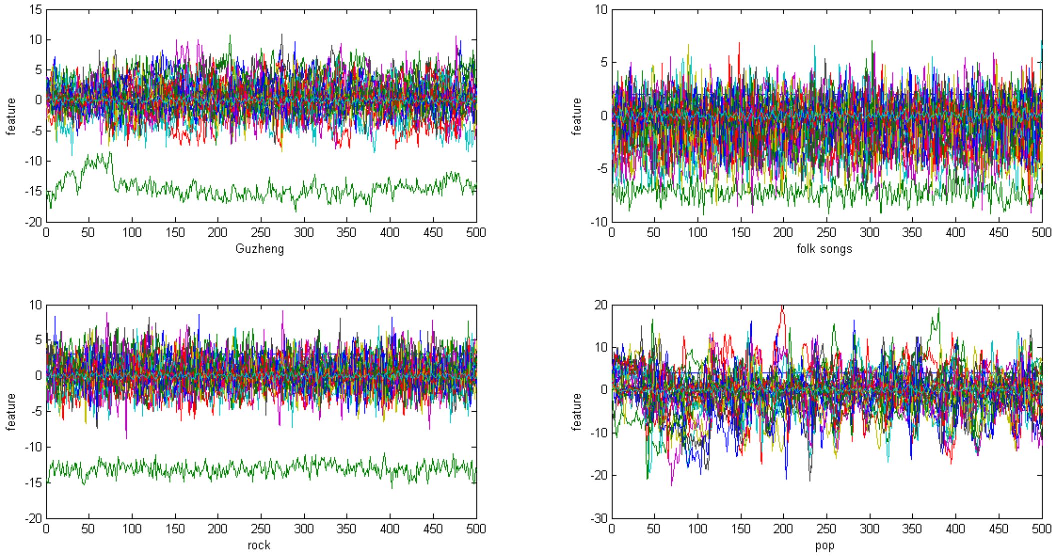

Fig.2. The speech feature signals of the four types of music

Table 1. Part of the speech feature

categories	feature1	feature2	feature3	......	feature24
1	-14.82713171	-3.001085509	1.520907785	......	0.213976743
1	-16.22886124	-2.801874337	-0.4108157	......	0.320707761
1	-15.1242838	-2.59871264	-0.359965674	......	0.596002471
1	-15.70314246	-2.530056621	0.255003254	......	0.491487215
......	......	......	......	......	......
2	-7.124675028	-3.291221296	-1.989485515	......	-0.190324063
2	-7.491168268	-3.876559747	-6.284460902	......	-0.164035009
2	-7.008073253	-3.046781147	-4.248054958	......	0.193836521
2	-6.470520074	-2.935980946	-3.135171374	......	0.114851434
......	......	......	......	......	......
3	-14.54861636	-2.546153699	-1.364847328	......	-0.074555423
3	-15.05664211	-1.842875047	-1.12674255	......	-0.20296232
3	-15.00872378	-0.756188396	1.334223717	......	-0.33626362
3	-12.49013913	-3.718298992	-2.615074106	......	0.22889579
......	......	......	......	......	......
4	-1.217017482	1.080606606	1.841872375	......	0.653276583
4	-3.388706732	3.036554219	0.355659097	......	0.547523167
4	-6.15672141	1.608502459	0.868304617	......	0.327454193
4	-7.527673153	0.994359732	4.094927525	......	0.248972345

Table 2. The matrix of the four types of speech

categories	matrix
1	[1 0 0 0]
2	[0 1 0 0]
3	[0 0 1 0]
4	[0 0 0 1]

2.4 Algorithm

The algorithm's flowchart in Fig. 3. In this processing, at the first Decide the Structure of BP Network, the input patterns and the output patterns etc. and then Initialize the threshold and weights of the BP network. Obtain the optimum threshold and weights using PSO, This network training procedure continues until the BP neural network weights are updated sufficiently.

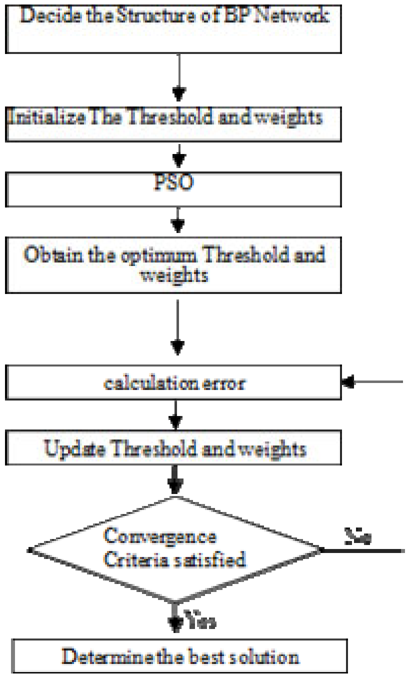

Fig. 3. Flowchart of the optimization

The prediction accuracy of the network is related to the selection of the number of nodes in the hidden layer of the network. The number of nodes is too large, training time is long, and over-fitting occurs easily. The number of nodes is too small and the training time is short, but the network training ends prematurely and the accuracy is low. The best number of nodes in the hidden layer can be computed as:

$$k < \sqrt{m+n} + l \tag{10}$$

Where k is the best number of nodes in the hidden layer, m is the number of input nodes, n is the number of output nodes, l is a constant of [0, 10]. In our BP neural network training, m=24, n=4, k=7. The structure of BP network as shown in Fig. 4.

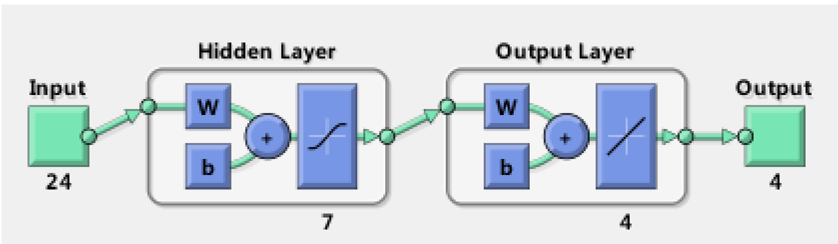

Fig. 4. The structure of BP network

The training dataset is N2000×24, we use the 1500 data as the training data, the lest 500 as the test data. The fitness of BP neural network F can be computed by:

$$F = \min\left(\sum_{i=1}^{q} \left| y_i - o_i \right|\right) \tag{11}$$

Where q is the number of the output nodes, y_i is the ith node's output of the BP neural network, o_i is the ith node's predict output of the BP neural network.

3 Experiment

The Parameters of the BP neural network with backpropagation training functions that will be applied to in the study as shown in Table 3.

Table 3. The Parameters of the BP neural network (matlab)

Parameter	Specification
Transfer function of *i*th layer	tansig
Backprop network training function	trainlm
Performance function	mse
net.trainParam.epochs	50
net.trainParam.lr	0.1
net.trainParam.goal	0.01
net.trainParam.show	100
net.trainParam.mc	0.95

The computer adopted in the experiment is Intel 2370M 2.4G with 16G RAM and a runtime environment of MATLAB R2012b. The mean squared error (mse) of BP as shown in Fig. 5. The mean squared error reaches the minimum value 0.057505 at the 17th epoch. The mean squared error (mse) of PSO-BP as shown in Fig. 6. The mean squared error reaches the minimum value 0.050207 at the 8th epoch. PSO-BP is better than BP neural network in the training epochs and mean squared error.

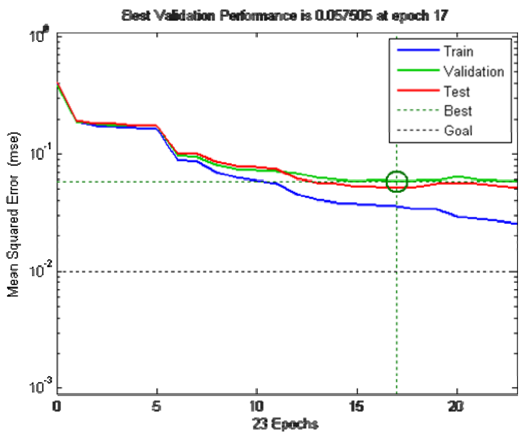

Fig. 5. The mean squared error (mse) of BP

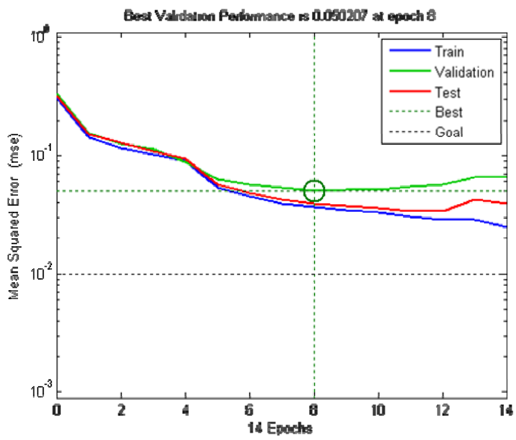

Fig. 6. The mean squared error (mse) of PSO-BP

Through experimental data analysis and comparison, PSO-BP has better training times and mean square error than unoptimized BP neural network. The best fitness as shown in Fig. 7. As illustrated in Fig. 8 and Fig. 9, the error of PSO-BP test is better than the BP test.

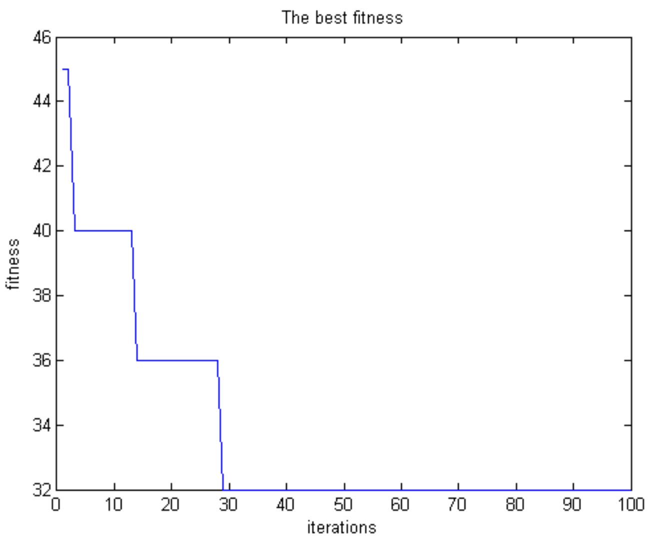

Fig. 7. The best fitness in the simulation

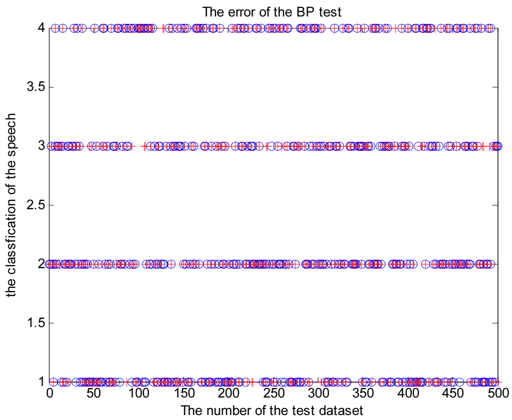

Fig. 8. The mean squared error (mse) of BP

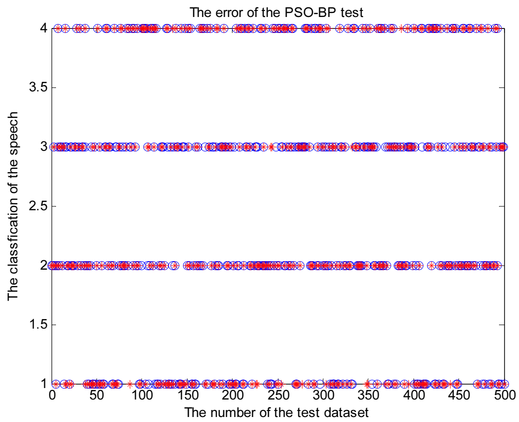

Fig. 9. The mean squared error (mse) of PSO-BP

The classification accuracy of speech classes is shown in Table 4. The classification accuracy of the first type of speech was improved by 1.89%, the second type was almost the same, the third type accuracy rate was increased by 4.56%, the fourth type accuracy rate was increased by 2.04%. The results of simulation show that the PSO-BP neural network achieves better classification effect in the classification of speech feature signals.

Table 4. The classification accuracy of speech classes

classes	1	2	3	4
The classification accuracy (BP)	0.8123	0.9907	0.9245	0.9561
The classification accuracy (PSO-BP)	0.8312	0.9910	0.9701	0.9765

4 Conclusion

The BP neural network was used to classify the speech feature signals. Four types of different music category data were selected. A BP neural network structure of 24-7-4 was established. The BP neural network with 24 dimensional feature data as input and four categories as output was used. The network classification model uses the PSO algorithm to optimize the weights and thresholds of the neural network. The PSO-BP neural network not only enhances the accuracy, but also reduces the computation.

However, the paper has some deficiencies, such as the optimization of BP neural network focuses on the weights and thresholds. On this basis, our next step is to optimized the structure of BP neural network.

Acknowledgements

This work was partially supported by 2018 Shenzhen Discipline Layout Project (JCYJ20170815145900 474), and Shenzhen Basic Research Project (No. JCYJ20170818115704188).

References

[1] Z.-B. Wang, Y.-N. Kong, Y.-C. Liu, Q. Zhang, Prediction of junction temperature for high power LED by optimizing BP neural network based on genetic algorithm, Journal of Optoelectronics·Laser 25(7)(2014) 1303-1309.

[2] G.-G. Wang, L.-H. Guo, H. Duan, L. Liu, H.-Q. Wang, Target threat assessment using glowworm swarm optimization and BP neural network, Journal of Jilin University (Engineering and Technology Edition) 43(4)(2013) 1064-1069.

[3] B. Gong, F. Zeng, A research on the improvement of artificial fish algorithm and its optimization on BP neural network, Journal of Hunan University of Science & Technology (Natural Science Edition) 31(1)(2016) 86-90.

[4] X. Li, J.-F. Cui, J. Xiong, S.-X. Yang, S.-D. Yang, Power demand forecasting based on BP neural network optimized by

clonal selection particle swarm, Journal of Hunan University (Natural Sciences) 35(6)(2008) 41-45.

[5] Y. Zhang, X. Gao, S. Katayama, Weld appearance prediction with BP neural network improved by genetic algorithm during disk laser welding, Journal of Manufacturing Systems 34(2015) 53-59.

[6] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proc.the IEEE International Conference on Neural Networks, 1995.

[7] E. Mezura-Montes, C.A.C. Coello, Constraint-handling in nature-inspired numerical optimization: past, present and future, Swarm and Evolutionary Computation 1(4)(2011) 173-194.

[8] Z.-Y. Li, J.-Y. Wang, C. Guo, A new method of BP network optimized based on particle swarm optimization and simulation test, Acta Electronica Sinica 36(11)(2008) 2224-2228.

[9] B.H.M. Sadeghi, A BP-neural network predictor model for plastic injection molding process, Journal of Materials Processing Technology 103(3)(2000) 411-416.

[10] C. Ren, N. An, J. Wang, L. Li, B. Hu, D. Shang, Optimal parameters selection for BP neural network based on particle swarm optimization: a case study of wind speed forecasting, Knowledge-Based Systems 56(2014) 226-239.

[11] C.X.J. Feng, A.C. Gowrisankar, A.E. Smith, Z.G.S. Yu, Practical guidelines for developing BP neural network models of measurement uncertainty data, Journal of Manufacturing Systems 25(4)(2006) 239-250.

[12] Z. Li, X. Zhao, BP artificial neural network based wave front correction for sensor-less free space optics communication, Optics Communications 385(2017) 219-228.

[13] G. Yang, C. Gu, Y. Huang, K. Yang, BP neural network integration model research for hydraulic metal structure health diagnosing, International Journal of Computational Intelligence Systems 7(6)(2014) 1148-1158.

[14] Z.-C. Yang, User-online load movement forecasting for social network site based on BP artificial neural network, Journal of Computers 8(12)(2013) 3176-3183.

[15] T.D. Jorgensen, B.P. Haynes, C.C. Norlund, Pruning artificial neural networks using neural complexity measures, International Journal of Neural Systems 18(5)(2008) 389-403.