International Journal of Information Studies

Print ISSN: 2349-8161
Online ISSN: 2349-817X

ED 2024: 15 (2)

https://doi.org/10.6025/ed/2024/15/259-67

Application Analysis of Improving English Teaching Quality Based on PSO Algorithm

Wang Jin
Monash University Malaysia
Jalan Lagoon Selatan,
Selangor Darul Ehsan, Malaysia
khdgfds23432@protonmail.com

ABSTRACT

With the acceleration of globalization, English, as the universal language for international communication, has become increasingly important in the field of education. This article studies the application analysis of particle swarm optimization algorithm (PSO) based on English teaching quality improvement. The aim is to use the PSO algorithm to optimize the English teaching process and improve teaching quality. This article first introduces the basic principle of the PSO algorithm and English teaching quality evaluation indicators and then elaborates on how to apply the PSO algorithm to the process of improving English teaching quality, including teaching plans, classroom teaching, extracurricular tutoring, and teaching evaluation. Finally, the effectiveness and feasibility of the English teaching quality improvement method based on the PSO algorithm were verified through experiments

Keywords: Particle Swarm Optimization, Bp Algorithm, English Teaching, Research

1 Introduction

Particle swarm optimization, also known as foraging algorithm (POS), is derived from the intelligent study of the laws of nature proposed by Kennedy and Eberhart in the mid 90s of last century, inspired by the [1] of birds. The most effective strategy for birds to search for food is to search for [2], the nearest bird in the current situation. In the following research, scholars gradually established the mathematical model of particle swarm optimization algorithm. The particle swarm algorithm model's structure is simple, with high search precision and convergence of short cycle. It is a kind of intelligent evolutionary algorithm, from the point of random solutions, the optimal solutions for finding the global calculation by iteration, the fitness to evaluate the optimal solution value [3]. Particle swarm algorithm is better than

genetic algorithm, simulated annealing algorithm and other intelligent algorithms for computing simple rules. It does not require complex operations such as crossover and mutation, only by following the current search of the optimal value will be able to find the global solution [4]. Artificial neural network (ANN) has the function of simulating the brain learning and storing large amounts of information. It is an important direction of artificial intelligence research, and has achieved extensive results in many industries and fields. Neural network, as an abstract mathematical model, uses a large number

2 State of the Art

Particle swarm optimization (PSO), as a parallel algorithm, is a representative of new intelligent optimization algorithms. In the algorithm, each particle is equivalent to a bird in the flock, it is equivalent to the specific problem of a potential solution, by using the fitness function which is the objective function to calculate the fitness of each particle, index selection and evaluation of its [5]. To search for the optimal solutions in the population, each particle velocity vector by itself in continuous learning, in the process of changing the particle velocity vector, the particle in the feasible solution for the optimal solution space [6]. There are three basic concepts involved here. One is the particle. As the most basic component of the algorithm, it embodies the essence of the optimization problem and represents the feasible solution to each problem. The particle corresponds to a point in the multidimensional space, and the movement of the point is the process of searching for the optimal value [7]. The second factor is population. The number of birds in the population is the number of particles. Population size represents the size of feasible solutions. The particles in the population have the ability to share and learn from each other in a group effect and finally find the optimal solution [8]. The third is the fitness function. That is to say, the objective function of the algorithm is presented in the particle, the objective function can be substituted, and the particle's fitness value, which can evaluate the performance of the particle; in iteration termination, the best fitness of the best particle is the global solution of [9].

BP neural network is the most attractive and widely used research model in artificial neural networks. The BP neural network adopts an error back propagation algorithm model and uses a [10] with a tutor training method. BP neural network can provide a simple way for nonlinear modeling of complex systems, to achieve arbitrary precision close to arbitrary nonlinear mapping, through independent learning changes in the internal network connection value in order to adapt to the change of the system, which has good fault tolerance and robustness, multiple input and output structure model, and system state make better use of multiple variables. The introduction of the particle swarm algorithm to optimize the BP neural network is mainly to solve the control problem of the inertia weight and learning factor algorithm, which can effectively improve the global search results.

3 Methodology

3.1 Particle Swarm Optimization Algorithm

The basic principle of particle swarm optimization is very similar to that of birds looking for food. Suppose that in a D-dimensional search space, there are n particles in a population. Here, the D dimensional vector of the particle is expressed as i, which can show the I particles in the range of search space location in the D dimension, which was also a solution to the algorithm problem. In the use of the fitness

function corresponding to each particle is calculated, the fitness value is the evaluation of the particle as a standard value of candidate solutions. When the value of the objective function is the minimum, it shows that the fitness value is the best, and the closer to the minimum, the better the performance is.

$$\begin{aligned} V_{id}^{k+1} &= wV_{id}^{k} + c_1 r_1 \left(P_{id}^{k} - X_{id}^{k}\right) + c_2 r_2 \left(P_{gd}^{k} - X_{id}^{k}\right) \\ X_{id}^{k+1} &= X_{id}^{k} + V_{id}^{k+1} \end{aligned} \tag{1}$$

The velocity of the i-th particle is set as $\mathbf{V}_i = (V_{i1}, V_{i2}, \ldots, V_{iD})^T$, the individual extremum is set as $P_i = (P_{i1}, P_{i2}, \ldots, P_{iD})^T$, and the population extremum of the population is set as $P_g = (P_{g1}, P_{g2}, \ldots, P_{gD})^T$.

In each iteration, each particle updates its position and speed by using the extreme values of individuals and groups and is represented by the formula (1). Among them, W is inertia weight, $d = 1, 2, \ldots, D; i = 1, 2, \ldots, n$, K is the current iteration number, V_{id} is the speed of particles and c_1, c_2 are generally called self-learning factors and global learning factors, on empirical value $c_1 = c_2 = 1.49445$. c_1 is the ability of particles to learn from themselves, and c_2 represents particles to learn from populations. r_1 and r_2 are the random numbers between $[0, 1]$. Figure 1 shows the procedure of the standard particle swarm optimization.

From the flow and basic formula of particle swarm optimization algorithm, it can be seen that the main results of the algorithm are three parts. One is the current iteration speed of particles, and the other is the cognitive ability of individual particles, which reflects the learning state of the particle itself. It is the key to obtain better search ability and prevent particle search from moving into local search. The third is the sharing of social information between particles. Under the combined action of the three parts, the particles adjust their individual states constantly, thus searching for the global optimal solution.

When the learning factor $C_1 = C_2 = 0$, it shows that the particle's individual learning ability and social learning capacity are not present, and the particles will only continue to fly at constant speed, so that the optimal solution cannot be found. At the same time, when $C_1 = 0$, the particle has no individual cognitive learning ability, which makes the particle swarm algorithm model possess only social cognitive ability. At this time, the whole search process of particles falls into the local optimum. When $C_2 = 0$, the particle's social learning cognitive ability is not present, and the particle swarm algorithm model has only individual learning cognitive ability, making it very difficult to find the optimal solution.

In the state of $C_1 = C_2 = 0$, the model will search the current optimal position and the most global position to control the particle speed. At this time, the particle search cycle and time will increase significantly, resulting in reduced efficiency. Figure 2 is a diagram of the particle velocity update, which represents the possible velocity state of the particles.

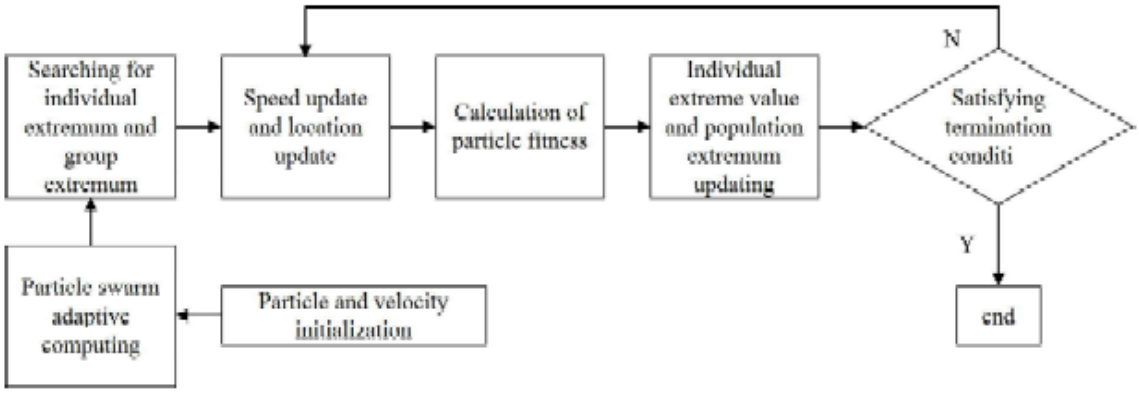


Figure 1: Flow chart of particle swarm optimization algorithm

In order to avoid the inefficiency of blind search in particle swarm optimization, it is necessary to limit the particle velocity to the V min and$V_{\max}$ interval. $V_{\max}$ represents the maximum velocity, that is, the maximum value of the coordinates of the particles that can occur in each iteration. If is too large, the particle will miss the optimal solution, resulting in the divergence of the iterative process. If $V_{\max}$ is too small, it will directly reduce the search speed of the population, or make the population search become the search for the local optimal value, and then cannot find the global optimal value. Assuming that X represents the position vector of a particle,$V_{\max} = X_{\max} - X_{\min}$ the maximum velocity of the particle is the best-performing state at that time. Figure 2 is a diagram of the particle velocity update indicating the possible velocity state of the particles. In order to improve the learning ability of learning factors, the calculation of inertia weight should be carried out. The inertia weight is a linear weight, and the weight value is mainly determined by experience value.

In the process of the inertia weight from the large to the small, the global search ability of the particles in the initial state is guaranteed, and the local search ability is also retained. Fig. 3 is the influence diagram of the inertia weight on the number of iterations.

3.2 Particle Swarm Optimization Algorithm to Optimize BP Neural Network

The basic algorithm of the BP neural network is to teach teacher learning. The basic learning algorithm of its main calculation steps is the initialization of the weights w and θ, and to determine:

- the input layer to the hidden layer neuron connection weights w_{ij}, and
- the hidden layer to the output layer weights w_{jk}.

The hidden layer threshold is set as θ_j, and the output layer neurons θ_k are set in a given threshold range (0, 1) between the smaller values. The input vector of the input value is determined as:

$$x = (x_1, x_2, \ldots, x_m),$$

and the input vector y:

$$y = (y_1, y_2, \ldots, y_u).$$

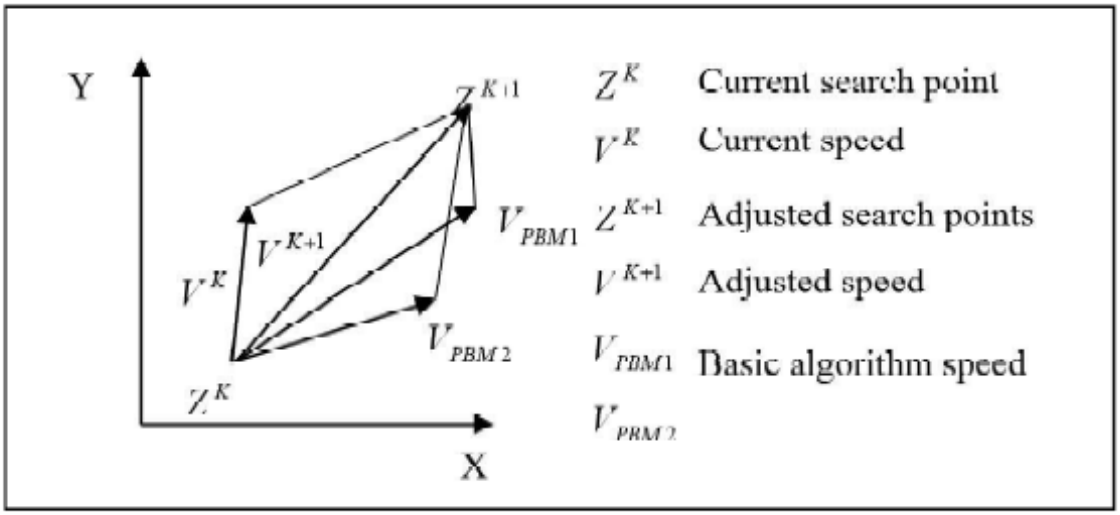


Figure 2: Sketch map of particle velocity update

for the forward calculation, or according to:

$$y_k = f\left(\sum_{j=1}^{u} w_{jk} x'_j - \theta_k\right), \quad k = 1, 2, \ldots, n,$$

to go through the reverse calculation. The error between the output...

The neuron output value and the expected output value are calculated. If the error result is in line with expectation, the training is finished. If the gap is too large, it will enter the reverse calculation link of model calculation again. After repeated correction function calculation, the weight to meet the requirements is obtained, the model calculation is finished, and the signal is output. Fig. 4 is a BP network model structure diagram.

The advantage of BP neural network is obvious, but the deficiency is obvious. It is mainly in the practical application that the convergence period is long, the speed is slow, easy to enter local search, the optimal value becomes the local minimum, and the generalization ability is not strong. Therefore, particle swarm optimization (PSO) is introduced to optimize the BP neural network to improve these defects. The principle of an improved particle swarm optimization algorithm for BP based on a neural network is to find a suitable set of BP network weights using a particle swarm algorithm, that is let the corresponding weights and thresholds of individual particle swarm particle location and BP network, the BP neural network weight and threshold value to decide calculation of dimension particle algorithm, BP network error as particle swarm algorithm to the initial value, if the smaller the error performance of the better quality of said particles, through the standard to test particle swarm update. The particle swarm algorithm is effective for the improvement of the evolution of its combination of BP network, in order to get a smaller error, the optimization algorithm is in error meet the needs of the design or the number of iterations is completed, the initial weights of BP network algorithm and output end result, that is the optimal position of the particle. Figure 5 is a flow chart of particle swarm optimization BP algorithm.

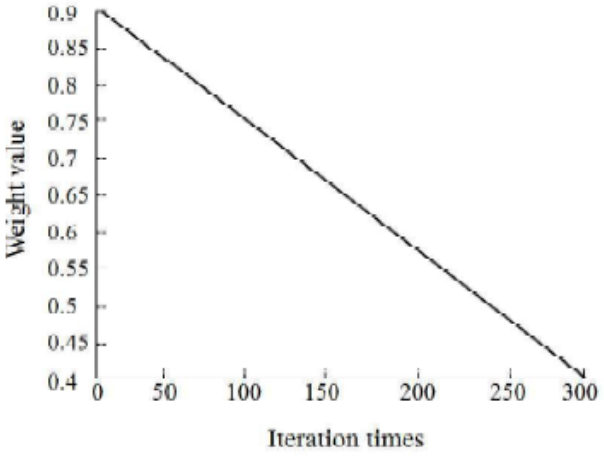


Figure 3: The change of weight and iteration number

The main idea of particle swarm optimization is to consider whether the particle will be affected, not enter the optimal region, and premature and premature convergence state. In order to improve the convergence ability of the basic particle swarm optimization algorithm. On the basis of the particle velocity formula, this paper introduces the inertia weight W to obtain the optimization algorithm formula of particle velocity, such as formula (2) to express.

$$X_{i,j}(t+1) = V_{i,j}(t+1) + X_{i,j}(t) \tag{2}$$

In order to find the global optimal solution, not only the appropriate inertia weight is needed, but also the learning factor is set. If c_1 is large and c_2 is small, particles will easily fall into the local search by self-learning.

If c_1 is smaller than c_2, particles will reduce the convergence of the algorithm, and the algorithm is not accurate enough.

So, the linear decrement function is introduced to reduce the learning factor with the increase of iterations. In order to achieve the regulation of particle motion, assuming $c_1 = c_2 = c$, then c is given as:

$$c = [c_{\min}, c_{\max}],$$

where $c_{\min}$ is the minimum learning ing factor value, max c is the maximum learning factor value. The optimization formula for the learning factor is shown in formula (3).

$$V_{i,j}(t+1) = \omega V_{i,j}(t) + c_1 \cdot r_{1,i,j}(t) \cdot (P_{i,j}(t) - X_{i,j}(t)) + c_2 \cdot r_{2,i,j}(t) \cdot (G_{i,j}(t) - X_{i,j}(t)) \tag{3}$$

Among them, T represents the current iteration number, and P denotes the positive coefficient of the hypothesis. At $t = 1$, in time $c = c_{\max}$, and when $t \to \infty$, $c = c_{\min}$.

Under this formula, the range of the learning factor is specified as:

$$c \in [c_{\min}, c_{\max}].$$

The main idea of particle swarm optimization is to consider whether the particle will be affected, not enter the optimal region, and premature and premature convergence state. In order to improve the convergence ability of the basic particle swarm optimization algorithm. On the basis of the particle velocity formula, this paper introduces the inertia weight W to obtain the optimization algorithm formula of particle velocity, such as formula (2) to express.

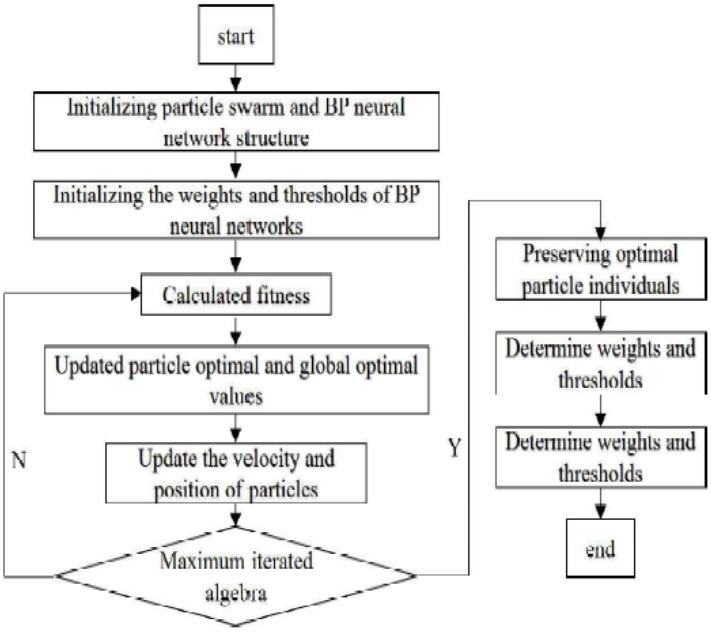


Figure 4: Flow Chart of Particle Swarm Optimization Bp Algorithm

In order to improve the particle search speed at the later stage and find the accurate global optimal solution, the local search is accurate. It is to estimate the sum of all applicable values in each iteration to solve the mean value and to judge the difference between the current adaptation values in each iteration. If the difference is less than zero, the average is:

$$c = \frac{(c_{\max} - c_{\min})}{t^p} + c_{\min} \tag{4}$$

4 Results

In this paper, the particle swarm optimization (PSO) algorithm is used to improve the English teaching effect of the BP algorithm in the Department of English of M University. First of all, the student's English learning ability data collection and sample selection. The sample data of 50 students from the English department were selected as training data, and the data of 10 students were selected as the test data, which were compared and analyzed by the model. Firstly,the number of nodes in the input layer of the BP neural network is set to 5, and the output energy node number is set to 1. The number of nodes in the hidden and hidden layers has a great impact on the performance of the whole network. In different network applications, the number of hidden nodes required by the hidden layer should be carefully considered according to the actual situation. In this paper, the experimental method is used to determine. The basic principle is that the number of nodes in the hidden layer can be reduced as much as possible when the input and output connections can be correctly responded to so that the network structure can

be effectively controlled. When the number of input nodes is greater than the number of nodes in a small network, the number of nodes in the best-hidden

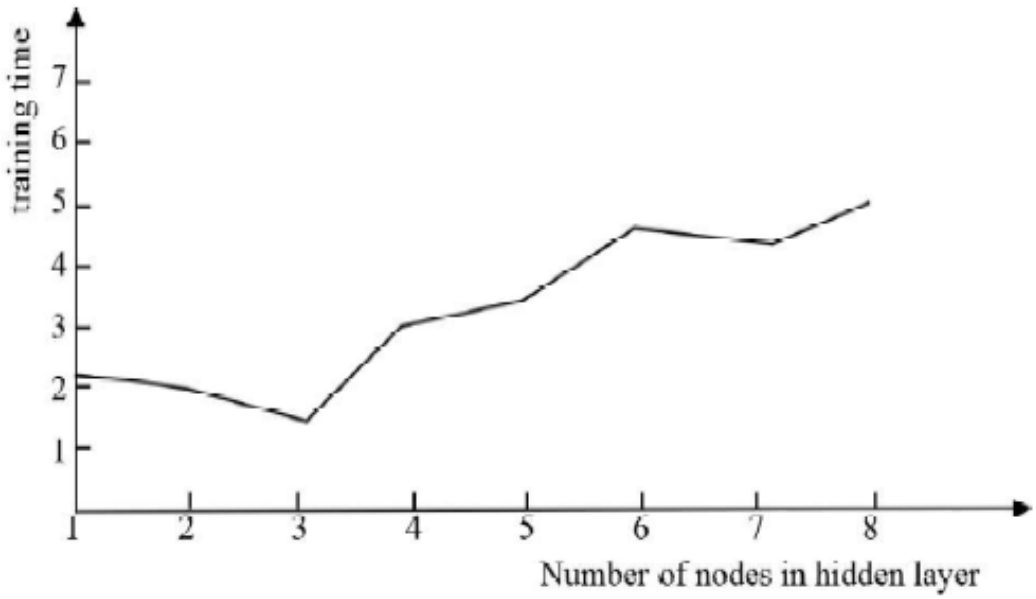


Figure 5: The change of training time when the number of hidden layer nodes increases

In this paper, the simulation test uses the DevC++ software on the Windows operating system platform to achieve, and the implementation of data mining classification processing is applied to the memory of 256M computer. In the sample set of learning characteristics of students learning English, the experiment of LCAM model is mainly carried out. The IPSOBP algorithm is used to train the neural network. After the data are pre-processed, the weights and thresholds of the initialization of the network are set. The initial weights, acceleration systems and inertia factors of the particles are randomly generated, and the constant premature factor is 0.01. The particle number of 5, 10 and 20 is selected to do experiments respectively, and the particle swarm algorithm proposed in this paper is used to improve the running efficiency of BP algorithm under different particle number. The results are shown in table 1.// Patterns are the number of samples, ParticleCoun is the number of particles, Epoch represents the number of iterations at the end of the training, Pbest_Number is the encoding of the global optimal particle, which corresponds to the weights of the neural network, and MSE is the mean square error value. From the table, it can be seen that the initialization parameters of IPSOBP algorithm are randomly selected, so the periodic value and MSE value are different when the training result is correct in each execution. When the number of particle swarms is different, the cycle of BP neural network training is not good enough, which shows that the algorithm has better convergence ability. But from the Epoch and MSE values, we can see that the effect of training convergence and the number of particle swarms does not have a direct impact. in which J is the number of hidden layer nodes, M1 is the number of hidden layer nodes, and N is the number of nodes in the input layer. After calculation, the number of nodes in the hidden layer is 2. In order to avoid the difference caused by random factors in the algorithm, the training time of each network is 5 times. Thus, the training time of the network is increased when the number of nodes in the hidden layer increases. As shown in figure 5.

Algorithm	Patterns	Particle Count	Epoch	Best Number	MSE
IPSOBP	30	5	245	3	0.03115
		10	436	6	0.04789
		20	299	5	0.03879

Table 1: Experimental Results of Algorithm Simulation

Patterns are the number of samples, ParticleCoun is the number of particles, Epoch represents the number of iterations at the end of the training, Pbest_Number is the encoding of the global optimal particle, which corresponds to the weights of the neural network, and MSE is the mean square error value. From the table, it can be seen that the initialization parameters of IPSOBP algorithm are randomly selected, so the periodic value and MSE value are different when the training result is correct in each execution. When the number of particle swarms is different, the cycle of BP neural network training is not good enough, which shows that the algorithm has better convergence ability. But from the Epoch and MSE values, we can see that the effect of training convergence and the number of particle swarms does not have a direct impact.

Agorithm	Paticle	MSEt	MSEg	Et	Eg
	5	0.0234	0.0453	6.8	14.5
IPSOBP	10	0.0315	0.3078	13.4	28.7
	20	0.0458	0.2897	11	29.6

Table 2: Experimental Results of Algorithm Simulation

Table 2 is the experimental result of 10 test samples. The data also show the expected value and the error value of the whole algorithm is small, which has good guidance and reference for improving English teaching.

5 Conclusion

The particle swarm algorithm through the study of birds' search for food in the process, to select the optimal group under the target value, the model of simple algorithm, the characteristics of single rule has been widely used in many fields of artificial intelligence, multi-objective optimization, data mining, constraint optimization etc. Therefore, this paper studies the effect of English teaching improved BP algorithm based on particle swarm algorithm, to improve the teaching methods and the quality of the target with the optimized algorithm. After analyzing the operation process and model the advantages of the BP neural network algorithm, aiming at the defects of the neural network model, introducing the particle swarm algorithm to improve the structure of the. And from the point of improving the accuracy of particle swarm algorithm, particle swarm algorithm, learning factor formula of inertia weight formula was improved, optimized with adaptive degree of formula, in order to promote better play the role of hybrid algorithm. Then, the improved particle swarm optimization (PSO) algorithm BP is applied to M College English teaching. The application effect of particle swarm optimization to BP algorithm is obtained. The application results show that the particle swarm optimization algorithm based on BP algorithm parameters optimization is successful.

References

[1] Zhang, Y., Wang, S., Ji, G. (2015). A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering*, 2015 (1), 1-38.

[2] Sun, J., Wu, X., Palade, V., et al. (2015). Random drift particle swarm optimization algorithm: convergence analysis and parameter selection. *Machine Learning*, 101 (1-3), 345-376.

[3] Zhang, L., Tang, Y., Hua, C., et al. (2015). A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques. *Applied Soft Computing*, 28 (C), 138-149.

[4] Cheng, R., Jin, Y. (2015). A social learning particle swarm optimization algorithm for scalable optimization. *Information Sciences*, 291 (6), 43-60.

[5] Yang, Z., Wei-Sheng, X. U., Ning W, et al. (2015). Multi-objective Siting and Sizing of Distributed Generation Planning Based on Improved Particle Swarm Optimization Algorithm. *Transactions of China Electrotechnical Society*, 19, 78.

[6] Yan, F. G., Fu, L. P. (2016). The Research & Application on the Credit Appraisal Model of Architecture Enterprise by the GA-BP Neural Network. *Key Engineering Materials*, 693, 2022-2030.

[7] Jia, W., Zhao, D., Shen, T., et al. (2015). An optimized classification algorithm by BP neural network based on PLS and HCA. *Applied Intelligence*, 43 (1), 1-16.

[8] Nouiri, M., Bekrar, A., Jemai, A., et al. (2015). An effective and distributed particle swarm optimization algorithm for flexible job-shop scheduling problem. *Journal of Intelligent Manufacturing*, 1-13.

[9] Zhang, S., Lv, J., Yuan, X., et al. (2016). BP Neural Network with Genetic Algorithm Optimization for Prediction of Geo-Stress State from Wellbore Pressures. *International Journal of Computational Intelligence & Applications*, 15 (03), 1650013.

[10] Hongmin, X. U., Guo, Z., Xiaoyu, L. I., et al. (2016). Prediction on crossing accident based on BP neural network of optimized Genetic Algorithm. *Railway Computer Application*, 32 (7), 5-29.