Scholar Publishing Group
International Journal of Neural Network
https://doi.org/10.38007/NN.2020.010103
ISSN 2790-2692 Vol. 1, Issue 1: 17-23

Multilayer Performance Improvement of Feedforward Neural Networks

Haoxian Zhan*

Zhejiang University, Hangzhou, China

**corresponding author*

Keywords: Feedforward Neural Network, Bp Neural Network, PSO Algorithm, Performance Improvement

Abstract: The feeder neural network is one of the widely used neural networks. It can transfer input data from the input layer to the output layer without feedback. The feeder neural network is improved by using different techniques and many network models with different functions are obtained. In this thesis, the improvement of the performance of the power supply nervous network is studied and applied. This thesis analyzes two types of neural network feeders, namely neural network and self-coded neural network. Then the algorithm is used to solve the two problems of slow convergence and easy to fall to the local minima in the algorithm of the feeder neural network, and the performance comparison experiment with other depth algorithms is carried out. According to the experimental results, the improved chopped feedforward neural network in this paper is an effective deep learning framework.

1. Introduction

As a model of effective realization of artificial intelligence, neural network is an important means in the development of intelligent technology at present by using interconnected structure and efficient way to process information in human brain simulation [1-2]. Due to its advantages of parallel architecture and distributed storage, artificial neural network not only has good adaptability and high fault tolerance rate, but also can realize function approximation and model classification in terms of optimization. It has been widely regarded as the most potential technology that can make great breakthroughs [3]. However, the existing artificial neural network structure is still only a preliminary simplified simulation of biological nervous system, which is far less complex than biological network [4]. In addition, the current structural design of neural networks mainly relies on experience and lacks effective theoretical support. How to efficiently construct neural network structure and give full play to intelligent behaviors such as self-learning and self-organization is a key problem facing many researchers at present, and also has important research significance [5].

For neural network optimization, slow learning is the main problem in its application. How to learn and determine parameters quickly under limited computing resources is also the key of relevant research at present. At the beginning of the emergence of swarm intelligence optimization algorithm, some researchers have applied it to the structural optimization problem of neural networks and achieved relatively ideal results. Subsequently, the research on structural optimization of neural networks has gradually deepened [6].

So far, in terms of the concept of typical Neural networks, researchers have invented and proposed a variety of Network structures, and currently commonly used Network structures include Feedforward Neural Network (FNN) and memory Network [7]. Among them, feedforward network includes typical artificial Neural network and convolutional Neural network with local connection and weight sharing, etc. Memory network includes Recurrent Neural Networks (RNN) and gated LSTM[8]. Among the above network types, the network structure of artificial neural network and memory network has obvious hierarchical sense, and the network node type is single [9]. As convolutional neural network contains network nodes of various functional types, the design difficulty is further increased [10]. Since the neural network was proposed, many scholars have paid attention to it. Error correction is usually used to optimize its parameters, and BP algorithm is one of the most typical methods [11]. However, the gradient descent method used by BP neural network to calculate the parameter correction will lead to its convergence speed is too slow or even not convergence. In the worst case, it may need to traverse the whole search space to find the optimal solution, resulting in low learning efficiency. Therefore, many scholars have put forward corresponding improvement measures, such as conjugate gradient method, quasi-Newton method, orthogonal least squares method and hierarchical learning algorithm [12]. However, such traditional methods can only reflect a local property and require the objective function to be continuously differentiable, which limits the application of this network in optimization problems to a great extent [13].

In this paper, the particle swarm optimization algorithm used to study the optimization problem of stochastic neural network feeder, which provides a new idea for improving the performance of the neural network feeder and algo swarm intelligence optimization number.

2. Feed-forward Neural Network based on Particle Swarm Optimization

2.1. Feed-forward Neural Network

Feed-forward neural network has powerful and diversified functions, which makes it the most popular network. Based on different training methods and different learning styles, feedforward networks have many different variants [14]. Here, two feedforward networks used in this paper are briefly introduced.

(1) BP neural network

The BP neural network is a multilayered neural supply network. In many neural networks, it has the widest range and most application scenarios. The neural network performs forward and backward functions in the data signal according to error [15]. If the output result is different from the expected result in the comparison, the network system shall return the output data information content backwards and adjust the connection weight between the network system layers through the deviation between the output value; and expected value [16].

In the training, BP neural network can think and associate the input data like human brain. By autonomously identifying the relationship between input data and expected output data and actively storing the relationship in the system, it can fit the functional relationship between input and output through constant calculation of automatic adjustment weights [17].

BP neural network has strong logic processing ability, and there is a good nonlinear mapping relationship between input data and output data. It can deal with and solve the internal complex nonlinear problems and has a high generalization ability. When new data is input to the trained network, it can automatically identify the data content and extract the information in the data. Complete the prediction evaluation [18].

BP algorithm consists of two processes: signal forward propagation and error back propagation. Given the name of the sample set $\{x^j, O^j\}$,, x^j is

The input vector, O^j is the corresponding ideal output. Let W denote the network weight and y^j denote the corresponding actual output of x^j. Taking batch learning as an example, the learning process of BP network is described briefly.

Signal forward propagation: sample x^j enters the network from the input layer, is processed by the hidden layer, and is transmitted to the output layer to obtain the corresponding actual output Y^j. When the actual output y^j is inconsistent with the ideal output O^j, learning is transferred to the backpropagation stage of error. In general, the error is defined as follows:

$$E(W) = \frac{1}{2}\sum_{j=1}^{J}(O^j - y^j)^2 \tag{1}$$

Error backpropagation: The error is somehow backpropagated from the output layer to the input layer through the hidden layer. Specific performance is the weight update:

$$W^{new} = W^{old} - \eta \frac{\partial E(W^{old})}{\partial (W)} \tag{2}$$

(2) Self-coding neural network

Self coding neural network is a kind of hidden layer feedforward neural network with a structure very similar to that of three-way neural network. The difference is that the self coding neural network takes the input itself as the ideal output, which represents the input itself and learns the network weight. Automatic coding network training does not require any information other than input data, such as class labels.

From the point of view of mathematical description, the input is ideal and the output means $O^j=x^j$, then the corresponding error function becomes:

$$E(W) = \frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}(x^j - y^j)^2 \tag{3}$$

Similarly, repeat signal forwarding and error back propagation to adjust the weight until the error is less than the specified standard. The automatic coding network can be trained not only by the slope descent method, but also by quasi Newton method (such as L-BFGS) and other optimization algorithms. Automatic encoding network can flexibly export various data expressions, such as rare expressions, compressed expressions, etc. The depth network is constructed through pre-training layer and reverse propagation adjustment for extracting depth characteristics and identifying conduction patterns. Common self-coding networks include the thin self-coding neural network (SAE), which denotes the self-coding neural network (DAE) and the self-coding neural network (DPAE).

2.2. Optimize Feed-forward Neural Network

Compared with the neural network learning algorithm based on the downhill, the traditional overflow learning algorithm needs a large number of hidden cells because of the random selection of the weight of the input layer and the limitation of hidden cells. On the one hand, too many hidden units increase the complexity of the network, easily lead to excessive learning, and affect the generalization ability of the network. In general, a good robust network also has good generalization performance. Good persistence means that the network output is not sensitive to input changes. The network has good fault tolerance and anti-interference ability.

When traditional ultrafinite learning machines randomly set input layer weights and implicit unit thresholds, they choose the optimal weights and thresholds with low probability, so they must be considered.

The initial data set is divided into training set and test set, and the training set is divided into training set and verification set.

Initialize population. Each component of each particle represents the weight of the input layer and the hidden lower limit of the single-layer candidate neural network. The components of each particle are randomly selected between [-1,1]. Initialize the initial speed, maximum and minimum flight speeds, population size and maximum repetitions of each particle.

Calculate the appropriate value for each particle. According to the weight of the network input layer and the hidden element threshold represented by each particle, combined with the training data set, the weight of the corresponding network output layer is calculated.

Each particle updates its position, creating a new population. All the components of the particle must be limited to [-1,1]. When a component of a particle goes outside this range, the direction of the corresponding velocity component goes in the opposite direction.

The above steps are iterated until a predetermined goal is reached or a maximum number of iterations is reached. At this time, the final stochastic feedforward neural network can be obtained from the population global optimal particle. Finally, the trained network is applied to predict unknown samples.

3. Algorithm Simulation Experiment

In order to verify the effectiveness of the algorithm PSO-ELM, the experiment in this section compares the algorithm PSO-ELM with deep ELM and other popular deep learning algorithms.

The software and hardware environment for all experiments is: laptop computer, Intel-I7 2.8ghz processor, 16GDDR4 memory, MATLAB2013b.

This section verifies the effectiveness of the algorithm proposed in this chapter on four relatively small benchmark datasets, including Diabetes, Wine, Satellite Image and image Segmentation.

In the experiment, the number of hidden layers in the deep random feedforward neural network is 3, and N1, N2 and N3 represent the number of hidden units in the first, second and third hidden layers of the deep random feedforward neural network, respectively.

4. Analysis of Experimental Results

4.1. Performance comparison on different datasets

Table 1. PSO-ELM performance on different datasets

	Diabetes	Wine	Satellite image	Image Segmentation
Accuracy of the test	81.54	99.0	89.17	96.52
Training time	1.324	1.335	1.412	1.573

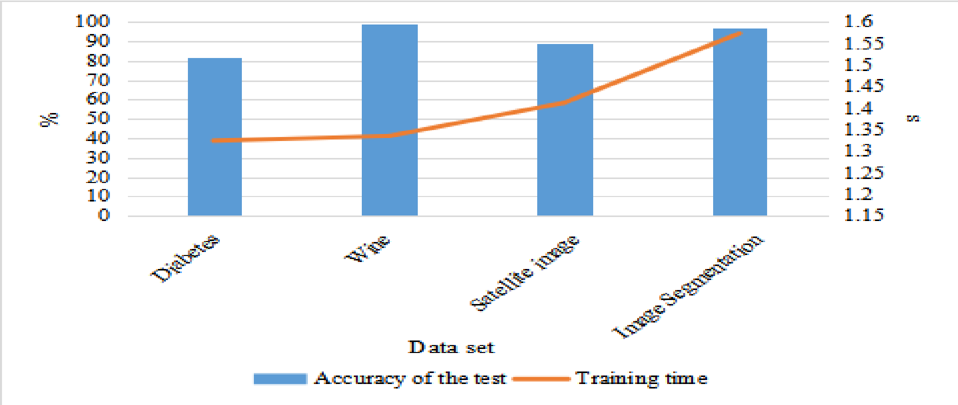

Figure 1. PSO-ELM performance results on four benchmark datasets

As shown in Table 1 and Figure 1, are the performance analysis results of PSO-ELM on four different data sets. It can be seen from the results that the highest accuracy of PSO-ELM on Wine is 99%, and the training time is more than 2S.

Table 2. ML-ELM performance on different datasets

	Diabetes	Wine	Satellite image	Image Segmentation
Accuracy of the test	82.02	100.0	87.45	96.71
Training time	2.569	2.486	90.136	21.315

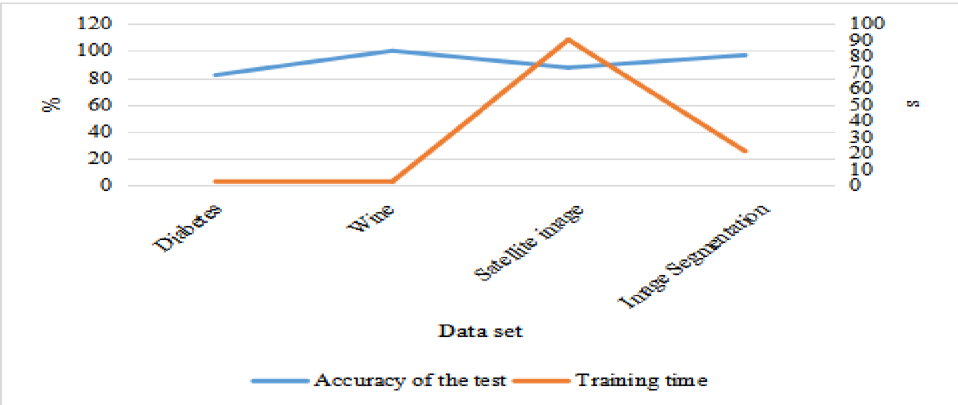

Figure 2. Results of ML-ELM performance analysis on four benchmark datasets

As shown in Table 2 and Figure 2, are the performance analysis results of ML-ELM on four different data sets. As can be seen, the accuracy of ML-ELM was slightly better than PSO-ELM on Diabetes and Wine datasets. However, in terms of training time, ML-ELM takes much longer than PSO-ELM, and even the training time of ML-ELM on Satellite Image dataset reaches 90.136s. This is caused by the parameter optimization of each autoencoder by PSO.

4.2. Compare the other deep learning algorithms

In this paper, PSO-ELM is compared with other popular deep learning algorithms, including SAE, DBN and ML-ELM, on the MINST dataset.

Table 3. Performance of different algorithms on MNIST dataset

Algorithm	Precision	Training time(s)
SAE	97.92%	4316
DBN	98.91%	21450
ML-ELM	97.13%	464
PSO-ELM	98.45%	15268

The experimental comparison of different deep learning algorithms on MINST dataset is listed in Table 3. The convergence accuracy of the feedforward neural network based on PSO optimization on MINST dataset is better than that of SAE and ML-ELM, but slightly lower than that of DBN. In terms of time cost, PSO-ELM is much lower than DBN. However, compared with ML-ELM and SAE, it has a large increase.

5. Conclusion

This paper mainly studies the optimization of learning algorithm with feedforward neural network. On this basis, the research content is summarized and the following problems are found to be further discussed: The performance of data-driven methods is directly affected by the quality of the data itself. This study is conducted on the basis of intact and sufficient process data samples. Therefore, subsequent studies should further consider the situation of missing process modeling data and insufficient sample number (small sample). The applications of deep networks in computer vision, natural language processing and other fields have attracted the attention of many scholars. Deep networks have become a hot spot in the research of neural networks. Therefore, I hope to analyze the convergence of deep autocoding network based on my own research content.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Jansenwinkeln B, Barberio M, Chalopin C, et al. Feedforward Artificial Neural Network-Based Colorectal Cancer Detection Using Hyperspectral Imaging: A Step towards Automatic Optical Biopsy. Cancers, 2020, 13(5):967. https://doi.org/10.3390/cancers13050967

[2] Hardy N F, Buonomano D V. Encoding Time in Feedforward Trajectories of a Recurrent Neural Network Model. Neural Computation, 2018, 30(2):378-396. https://doi.org/10.1162/neco_a_01041

[3] Alfahady N H, Abbo K K, Mahmood E. Development a new conjugate gradient algorithm for improvement feedforward artificial neural networks for predicting the numbers of the Iraqi. Journal of Physics: Conference Series, 2020, 1999(1):012089 (11pp). https://doi.org/10.1088/1742-6596/1999/1/012089

[4] Renisha G, Jayasree T. Cascaded Feedforward Neural Networks for speaker identification

using Perceptual Wavelet based Cepstral Coefficients. Journal of Intelligent & Fuzzy Systems, 2019, 37(1):1141-1153. https://doi.org/10.3233/JIFS-182599

[5] Musikawan P, Sunat K, Kongsorot Y, et al. Parallelized Metaheuristic-Ensemble of Heterogeneous Feedforward Neural Networks for Regression Problems. IEEE Access, 2019, 7(1):26909-26932. https://doi.org/10.1109/ACCESS.2019.2900563

[6] Saleem N, Khattak M I, Al-Hasan M, et al. On Learning Spectral Masking for Single Channel Speech Enhancement Using Feedforward and Recurrent Neural Networks. IEEE Access, 2020, PP(99):1-1. https://doi.org/10.1109/ACCESS.2020.3021061

[7] R, Santhoshkumar, M, et al. Deep Learning Approach for Emotion Recognition from Human Body Movements with Feedforward Deep Convolution Neural Networks - ScienceDirect. Procedia Computer Science, 2019, 152(C):158-165. https://doi.org/10.1016/j.procs.2019.05.038

[8] Malmstrm M, Skog I, Axehill D, et al. Asymptotic Prediction Error Variance for Feedforward Neural Networks. IFAC-PapersOnLine, 2020, 53(2):1108-1113. https://doi.org/10.1016/j.ifacol.2020.12.1310

[9] Fedchenko V, Neglia G, Ribeiro B. Feedforward Neural Networks for Caching. Acm Sigmetrics Performance Evaluation Review, 2019, 46(3):139-142. https://doi.org/10.1145/3308897.3308958

[10] Medus L D, Iakymchuk T, Frances-Villora J V, et al. A Novel Systolic Parallel Hardware Architecture for the FPGA Acceleration of Feedforward Neural Networks. IEEE Access, 2019, PP(99):1-1. https://doi.org/10.1109/ACCESS.2019.2920885

[11] Muir D R. Feedforward Approximations to Dynamic Recurrent Network Architectures. Neural Computation, 2018, 30(2):546-567. https://doi.org/10.1162/neco_a_01042

[12] Murti I. Improvement accuracy of recognition isolated Balinese characters with Deep Convolution Neural Network. Journal of Applied Intelligent System, 2019, 4(1):22-27. https://doi.org/10.33633/jais.v4i1.2289

[13] Gianto R. Neural network-based stabilizer for electric power system stability improvement. Journal of Theoretical and Applied Information Technology, 2017, 95(19):4983-4991.

[14] Herawati S, Negara Y, Latif M. Complete ensemble empirical mode decomposition with adaptive noise integrating feedforward neural network for tourist arrival forecasting. Journal of Physics: Conference Series, 2020, 2193(1):012049. https://doi.org/10.1088/1742-6596/2193/1/012049

[15] [1]Etsuro, Nakamura, Tsuyoshi, et al. Automatic Extraction of Lip Shape Using Feedforward Neural Network. IEEJ Transactions on Electronics, Information and Systems, 2018, 138(12):1604-1612. https://doi.org/10.1541/ieejeiss.138.1604

[16] Alghifari M F, Gunawan T S, Kartiwi M. Speech emotion recognition using deep feedforward neural network. Indonesian Journal of Electrical Engineering and Computer Science, 2018, 10(2):554-561. https://doi.org/10.11591/ijeecs.v10.i2.pp554-561

[17] Kozlenko M, Zamikhovska O, Zamikhovskyi L. Software implemented fault diagnosis of natural gas pumping unit based on feedforward neural network. Eastern-European Journal of Enterprise Technologies, 2020, 2(2 (110)):99-109. https://doi.org/10.15587/1729-4061.2020.229859

[18] Krape P, Vidmar M, Munih M. Distance Measurements in UWB-Radio Localization Systems Corrected with a Feedforward Neural Network Model. Sensors, 2020, 21(7):2294. https://doi.org/10.3390/s21072294