



Multi-parameter online measurement IoT system based on BP neural network algorithm

Weiping Zhang¹ · Mohit Kumar² · Jingqing Liu³

Received: 4 August 2018 / Accepted: 26 October 2018 / Published online: 8 November 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract

Aiming at the problems of long measurement period of online measurement parameters and untimely feedback of IoT technology based on wireless sensor network, this paper designs a multi-parameter online measurement method based on BP neural network algorithm. The collection, analysis, processing and display of parameters are completed through the sensing layer, the network transmission layer and the integrated application layer. The BP neural network algorithm is added to the integrated application layer to optimize the real-time acquisition parameters to shorten the parameter measurement time and accurate prediction. That is, based on the collection of environmental information, by training and learning the BP model with known historical data, it is possible to predict the environmental value at a later time. The known three-layer forward propagation (BP) neural network has the property of approximating the nonlinear curve, and it can achieve good results by predicting the temperature trend. The experimental results show that the system has better ability to monitor and predict the temperature trend. The algorithm simulation experiment shows that the online measurement system based on BP neural network algorithm proposed in this paper can speed up the data collection time, accurately predict the trend of environmental parameters and provide timely warning for potential safety hazards.

Keywords Internet of Things technology · Perception layer · Integrated application layer · BP neural network

1 Introduction

At present, although the Internet of Things technology [1] is widely used in smart enterprises, in the online measurement of the Internet of Things [2], the parameters collected in real time are not analyzed and processed. As a result, the data acquisition time is too long, and the abnormal feedback is not timely. How to build a smart

factory with reliable perception, real-time transmission, intelligent calculation, precise control and information service is an urgent problem to be solved. The basis is to build a real-time sensing and measurement system based on the Internet of Things [3]. At the same time, for intelligent factories, how to ensure the normal and effective operation of various electronic equipments is the guarantee and foundation for maintaining the normal operation of the enterprise. In order to ensure the normal and effective operation of its electronic equipment, connect various sensors, controllers, environmental data, related users, etc. through IoT technology [4, 5] and its IPV6 wireless network communication technology [6], forming a network in which people and things, objects and objects are connected, thereby implementing periodic measurement acquisition and monitoring functions for the environment.

Shuanning Zheng et al. [7] established a real-time measurement system for the environmental Internet of Things subsystem to measure the WEC value of Pingtan Island in Fujian in real time. The system uses four layers to handle data measurements, simultaneous data transfers, data manipulation and application tasks. The results show

✉ Jingqing Liu
liujingqing@zju.edu.cn

Weiping Zhang
zhangweiping@ncu.edu.cn

Mohit Kumar
mohit.kumar@uni-rostock.de

¹ Department of Electronic Information Engineering,
Nanchang University, Nanchang, China

² Faculty of Computer Science and Electrical Engineering,
University of Rostock, Rostock, Germany

³ Zhejiang University College of Civil Engineering and
Architecture, Hangzhou, China

that EIOT enables the system to successfully perform real-time measurements in urban areas. Bandyopadhyay and Sen [2] applied the Internet of Things sensing technology to the mine gas monitoring system, established a perceptual layer distributed wireless sensor network and studied the information fusion algorithm of key technologies in the Internet of Things. According to different fusion levels, applying fuzzy approximation rules, DS evidence theory and fusion method based on gray relational analysis, the intelligent mobile Sink node is designed and developed to provide a good sensing layer solution for mine gas monitoring and measurement application Internet of Things. Experiments show that the application of sensing technology in the Internet of Things can improve information collection efficiency and system tolerance error. Lan and Zhang [8] used the Internet of Things and other technologies to establish an online measurement and remote fault diagnosis system framework based on the Internet of Things. They designed the RFID lithium battery protection board to measure the electric current, voltage, temperature and depth of discharge of the lithium battery, enabling life cycle management and remote fault diagnosis of the lithium battery. In order to achieve real-time temperature monitoring, Cao [9] placed the wireless temperature sensor network at the temperature measurement point of the substation and then transmitted it to the base station wirelessly. Finally, it is transmitted to the management center through the power communication network. The temperature management software monitors the temperature in time, visualizes the display and alarm linkage, etc. and can accurately realize the online temperature monitoring of the substation.

Due to the rapid development of the Internet of Things, a large number of data results can be generated. Extracting advanced patterns from raw data has become an important part of IoT middleware. Predictive analysis is an important technology to support active complex event processing [10]. BP neural network model [11] is one of the most widely used forecasting models and has good nonlinear prediction ability [12]. Therefore, this model is very suitable for data prediction in complex environments such as Internet of Things, and it is often widely used in classification or prediction models of the Internet of Things [13].

Cui et al. [14] proposed the application of BP neural network in the Internet of Things. Experiments show that BP is superior to RFID in the Internet of Things. Yang [15] proposed a fire detection algorithm based on neural network and fuzzy theory. By using the neural network toolbox and fuzzy logic toolbox, the combination of the two algorithms is applied to the conclusion of fire detection. Wang [16] has a slow convergence rate and is easy to fall into the local minimum for traditional BP neural networks. It limits the application of BP neural network in the

Internet of Things. An improved BP algorithm is proposed, which integrates variable learning rate. Experimental results show that the algorithm has better data classification performance.

Based on the above research, this paper designs a multi-parameter online measurement IoT system based on BP neural network algorithm [17, 18] and an online test and early warning system based on BP neural network. According to the actual operating environment, after the environmental data acquisition, a nonlinear BP neural network [19] is adopted as the prediction model. According to the characteristics of the three-layer BP network model [20], any nonlinear function can be approximated. The BP network is learned and trained by using the collected historical environmental data to establish an accurate prediction model system [21]. It not only provides the function of monitoring environmental information, but also effectively predicts the trend of temperature and humidity. The BP neural network is used to establish a prediction model using historical temperature and humidity information. When the trend of the environmental parameters exceeds the threshold, an early warning is made to detect hidden dangers of equipment in advance. In this paper, the temperature prediction results and actual results from the four moments are compared with the gray linear regression combination model. The overall error based on the BP network model is smaller, the prediction accuracy is higher, and it is easier to meet the prediction requirements.

2 Proposed method

The Internet of Things (IoT) is a network that connects embedded electronic devices, wireless sensor networks, software systems and the Internet, enabling it to access vast amounts of data through extensive data exchange with other devices. Each of these physical devices can only be uniquely addressed in all infrastructure of the Internet. With the dramatic increase in the number of devices connected to the Internet, massive amounts of data are gathered through a vast network of sensors distributed throughout the world, making it possible to store and process and analyze the data.

2.1 Internet of Things technology architecture

The Internet of Things technology covers the whole process of acquisition, transmission, storage and application of sensing information, namely the three-layer structure model of the Internet of Things: the perceptual recognition layer, the network construction layer and the integrated application layer, as shown in Fig. 1.

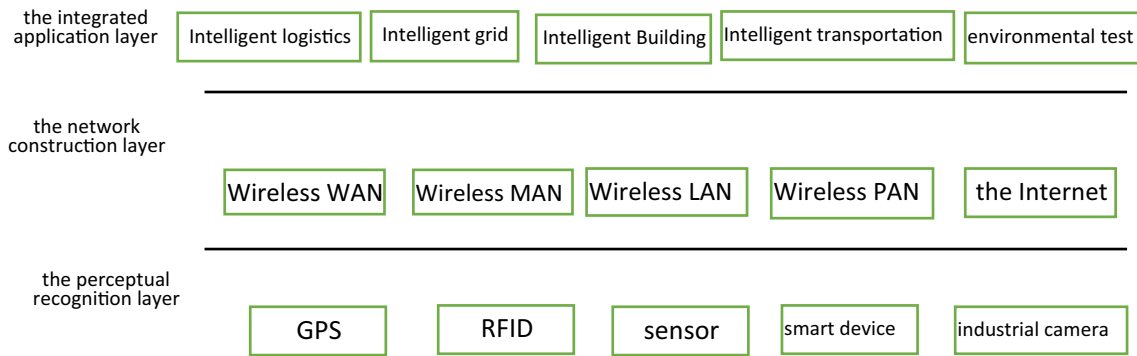


Fig. 1 Frame of IOT system

1. The perceptual recognition layer. It consists of sensors, RFID, QR code, GPS and other data acquisition devices. The main role is to identify objects, collect data information and realize real-time collection and transmission of multi-point data, which is the link between the physical world and the information world. An important feature of the Internet of Things that distinguishes it from other networks is that its information is generated in a variety of ways.
2. The network construction layer. This layer is located between the sensing layer and the service management layer to implement information transmission and processing. It is mainly responsible for ensuring the reliability of information transmission between the identification layer and the service management layer. This guarantee is achieved through various network communication technologies.
3. The integrated application layer. It is at the top of the IoT model framework and is mainly responsible for solving the data processing analysis and human-machine interface problems of the perception layer. The application layer is the interface between the Internet of Things and the user, directly meeting the service needs of various industry sectors. This layer contains various application servers for real-time monitoring of various parameters of the enterprise production process or environment and must also analyze and process the data collected in real time. In this paper, an online measurement technology based on BP artificial neural network model is proposed. BP network model is added to the application layer server for data analysis and processing, in order to shorten the time of parameter measurement and display.

2.2 Artificial neural network model

In the field of machine learning and cognitive science, artificial neural networks are a widely used basic model inspired by biological neural networks for evaluating or

predicting something or an indicator. Artificial neural networks usually consist of interconnected neurons, each of which is shown in Fig. 2.

From the above figure, a_1-a_n are the component values of the input vector of the neuron, and w_1-w_n is the weight of each synapse connected between the neurons. b is the output threshold, and the output of the transfer function is not zero only when the product of the input vector and the weight is greater than the threshold. The function f is a transfer function, and the function f is usually a nonlinear function, usually a $tansig()$ function, a $sigmod()$ function, a $logsig()$ function. The function of a neuron is to find the inner product of the input vector and the weight vector and then pass the transfer function to process the final result, which can be used to calculate the next layer of neurons. Assuming the neuron output is t , the single neuron output is:

$$t = f(a_1w_1 + a_2w_2 + \dots + a_nw_n + b) \tag{1}$$

Through a large amount of data, the neural network learns and trains, thereby adjusting the weights of the synapses of each neuron. The neural network has the ability to have adaptive output. The overall schematic is shown in Fig. 3.

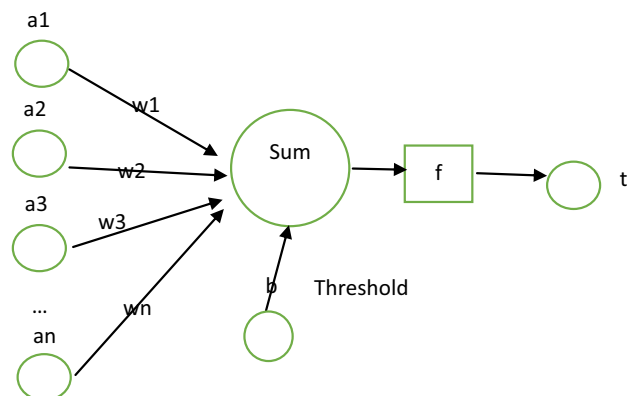


Fig. 2 Single neural

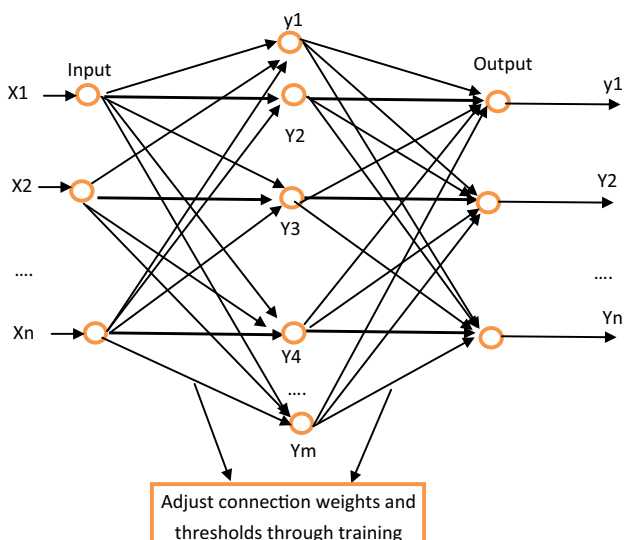


Fig. 3 Neural network

2.3 BP neural network model

According to the direction of error propagation inside the neural network, the neural network can be divided into a neural network with a forward architecture and a neural network with a feedback architecture. As a multilayer feed forward network trained by error inverse propagation algorithm, BP neural network is one of the most widely used neural network models. The learning rule of BP neural network is to constantly adjust the weights and thresholds of the network through backpropagation to minimize the sum of squared errors of the network. The BP neural network model topology includes an input layer, a hide layer and an output layer.

Based on the incremental learning rules (a rule of error correction), the BP algorithm searches and traverses the neural nodes of the entire network, and finally minimizes the error between the actual output of the network and the expected result. The training process of the BP algorithm can be divided into two parts:

1. Part of the forward training process of the sample; the forward training process of the sample means that when the topology of the network (the number of network layers, the number of nodes per layer) is determined, the sample is input from the input layer of the neural network, each neuron weighted sum is calculated in the hidden layer, and the output is passed to the output layer to obtain an output sample. If the network output value deviates too much from the network expectation error, error backpropagation is performed.
2. The error is backpropagated to correct the weights and thresholds of each node of the neural network. The

backpropagation process of the error means that after calculating the network output value and the network expectation value, the total output error is obtained. The total error is backpropagated, and the weights and thresholds of each layer are gradually adjusted during the propagation process, thereby making the error smaller. The BP network continuously repeats this process for training and gradually adjusts the weights and thresholds of each layer until the total error of the network output meets the requirements or the BP network reaches the specified number of trainings before stopping the training. The BP algorithm flowchart is shown in Fig. 4.

Suppose there is a three-layer BP network. The three layers are the input layer vector $X = (x_1, x_2, \dots, x_n)$ as the data entry, the hidden layer output vector $Y = (y_1, y_2, \dots, y_m)$ and the output layer output vector of the predicted result exit $O = (o_1, o_2, \dots, o_l)$, record the expected value of the output $D = (d_1, d_2, \dots, d_l)$. The synaptic weight between the X layer and the Y layer is set to $V = (v_1, v_2, \dots, v_n)$ and

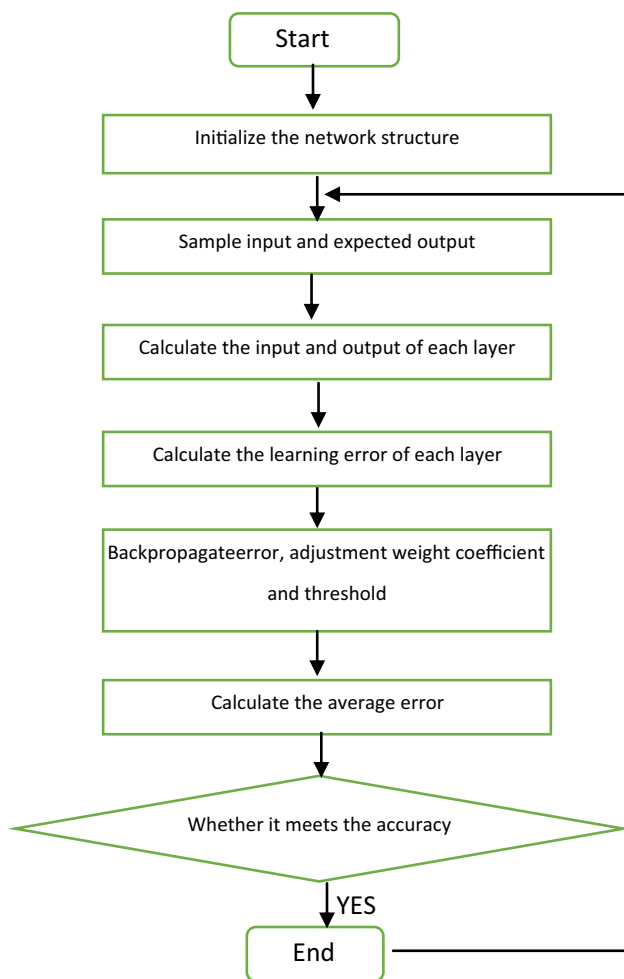


Fig. 4 Flowchart of BP algorithm

the threshold is $B = (b_1, b_2, \dots, b_m)$. The weight between the Y layer and the O layer is denoted as $W = (w_1, w_2, \dots, w_m)$, and the threshold is $C = (c_1, c_2, \dots, c_l)$. The number of neurons included in the input layer X is set to n , the number of neurons included in the hidden layer Y is set to m , and the number of neurons included in the output layer is set to 1. When the sample is set to P, the total error of the BP network output is $E_{RME} = \sqrt{\frac{1}{P} \sum_{p=1}^P (E^p)^2}$. Suppose the input of the training sample is X, the expected output of the training sample is D, and the actual output is O. The calculation flow is as follows:

- (1) Calculate the output of the hidden and output layers in the network:

$$\text{Hidden layer: } y_j = f(V_j^T X + b_j), \quad j = 1, 2, \dots, m$$

$$\text{Output layer: } O_k = f(W_k^T Y + c_k), \quad k = 1, 2, \dots, l$$

- (2) Calculate the overall output error of the BP network:

$$E^p = \sqrt{\sum_{k=1}^l (d_k^p - o_k^p)^2}$$

- (3) Calculate the error between the output layer and the hidden layer in the network:

$$\text{The error of output layer: } \delta_k^o = (d_k - o_k)(1 - o_k), \quad k = 1, 2, \dots, l$$

$$\text{The error of hidden layer: } \delta_j^y$$

$$= \left(\sum_{k=1}^l \delta_k^o W_{jk} \right) (1 - y_j), \quad j = 1, 2, \dots, m$$

- (4) Adjust the weights of synapses between layers of the network:

The weight between the hidden layer in the middle of the network and the output layer at the end of the network is adjusted as follows:

$$W_{jk} = W_{jk} + \eta \delta_k^o y_j, \quad k = 1, 2, \dots, l, \quad j = 0, 1, \dots, m$$

The threshold is adjusted as follows:

$$B_{jk} = B_{jk} + \eta \delta_k^y, \quad k = 1, 2, \dots, l, \quad j = 0, 1, \dots, m$$

The input layer and implicit layer weights at the head end of the network are adjusted as follows:

$$V_{ij} = V_{ij} + \eta \delta_j^y x_j, \quad j = 0, 1, \dots, m, \quad i = 0, 1, \dots, n$$

The threshold is adjusted as follows:

$$A_{ij} = A_{ij} + \eta \delta_j^y, \quad k = 1, 2, \dots, l, \quad j = 0, 1, \dots, m$$

Repeat the above steps to make the BP artificial neural network continuously trained, and the weights and

thresholds of each layer of the neural network are gradually adjusted. End of learning process until the output layer error is below the set error accuracy or the number of learning reaches a maximum.

2.4 Construction of network prediction model

The sensor network collects real-time information of each sensor node and predicts the environmental data at a later time using the BP neural network model, thereby knowing the trend graph of the parameter, and according to the trend graph, the user knows the future status of the parameter in advance.

1. Configure the sensor node and sensor network, set the transmission period and transfer the collected data information to the SQL Server database located in the server every fixed period.
2. In order to establish a BP neural network model, historical data are obtained from SQL Server; that is, a sensor node datum in a fixed time interval is selected, and two samples are formed after normalization to be used for training and testing.
3. The training of the neural network is performed using the training samples obtained from (2), and the trained BP model is obtained.
4. After the model training is completed, record and save this network structure, for example, the weights, thresholds and other important parameters of each node of each layer of the network, and then use the test samples obtained in (2) to detect whether the BP model meets the accuracy requirements.
5. If the prediction model finally obtained in the above steps satisfies the functional requirements of the system, the model is stored for subsequent prediction.

After determining the BP network model, the MATLAB code is compiled into a jar package that is independent of MATLAB using the built-in compiler and JAVA library in MATLAB. The generated jar package file is imported into the project and referenced in the class to apply the model. The overall structure of the BP model is shown in Fig. 5.

3 Relate work

The purpose of this study is to collect the real-time information of the sensor nodes through the sensor network and use the collected data to predict the environmental data in a certain period of time, so as to obtain the trend graph of the parameter, and let the user know the future of the parameter according to the trend graph. The collected data is used to predict the environmental data at a certain time later, thereby obtaining a trend graph of the parameter, and according to the trend graph, the user can know the future

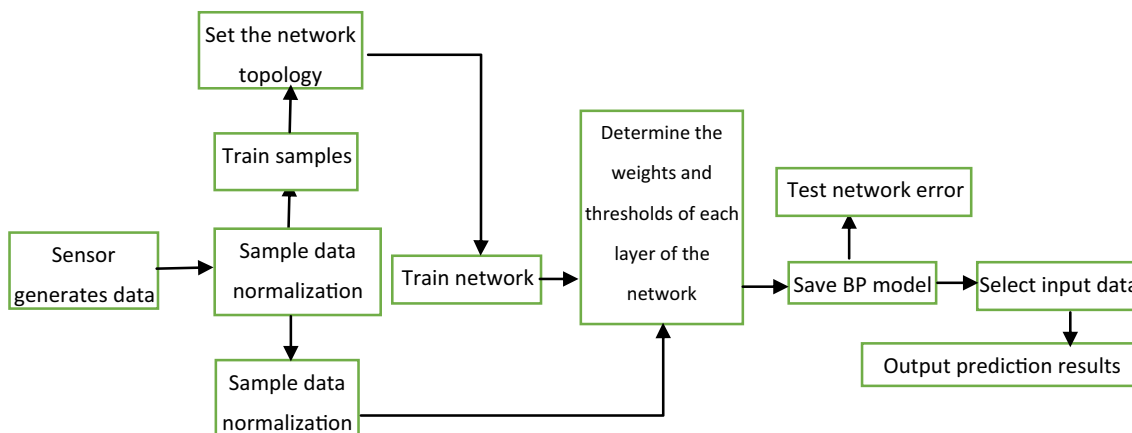


Fig. 5 Design of BP neural network

status of the parameter in advance. The model uses the Windows 7 operating system and MATLAB R2014a, as well as the SQL Server database for simulation experiments. The neural network parameters are set by calling the neural network toolbox in MATLAB. After determining the BP network model, the MATLAB code is compiled into a jar package that is independent of MATLAB using the built-in compiler and JAVA library in MATLAB. The generated jar package file is imported into the project and referenced in the class to apply the model.

1. Determining the number of BP network layers: The more the number of neural network layers, the higher the prediction accuracy of the network, but the more complicated the network structure at the same time, which increase the training time of the weight and threshold of the network, so a three-layer BP network structure is used to meet the requirements in the system model.
2. Training function for setting threshold and weight: Select tradingdm() function as learning function. This function has faster convergence speed and avoids the local minimum problem in the network.
3. Setting the learning rate: The learning rate is very important for establishing a suitable neural network model. Generally, a smaller learning rate is selected, and the learning rate in this model is 0.03.
4. Set the transfer function: the transfer function is usually a nonlinear function, using the logsig() function. $f(x) = \frac{1}{1+e^{-x}}$, which is weaker than the larger input amplification effect. The input amplification effect is enhanced.

4 Experiments

We use Windows 7 operating system, MATLAB R2014a and SQL Server database in this paper to simulate the experiment process, and the neural network parameter

settings are made by calling the neural network toolbox in MATLAB.

1. Sample collection and preparation

The application of this model is to obtain the original sample by collecting data on the temperature sensor nodes in an environment. The software used is MATLAB 2014b, and the operating system uses Windows 2010.

According to the method described above, the training samples and test samples of the network are first selected. Normalize the sample. Then the specific parameters of the model are then determined. In the process of determining the number of neurons in the hidden layer of the model, a new optimization scheme is proposed and demonstrated through experiments.

It is planned to use the data collected in the former 8 times to predict the 9th time. For example, the 9th time is predicted from the 1st to the 8th, and the 10th time is predicted from the 2nd to the 9th. There are a total of 40 samples, of which 36 are training samples and 4 are test network performance samples. The sample is normalized, using the function premmx in MATLAB, the normalized data will be distributed in $[-1, 1]$, and the resulting partial structure is shown in Table 1.

Table 1 Use function premmx to normalize the samples

4	5	6	7	8	9	10
0.5792	0.5833	0.3611	0.2083	0.1389	0.0417	-0.0556
0.5946	0.4459	0.3784	0.2029	0.1014	0.0234	-0.029
0.5396	0.4348	0.2754	0.2838	0.1892	0.1622	0.0108
0.4881	0.3919	0.2973	0.2571	0.2296	0.1714	0-0.0123
0.3415	0.3203	0.2941	0.2703	0.085	0.0511	0
0.3125	0.2927	0.1462	0.2418	0.122	0.0327	0.0588
0.2912	0.1515	0.103	0.1273	0.061	0.111	-0.0732

Table 2 Process of choosing neurons

Number of training	Number of hidden layer nodes	Prediction accuracy
10,000	13	0.0073
10,000	18	0.0067
9678	20	0.0056
8670	21	0.0053
9513	26	0.0041

2. The dichotomy method is used to optimize the number of hidden layer nodes: the maximum number of training sessions is 10,000, and the final target error is less than 0.5%.

- (a) Since the number of input layer nodes is $n = 8$, the number of output layer nodes is $l = 1$. $A = 5$ and $b = 20$ can be obtained by the formula. The first point 12 is determined on the interval [5, 21], and the experimental result when the number of nodes is 12 is recorded; that is, the network error is 0.0073.
- (b) Determine the position of the second test point, point 18 of [11, 21], and record the experimental results; that is, the network error is 0.0067, which still does not meet the demand.
- (c) The position of the test point is determined for the third time. When the interval is [17, 21], that is, when the number of hidden layer nodes is 20, the error is 0.0056, which still does not meet the demand.
- (d) The position of the test point is determined for the third time. When the interval is [19, 21], that is, when the number of hidden layer nodes is 21, the error is 0.0053, which still does not meet the demand.
- (e) The initial interval is expanded to obtain the interval [21, 33], and the above steps are continued for the extended interval. It is found that the optimal solution satisfying the condition is obtained when the number of nodes is 26, and the precision is 0.0041, as shown in Table 2.

From the above, the number of hidden layer nodes satisfying the error can be selected only by 5 calculations, and the method can quickly and accurately determine the number of nodes satisfying the requirements in comparison with the trial and error method to randomly set the number of hidden layer nodes.

3. Training of BP neural network model

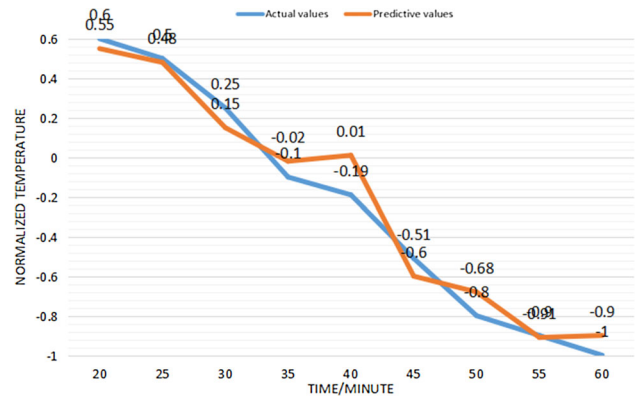


Fig. 6 Actual output predictive output after training in first day

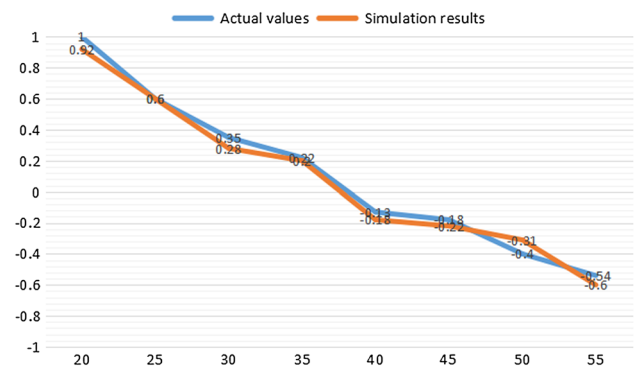


Fig. 7 Actual output and predictive output after training in second day

From the above analysis process, the number of neurons in the hidden layer is 29, the learning rate is 0.03, the maximum number of training times is 5000, the accuracy of the training requires a prediction error of 0.5%, and the results of the first day of training and the second day of training are shown in Figs. 6 and 7. The actual results and simulation results after training are shown in Fig. 6.

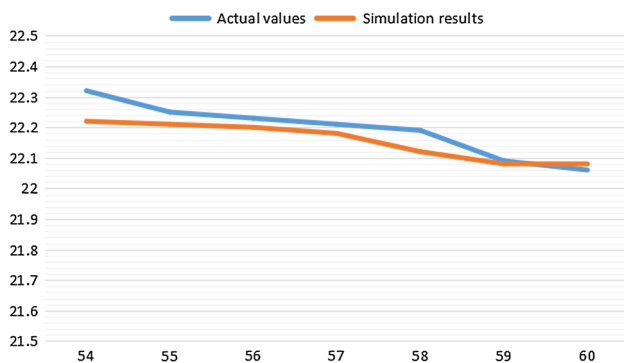
It can be seen that the error between the actual temperature results of the BP neural network model’s temperature data prediction results is not very large, and the predicted temperature data curve can well approximate the actual temperature curve (Fig. 8).

4. Error Analysis

The BP network model and the gray linear regression combined model prediction results are compared with each other. The results are shown in Table 3. A total of four measurements, the first measurement of BP network actual results and prediction results of the error is 0.40%, while the gray linear regression combination model prediction error is 0.36%. The error between the second measurement actual result and the prediction result is 0.13%, and the prediction error of the gray linear regression combination model is 0.27%. The error between the third actual result

Table 3 Error of BP network model and gray linear regression combination model

Measurements	The first time	The second time	The third time	The fourth time
Actual results	22.32	22.23	22.15	22.03
BP network model prediction results	22.23	22.20	22.10	22.06
Error size	0.09	0.03	0.05	0.03
Error percentage	0.004	0.0013	0.0027	0.0018
Gray linear regression combined model prediction result	22.21	22.17	22.08	22.08
Error size	0.11	0.06	0.07	0.05
Error percentage	0.0036	0.0027	0.0032	0.0023

**Fig. 8** Actual output and the output of BP network simulation after test

and the prediction result is 0.27%, and the prediction error of the gray linear regression combination model is 0.32%. The error of the fourth measurement is 0.18%, while the error of the gray linear regression combination model is 0.23%. It can be seen from the temperature prediction results and actual results of the above four moments that compared with the gray linear regression combined model, the overall error of the BP network is smaller, the prediction accuracy is higher, and the prediction requirements are more satisfied.

5 Conclusion

This paper designs a multi-parameter online measurement IoT based on BP neural network algorithm. That is to say, for the forecasting demand, a general scheme based on BP prediction network, environmental temperature prediction model is designed. In the paper, the setting process of some important parameters of the model and the number of hidden layer nodes are introduced in detail, and a new optimization scheme is proposed. This paper simulates and experiments on the BP model. The model is implemented in MATLAB, and finally, the prediction results of the BP model and the gray linear regression combination model are compared. Compared with the gray linear regression

combination model, the BP network has smaller overall error, higher prediction accuracy and higher prediction requirements. By comparison, the model has good predictive performance.

This paper constructs a BP model that can accurately predict temperature and humidity, and optimizes the process of determining the number of hidden nodes in the model. It not only realizes real-time monitoring of environmental data, but also achieves the goal of temperature and humidity warning. The users can monitor and manage the environment more easily through this system. There are many imperfections in this article, it may exhibit nonlinear characteristics when the temperature change in the scene due to different heating characteristics, and the same BP prediction network model cannot be directly applied to predict and warn. The generalization ability of the neural network model can be considered. Just use a similar method to learn and train the neural network and then form a suitable model before forecasting and warning.

Compliance with ethical standards

Conflict of interest No conflict of interests.

References

- Bertino E, Choo KKR, Georgakopoulos D, Nepal S (2016) Internet of things (IoT): smart and secure service delivery. *Acm Trans Internet Technol* 16(4):22
- Bandyopadhyay D, Sen J (2011) Internet of things: applications and challenges in technology and standardization. *Wirel Pers Commun* 58(1):49–69
- Sung WT, Hsu YC (2011) Designing an industrial real-time measurement and monitoring system based on embedded system and ZigBee. *Expert Syst Appl* 38(4):4522–4529
- Han X (2011) Research on the design of the intelligent campus system based on the Internet of Things technology. *Electron Des Eng* 92(1):235–253
- Yang L, Yang SH, Plotnick L (2013) How the Internet of Things technology enhances emergency response operations. *Technol Forecast Soc Change* 80(9):1854–1867

6. Wang YH, Hsu CP, Lai JY (2004) A mobile IPv6 based seamless handoff strategy for heterogeneous wireless networks. In: International conference on computer and information technology. IEEE, pp 600–605
7. Zheng S, Xiong X, Vause J et al (2013) Real-time measurement of wind environment comfort in urban areas by Environmental Internet of Things. *Int J Sustain Dev & World Ecol* 20(3):254–260
8. Lan Y, Zhang B (2011) An internet based interconnection and online technology of embedded device in Internet of Things. *Comput Meas Control* 19(6):1449–1451
9. Cao Y, Li Y, Chen L (2012) Online measurement and remote fault diagnosis for lithium battery based on technologies of internet of things. *J Soochow Univ* 32(2):1–9
10. Variations AI (2013) Predictive analytics by using Bayesian model averaging for large-scale Internet of Things. *Int J Distrib Sens Netw* 2013(3):1–10
11. Ni SH, Bai YH (2000) Application of BP neural network model in groundwater quality evaluation. *Syst Eng Theory Pract* 20(8):124–127
12. Hosseini MP, Pompili D, Elisevich K et al (2017) Optimized deep learning for EEG big data and seizure prediction BCI via Internet of Things. *IEEE Trans Big Data PP*(99):1
13. Zhang P, Liu Y, Wu F et al (2016) Low-overhead and high-precision prediction model for content-based sensor search in the Internet of Things. *IEEE Commun Lett* 20(4):720–723
14. Cui DD, Liu F (2012) The application of BP neural network in Internet of Things. *Adv Eng Forum* 6–7(9):1098–1102
15. Yang X, Sun J, Niu C et al (2016) Application of data fusion technology in fire detection system based on the internet of things. *Electron Meas Technol* 3:100–105
16. Wang F, Niu L (2016) An improved BP neural network in Internet of Things data classification application research. In: Information technology, networking, electronic and automation control conference, IEEE. IEEE, pp 805–808
17. Li J, Cheng JH, Shi JY et al (2012) Brief introduction of back propagation (BP) neural network algorithm and its improvement. *Adv Comput Sci Inf Eng* 169:553–558
18. Ding S, Su C, Yu J (2011) An optimizing BP neural network algorithm based on genetic algorithm. *Artif Intell Rev* 36(2):153–162
19. Zhang YQ, Zhong YY, Zhou YH (2013) Forecast of grain output using a method of nonlinear BP neural network based on linear ARIMA modified. *Math Pract Theory* 43:191–195
20. Mao HL, Gao JW, Chen XJ et al (2014) Demand prediction of the rarely used spare parts Based on the BP neural network. *Appl Mech Mater* 519–520:1513–1519
21. Elliott MS, Rasmussen BP (2013) Decentralized model predictive control of a multi-evaporator air conditioning system. *Control Eng Pract* 21(12):1665–1677