Identification of Typical Fault States of Marine Diesel Engines Based on Optimized BP Neural Network

Ziyang Tong[1], Yi Sun[2], Junwen She[1], Yuanyan Zhu[3], Zhiqiang Zhao[1,4,5,] *

[1.] Maritime College, Guangdong Ocean University, Zhanjiang, 524088, China;

[2.] Dubai Branch, China Classification Society, Dubai, United Arab Emirates;

[3.] School of Literature and Journalism, Guangdong Ocean University, Zhanjiang, 524088, China;

[4.] Southern Marine Science and Engineering Guangdong Laboratory (Zhanjiang), Zhanjiang 524006, China;

[5.] Guangdong Provincial Shipping Intelligence and Safety Engineering Technology Research Center, Zhanjiang 524088, China.

* Corresponding Author Email: zhaozhiqiang929@gdou.edu.cn

Abstract. Ship diesel engine is the core equipment of the ship, and its working condition is directly related to the safety and reliability of ship navigation. Once the ship diesel engine fails, it may cause different degrees of sea damage accidents, bringing economic losses and even endangering the life safety of crew members. The fault diagnosis can monitor the state of diesel engine during the operation of the ship and capture the fault signal to ensure that the fault can be found and eliminated in time. Therefore, the fault diagnosis research of ship diesel engine is an important research direction at present. This paper verifies that BP neural network has disadvantages such as inability to escape from local optimal solution and long convergence time, and the BP neural network optimized by genetic algorithm is based on intelligent fault state recognition. The optimized BP neural network has significantly improved in the fitting performance and classification performance. The research results have certain reference value and provide a basis for the research of intelligent fault diagnosis of marine diesel engines.

Keywords: Diesel engine; Fault diagnosis; BP neural network; Genetic algorithm.

1. Introduction

Ship diesel engine is the heart of most ships and the source of ship power. Ship diesel engines are generally large in size, complex in structure and numerous in parts. Once they break down, they may be shut down for repair, or cause major ship accidents due to short or long time loss of ship power, which concerns the safety and economic efficiency of the crew. The regular maintenance system of ships has certain drawbacks, as regular disassembly and assembly will destroy the frictional state of the parts, and may lead to new failures due to improper assembly. Therefore, efficient and intelligent ship fault diagnosis technology is particularly important[1-2]. Based on the analysis of operation monitoring data, the intelligent operation and maintenance mode, which is based on the condition of ship diesel engine to realize the situation-based maintenance and supplemented by preventive maintenance, can not only reduce the labor and time cost in the traditional three-stage maintenance system, but also improve the safety and reliability of ship diesel engine through relatively accurate fault diagnosis method[3]. It improves the efficiency of power unit maintenance and reduces the maintenance cost of enterprises.

This paper takes marine diesel engine as the research object and combines the BP algorithm optimized by genetic algorithm with fault diagnosis analysis, which can improve the accuracy of fault diagnosis and reduce the cost of fault diagnosis. After determining the BP neural network, it is optimized by genetic algorithm, and simulated and tested after completing the training to obtain the accuracy of fault diagnosis for five operating conditions, which is used to provide a basis for the research of fault diagnosis algorithm for marine diesel engines[4-6].

2. BP Neural Network

2.1 Principle of BP Neural Network

BP neural network is an intelligent information processing system with an algorithm called BP algorithm, which uses a gradient search technique to minimize the mean squared difference of the error between the actual and expected output values of the artificial neural network.

The basic BP algorithm consists of two processes: forward propagation of the signal and backward propagation of the error. In forward propagation, the input signal acts on the output node through the hidden layer and undergoes a nonlinear transformation to produce the output signal. If the actual output does not match the desired output, it is transferred to the back-propagation process of the error. The error back-propagation is to back-propagate the output error through the hidden layer to the input layer one by another, and to apportion the error to all units in each layer, using the error signal obtained from each layer as the basis for adjusting the weights of each unit. The error is decreased along the gradient direction by adjusting the connection strength of the input nodes to the hidden layer nodes, the connection strength of the hidden layer nodes to the output nodes, and the threshold value[7]. After repeated learning training, the training is stopped by determining the weights and thresholds corresponding to the minimum error. At this point, the trained BP neural network can automatically process the input information of similar samples with non-linear transformation to minimize the output error[8,9].

2.2 BP Neural Network Structure

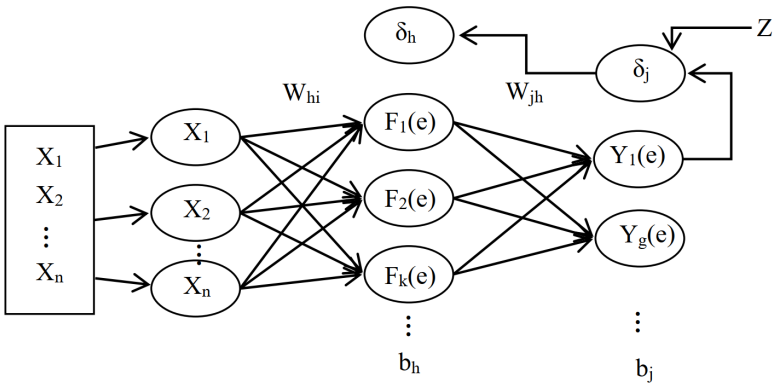


Figure 1. Structure of Neural Network

As shown in Figure 1, the input signal Xn will pass through each neuron in the input layer directly to each neuron in the hidden layer, and the neurons in the same layer are not connected to each other, only the inter-layer neurons are connected. The connection is accompanied by its associated weights Whi and Wjh. When the input signal to each neuron is empty, the neuron's own bias bh, bj will keep its neurons in the still-activable state, which is the structural feature of the BP neural network. This is the structural feature of the BP neural network. The hidden layer is a collection of neurons that process the input signal linearly or nonlinearly, and the more the number of hidden layers, the more effective the network training will be. When dealing with more complex mathematical models, the number of neurons in each hidden layer should be within a suitable range in order to achieve a reasonable balance between accuracy and learning efficiency.

Input layer to hidden layer of the forward network:

Compute the input-weighted summation of the hidden layer neuron h

$$In_h = \sum_{i=1}^{n} (X_i \times W_{hi}) + b_h \tag{1}$$

Calculate the output of the hidden layer neuron h, which is necessary to pass its transfer function Fh(e)

$$Out_h = F_h(In_h) \tag{2}$$

Hidden layer to output layer:

Compute the weighted summation of the output layer neurons

$$In_j = \sum_{h=1}^{k} (Out_h \times W_{jh}) + b_j \tag{3}$$

Calculate the output of the hidden layer neuron h, which is necessary to pass its transfer function Fj(e)

$$Out_j = Y_j(In_j) \tag{4}$$

Back propagation of the network

Update the weights Wjh

$$E_{Out_j} = \frac{1}{2}(expected_{Out_j} - Out_j)^2 \tag{5}$$

$$E_{total_j} = \sum_{j=1}^{g} E_{Out_j} \tag{6}$$

According to the chain rule

$$\frac{\partial E_{total_j}}{\partial W_{jh}} = \frac{\partial E_{total_j}}{\partial Out_j} \cdot \frac{\partial Out_j}{\partial In_j} \cdot \frac{\partial In_j}{\partial W_{jh}} \tag{7}$$

$$W_{jh}^{+} = W_{jh} - \eta \frac{\partial E_{total_j}}{\partial W_{jh}} \tag{8}$$

Similarly, update the weights bj

$$b_j^{+} = b_j - \eta \frac{\partial E_{total_j}}{\partial b_j} \tag{9}$$

Update the weights Whi

$$E_{Out_h} = \frac{1}{2}(expected_{Out_h} - Out_h)^2 \tag{10}$$

$$E_{total_h} = \sum_{j=1}^{g} E_{Out_h} \tag{11}$$

According to the chain rule

$$\frac{\partial E_{total_h}}{\partial W_{hi}} = \frac{\partial E_{total_h}}{\partial Out_h} \cdot \frac{\partial Out_h}{\partial In_h} \cdot \frac{\partial In_h}{\partial W_{hi}} \tag{12}$$

$$W_{hi}^{+} = W_{hi} - \eta \frac{\partial E_{total_h}}{\partial W_{hi}} \tag{13}$$

Similarly update the weights bh

$$b_h^{+} = b_h - \eta \frac{\partial E_{total_h}}{\partial b_h} \tag{14}$$

After the output signal has passed through the hidden layer, it will be used as the input signal of the next layer, namely the output layer (or the next hidden layer), and finally processed by each neuron and output as the predicted value. When the actual value of the output has an error δ with the predicted value, the derivative of the error value of each neuron is calculated by back-propagating the loss function by the original forward propagation channel called the gradient. The next gradient is calculated using the previous gradient, and this process is repeated continuously to obtain each full-time gradient. Each neuron weight is different from its gradient value for convergent modification. This is closer to the local minimum, in order to achieve a BP neural network continuously approaching the optimal state and finally reaching a suitable performance range[10].

3. BP Neural Network-based Ship Diesel Engine Typical Fault State Recognition

3.1 Inputs, Outputs and Samples

The reduction of training-irrelevant parameters can not only effectively reduce the training time, but also improve the accuracy of the neural network. Through attribute reduction, 12 feature parameters were selected from 17 parameters and 150 sets of samples of the source data, namely: power, maximum burst pressure, compressor flow, compressor outlet temperature, post-intercooler temperature, intercooler temperature difference, sweep gas temperature, sweep gas pressure, exhaust port temperature, exhaust gas inlet turbine temperature, and exhaust gas outlet turbine temperature[11]. These parameters have direct or indirect correlations to the five typical failure states of classified marine diesel engines: normal operating conditions, reduced cooler efficiency, reduced pressurizer efficiency, extended combustion duration, and reduced fuel injection.

The distribution of the parameter performance values in various aspects varies greatly from one diesel engine series to another. By comparing the performance of the BP neural network when the data is normalized to the range of 0~1, -1~1, and no normalization, the final decision was made to process the data normalization to the range of 0~1, which reduces the network gradient descent time and improves the accuracy of the network at the same time.

3.2 Network Structure and Determination of Hyperparameters

In the construction and training of BP neural network, the initial weight threshold, learning function, and performance function all affect the performance of the network to different degrees. Using the system randomly generated weight thresholds as the initial weight thresholds in the previous stage will cause the network instability and lead to different updates of the weight thresholds in the network training[12,13].

Based on empirical formulas $m = \sqrt{n+l} + \alpha$ (m is the number of neurons in the hidden layer, n is the number of neurons in the input layer, l is the number of neurons in the output layer, and α is a constant from 1 to 10) and through continuous comparison runs, a 12-5-1 network is finally used as the framework. Among them, this program will establish the natural numbers from 1 to 5 (including 1 and 5) as the outputs. 1 to 5 outputs represent the five cases of normal operating conditions, reduced intercooler efficiency, reduced compressor efficiency, extended combustion duration, and reduced fuel injection, respectively.

(1) Selection of Learning Rate

The learning rate affects the learning time of the whole network, which is mainly reflected in the decreasing rate of the loss function. The larger the learning rate, the larger the weight change each time, and the faster the convergence rate, when the network with the appropriate learning rate will also improve its learning performance and the accuracy of predictive classification recognition. Too large a learning rate will cause the neural network to oscillate, and too small a learning rate will easily cause the network to be trapped in a local optimal solution and the training learning rate will be too slow, which will seriously affect the performance of the network. Through repeated debugging, the learning rate of 0.003 was determined to be relatively appropriate[14].

(2) The Choice of Activation Function

The choice of activation function needs to be tried and compared by various combinations of activation functions. According to the processed data set and its output value, as the input data set is processed in the range of 0~1, avoiding too large positive or negative numbers, resulting in serious gradient dispersion, the tansig function can be used as the activation function for the hidden layer neurons with other layers[15]. Furthermore, since the final output set is the set of natural numbers from 1 to 5, when the tansig function finishes processing the data in the range of 0 to 1, it is processed by the final selection of the purelin function of the output layer blunt linear.

(3) Selection of the Number of Learning Rounds Epoch

The appropriate choice of the number of learning rounds can prevent the network from entering a dead loop. When it does not converge or converges too slowly, if there is no limit on the number of learning rounds, the network will keep training without stopping, and the training in this case does not achieve the desired results. During the debugging process, we first set a larger number of learning rounds, and then find the number of training sessions with a small rate of accuracy change to ensure that the vast majority of training sessions do not exceed this value under the required network training performance[16]. Finally, 200 times was set as the number of learning rounds for network training to avoid network training overfitting.

After selecting reasonable hyperparameters, the trained BP neural network is tested. If the desired effect is achieved, the weight threshold of each layer of the network is output and the trained network is saved for subsequent calls.

3.3 Network Testing and Results Analysis

Among the large amount of experimental data, the following two typical cases were selected for analysis.

(1) Typical Case I

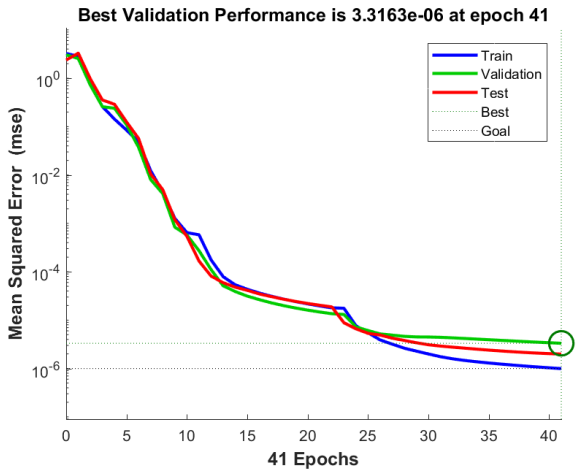


Figure 2. Performance Result of Case I

As shown in Figure 2, the trend of Train, Validation and Test of these three curves with the increase of training times shows relatively good training effect. The three lines maintain a certain relationship, and the error decline rate in the early stage is larger, which is convenient to save training time. The slope of the middle period gradually becomes smaller until the convergence of the BP neural network training, and the training will stop when the set training error is reached. It can be clearly seen that the set error requirement of 1×10-6 is reached when the number of training epochs reaches more than 40.

Table 1. Results of BP Neural Network Simulation Tests

Dataset	Fault Recognition Rate									
1~10	99.90%	99.90%	99.90%	99.90%	99.93%	99.84%	99. 978%	99.88%	99.99%	99.95%
11~20	97.54%	99.90%	99.71%	99.81%	99.84%	99.67%	99.66%	99 .978%	99.94%	99.83%
21~30	99.88%	99.91%	99.98%	99.93%	99.96%	99.81%	99.99%	100.00%	100.00%	100.00%

Table 1 shows the results of the test data sets from 1 to 30 by this well-trained BP neural network simulation test, which belong to the five cases of normal operating conditions, reduced intercooler efficiency, reduced compressor efficiency, extended combustion duration, and reduced fuel injection. The recognition rate is satisfactory, and the data show that it is possible to use this BP neural network for the identification and prediction of marine diesel engine fault states, i.e., to save the weights and thresholds after the training is completed and keep them for further use.

Typical Case II

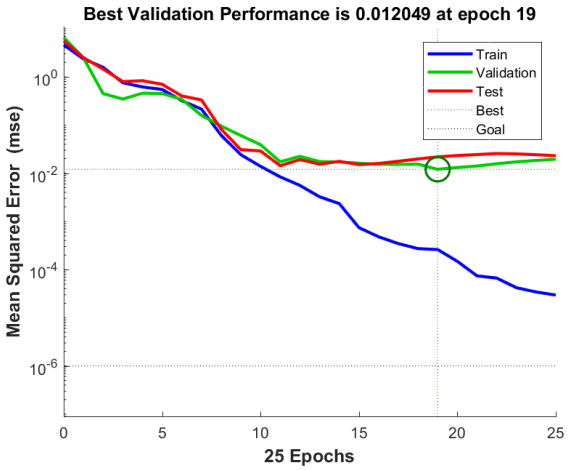


Figure 3. Performance Result of Case II

From Figure 3, we can see that the Train line is separated from the Validation and Test lines, and the three lines do not have a certain relationship slowly decreasing, and in the case of not reaching the error set by the training, the mean squared difference line of Validation and Test reaches the minimum mean squared difference of 0.012049 in 19 epochs. The subsequent 6 epochs have an increasing trend of error. In order to avoid the phenomenon of overfitting, the network finally ended the training at 25 epochs. The effect of case 2 is not as ideal as that of case I. The reason for this is that when the BP network does not set the initial weight threshold artificially, it takes random values according to certain algorithm rules, which leads to the instability of the BP neural network. However, because of such randomness, the BP neural network has a high generalization ability.

3.4 Genetic Algorithm to Optimize BP Neural Network

(1) The Basic Idea of Genetic Algorithm to Optimize Neural Network

Genetic algorithm is a stochastic search algorithm for the optimization of highly nonlinear objective functions without analytic expressions. Through selection, crossover and mutation operations, the genetic algorithm selects individuals from the initial population that are more adapted to the environment, so that the population evolves to a better region in the search space, evolves iteratively, and finally obtains the most adapted individuals to find the optimal solution to the problem.

(2) Genetic Algorithm Independent Variable Dimensionality Reduction

The individuals are coded to produce the initial population. The fitness function is used to measure the degree of superiority of each individual to the optimal solution, and the higher the value of the fitness function, the higher the probability that an individual will be inherited into the next epoch. The selection operation uses the proportional selection operator, and the probability of inheriting an individual from the parent to the offspring population is proportional to the fitness function value of the individual. The crossover operation uses a single-point crossover operator, and the variation operation uses a single-point variation operator, which randomly generates a variation point and changes the gene value on its corresponding locus.

(3) Genetic Algorithm to Optimize BP Neural Network Weights and Thresholds

When downscaling the input independent variables of the BP neural network, it is necessary to build the BP neural network and use the reciprocal of the sum of squared errors of its output values as the fitness function. In order to avoid the influence of the randomness of the initial weights and thresholds on the value of the fitness function, a genetic algorithm is introduced to optimize the initial weights and thresholds of the established BP neural network when calculating the fitness for each individual.

(4) Design of Control Parameters

The different selection of control parameters in the optimization of genetic algorithms has a non-negligible effect on its performance impact. A relatively ideal optimization performance can be obtained only when the individual parameters are reasonably matched. In the operation of a genetic

algorithm, the maximum number of evolution determines the number of evolutionary epochs of the algorithm. As a rule of thumb, it is generally taken as 20~50 for the operation. The crossover probability PX and variation probability PM control the application probability of crossover operator and variation operator respectively in the genetic algorithm, and PX=0.7 and PM=0.01 are chosen in this example.

(5) Test and Result Analysis of Genetic Algorithm Optimized BP Network

The simulation error results obtained using random weight thresholds are compared with those obtained using optimized weight thresholds, and the advantages and disadvantages of each simulation error result for the tested samples and the training samples are analyzed.

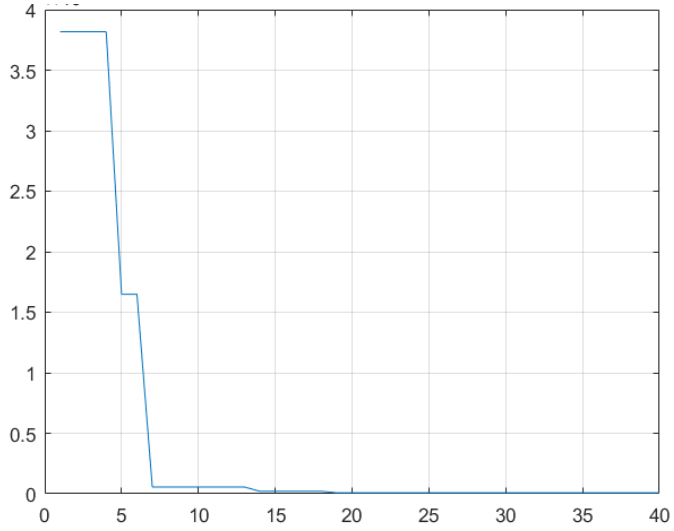


Figure 4. Genetic Algorithm Optimization P-neural Network Error Variation

As shown in Figure 4, after using the genetic algorithm to optimize the BP algorithm, the error decreases continuously from epoch 0, and after epoch 7 the error has converged to 0. Compared with the BP neural network in typical case I, the genetic algorithm effectively optimizes the BP neural network.

Table 2. Simulation Error Results of Random Weights and Thresholds, As Well As Optimized Weights and Thresholds

	Random Weights and Thresholds	Optimized Weights and Thresholds
Test Sample Simulation Error	0.41311	0.0026933
Training Sample Simulation Error	1.9244	0.0058401

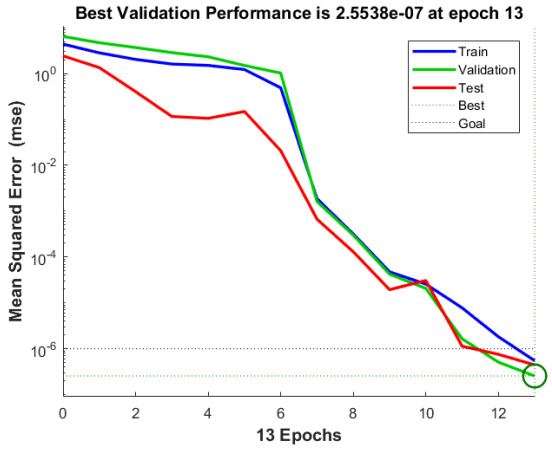


Figure 5. Error Results of BP Neural Network Optimized by Genetic Algorithm

Table 2 shows the simulation error results for the random weights and thresholds and the optimized weights and thresholds, where each simulation error result is presented in the form of a parametric

number. Obviously, each simulation error for both the test and training samples with the optimized weights and thresholds by the genetic algorithm is much better than the error results when the genetic algorithm is not used for optimization. The optimized BP neural network can quickly find the global optimal solution with enhanced prediction capability.

Figure 5 shows the trend of the three curves of Test, Train, and Validation after multiple epochs of genetic evolution after the initial weights and thresholds of the BP neural network are optimized by the genetic algorithm, which is similar to the above case and is obviously typical of case 1, a more ideal BP neural network. And all three lines reached the error target of 1×10-6.

4. Conclusion

In this paper, we propose a genetic algorithm-based BP neural network diagnosis technology for ship diesel engine fault diagnosis, which can improve the economy of the ship and protect the life safety of the crew. The two complement each other to realize the diagnosis of the fault state of ship diesel engine with high accuracy and efficiency.

In this paper, the initial weight threshold of the BP algorithm is optimized using genetic algorithm, and the simulation experiments demonstrate that the results of fault state identification based on genetic algorithm are in good agreement with the measured values. The simulation errors of both the test and training samples optimized by the genetic algorithm are much better than those without the genetic algorithm. In addition, when encountering a more complex network structure, the structural parameters of its BP neural network can be further optimized, and it can also be combined with other neural networks or even deep learning for further optimization.

The marine diesel engine fault diagnosis technique used in this paper can effectively diagnose marine diesel engine faults through simulation experiments, and can provide reference for marine diesel engine fault diagnosis research.

Acknowledgements

Fund Projects: The National Natural Science Foundation of China (NSFC) Funded Project (52071090); Guangdong Province General University Special Fund for Key Areas Funded Project (2020ZDZX3063); Non-funded Science and Technology Research Projects of Zhanjiang (2021B01075); 2021 Guangdong Ocean University Undergraduate Innovation Team (CXTD2021021); College Students' Innovation and Entrepreneurship Training Program Project (CXXL2022198).

References

[1] Zhang, Y. Fault Diagnosis of Naval Diesel Engine of Support Vector Machine [J]. Ship Science and Technology, 2019, 41(22):106-108.

[2] Yuan, J. Fault Diagnosis of Naval Diesel Engine Based on Vibration Signal[J]. Ship Science and Technology, 2018, 40(08):61-63.

[3] Zhao, Z.; Wei, B.; Wang, Y.; Huang, X.; Li, B.; Lin, F.; Ma, L.; et al. An Array of Flag-Type Triboelectric Nanogenerators for Harvesting Wind Energy. Nanomaterials, 2022, 12, 721: 1-11.

[4] JIA B.; JIA Z.; YU P.; et al. Data-driven vessel smart fault diagnosis method[J]. Control Engineering of China, 2019, 26(10): 1892-1898.

[5] Feng, D.H. Zhou, W. Decomposition of Ship Engine Fault Signal Based on Hilbert Spectrum Extraction[J]. Bulletin of Science and Technology, 2014, 30(10):214-216.

[6] HE P.; SUN Y.; ZHAO Z. Optimization of structural parameters for SCR of 6105AZLD marine diesel engine[J]. Internal Combustion Engines, 2019(6): 25-29.

[7] Zhou, M. Lv, Z.G. Di, R.H. Li, Y. BP Neural Network Modeling Based on Small Sample Data[J]. Science Technology and Engineering, 2022, 22(07): 2754-2760.

[8] Chen, T. Research on the Fault Diagnosis of Marine Diesel Engine Based on Neural Network[D]. Wuhan University of Technology, 2014.

[9] Ke, C. Gan, Y. Wang, J. Zhu, R.J. Chen, W. PEMFC System Modeling Based on BPNN Identification Model[J]. Software Engineering, 2020,23(03):14-16.

[10] Xu, L.J.. Neural Network-based Fault Diagnosis Technology for Marine Diesel Engines[D]. Jiangsu University of Science and Technology, 2013.

[11] Zhao, Z.Q. Zhang, J.R. Luo, C.H. Jia, B.Z. An. L.T. TyAVL BOOST-based typical fault simulation and data analysis of marine diesel engine[J]. Chinese Journal of Ship Research, 2021, 16(02): 176-181+193.

[12] Liao Z, Song X, Jia B, et al. Automatic Bearing Fault Feature Extraction Method via PFDIC and DBAS[J]. Mathematical Problems in Engineering, 2021, 6655081: 1-13.

[13] ZHONG G.; JIA B.; XIAO F.; et al. Intelligent fault diagnosis of marine diesel engine based on deep belief network[J]. Chinese Journal of Ship Research, 2020, 15(3): 136-142+184.

[14] Ke, Y. Song, E.Z. Yao, C. Dong, Q. A review:ship diesel engine prognostics and health management technology[J]. Journal of Harbin Engineering University, 2020, 41(01):125-131.

[15] Cao, K. Tan, C. Liu, H. Zheng, M. Data fusion algorithm of wireless sensor network based on BP neural network optimized by improved grey wolf optimizer[J]. Journal of the University of Chinese Academy of Sciences, 2022, 39(02):232-239.

[16] Zhao, Z.Q. Yan, Y.S. Huang, L.Z. et al. Nonlinear compensation of ship wing-sail slewing speed[J]. Journal of Harbin Engineering University, 2015, 36(10):1346-1350.