Genetic algorithm to improve Back Propagation Neural Network ship track prediction

Xingqiang Chen*, Xin Meng and Yue Zhao
Maritime College, Tianjin University Of Technology, Tianjin, China

*Corresponding author: chenxingqiang22@163.com

Abstract. This paper studies a method that uses genetic algorithm to simultaneously optimize the number and weights of Back Propagation (BP) neural network neurons to predict the ship's trajectory, so as to accurately predict the ship's trajectory. The trajectory of the ship can be predicted by the neural network, but the selection of the setting parameters of the neural network requires rich experience and a large amount of attempts. In this study, the genetic algorithm (GA) is used to optimize the structure and weights of the neural network at the same time, avoiding the manual setting of parameters and improving the accuracy of the neural network prediction.

Keywords: bp, neural network, Genetic algorithm, ship track prediction

1. Introduction

1.1. Research background
With the deepening of economic globalization, the deepening of economic integration, the continuous development of reform and opening up, China's foreign trade volume has steadily increased. Shipping plays an important role in the international cargo transportation. 2/3 of the world's international trade and 90% of China's international trade goods are transported by ship. With the vigorous development of the ship transportation industry, the problems in ship navigation are constantly exposed. Traditional radar and Very High Frequency (VHF) equipment are not enough to ensure the safety of ship navigation. Automatic Identification System (AIS) has been widely developed as a device that can automatically send and receive information about own ship and other ships. AIS has been recognized by the international transportation industry, the International Maritime Organization stipulates that AIS equipment must be installed. Some ships collide in the sea due to the untimely or unreasonable avoidance. The equipment based on the principle of automatic transmission and reception provides the ship with information about other ships, and provides convenience for ship avoidance, maritime rescue, maritime communication, and marine driving environment. The ship's pilot lacks a system or software for predicting the trajectory of other ships.

1.2. Research meaning
In order to ensure the safety of ships, it is an effective way to predict the trajectory of ships. The prediction of the ship's trajectory is not only forward-looking, but also supplements the sudden loss of AIS information, avoiding the lack of information. Avoiding ship collision ensures the safety of crew and ship.
When the ship is sailing in the sea, the trajectory of other ships or our ships in a certain period of time in the future is very important for our ship's operation at this time. The predicted trajectory can be used to control the ship in advance, rather than to control the ship after almost collision. When the ship meets at sea, the pilot should navigate carefully and take avoidance measures. The predicted ship trajectory can assist the pilot to take avoidance measures.

1.3. Research status
In reference [1], the ship's track is predicted based on AIS by Kalman filtering algorithm; In reference [2], the improved Kalman filtering algorithm is used to predict the ship's track of the restricted river; in reference [3], the ship's track is predicted by improved grey prediction model; In reference [4], the author uses the Gaussian process regression method to predict the ship's track; In reference [5], the author uses BP neural network to predict the ship position difference, so as to indirectly predict the ship track. Reference [1] found that using Kalman filtering algorithm can improve the accuracy of ship position prediction; Reference [2] proposes a Kalman filter with real-time estimation of system noise, the simulation shows higher accuracy than traditional Kalman filter; Reference [3] uses an improved grey prediction model to predict the ship's trajectory, and simulations show that it has better accuracy than the traditional grey prediction model. Reference [4] based on Gaussian process regression for ship trajectory prediction, which predicts the ship trajectory in the next 24 minutes and finds that the error in the first 9 minutes is low and the final error is 8.9%, which meets the prediction requirements. Reference [5] uses BP neural network to predict the ship's position error and indirectly predict the ship's trajectory. By predicting the ship's position in the next 20 minutes, it finds that it has higher accuracy than the traditional Mercator algorithm. However, the prediction accuracy of the above methods is a little low.

1.4. Highlights of this article
In this paper, genetic algorithm is used to optimize the weight and threshold of BP neural network, optimize the number of hidden layer nodes of BP neural network, improve the prediction accuracy of BP neural network, and predict the trajectory of the ship well.

2. BP neural network model
2.1. The history of Back Propagation neural networks
In the history of the development of artificial neural network, perceptron artificial neural network has played a very important role, and it is also regarded as a real available artificial neural network model. The single-layer perceptron of neural network brings about the first wave of neural network development, but the single-layer perceptron is simple in structure and can not deal with nonlinear problems. Research scholars put forward the idea of multi-layer perceptron, but there is no algorithm matching with multi-layer perceptron. The development of neural network encountered a cold winter. Until the 1980s, some scholars proposed an error back propagation algorithm, also known Back Propagation (BP) algorithm. The Back (BP) algorithm can train the weights and thresholds of the neural network, and the scholar gives a mathematical model and demonstrates it. BP neural network is a multilayer perceptron using BP algorithm, which includes input layer, hidden layer and output layer.

2.2. BP algorithm forward derivation
In forward propagation, from the hidden layer to the output layer:

\[ o_k = f(\text{net}_k) \quad k = 1, 2, \ldots, l \] (1)

\[ \text{net}_k = \sum_{j=0}^{m} \omega_{jk}y_j \quad k = 1, 2, \ldots, l \] (2)

When j is equal to 0 in (1) and (2), the product of \( \omega_{0k}y_0 \) represents the threshold of the k-th node in the output layer.

From input to hidden layer:
When $i$ is equal to 0 in (3) and (4), the product of $v_0jx_0$ represents the threshold value of the $j$-th node in the hidden layer.

The transfer function or activation function is a sigmoid function:

$$f(x) = \frac{1}{1+e^{-x}}$$

Figure 1. Schematic diagram of artificial neurons

Figure 2. Schematic diagram of BP neural network

2.3. Back propagation stage of BP algorithm

Error between actual output and expected output:

$$E = \frac{1}{2} (d - O) = \frac{1}{2} \sum_{k=1}^l (d_k - o_k)^2$$

Expand the above formula:

$$E = \frac{1}{2} \sum_{k=1}^l [d_k - f(\text{net}_k)]^2$$

$$E = \frac{1}{2} \sum_{k=1}^l [d_k - f(\sum_{j=0}^m \omega_{jk}f(\text{net}_j))]^2$$

Expand the above formula further:

$$E = \frac{1}{2} \sum_{k=1}^l [d_k - f(\sum_{j=0}^m \omega_{jk}f(\sum_{i=0}^n v_{ij}x_i))]^2$$

$$E = \frac{1}{2} \sum_{k=1}^l [d_k - f(\sum_{j=0}^m \omega_{jk}f(\sum_{i=0}^n v_{ij}x_i))]^2$$
From (8), we can know that the error of the neural network is related to the weights $v_{ij}$ and $\omega_{jk}$. Proper adjustment of the weights can reduce the error of the neural network. We know that the value to be reduced should be adjusted in the direction of gradient decrease, it's the following:

$$\Delta \omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} \quad j = 0,1,2,\cdots,m; \quad k = 1,2,\cdots,l$$

$$\Delta v_{ij} = -\eta \frac{\partial E}{\partial v_{ij}} \quad i = 0,1,2,\cdots,n; \quad j = 1,2,\cdots,m$$

Correction of weight between output layer and hidden layer:

$$\Delta \omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} = -\eta \frac{\partial E}{\partial \text{net}_k} \frac{\partial \text{net}_k}{\partial \omega_{jk}}$$

Correction of weight between hidden layer and input layer:

$$\Delta v_{ij} = -\eta \frac{\partial E}{\partial v_{ij}} = -\eta \frac{\partial E}{\partial \text{net}_j} \frac{\partial \text{net}_j}{\partial v_{ij}}$$

Define two variables:

$$\delta^o_k = -\frac{\partial E}{\partial \text{net}_k}$$

$$\delta^y_j = -\frac{\partial E}{\partial \text{net}_j}$$

So the formula above is written as follows:

$$\Delta \omega_{jk} = \eta \delta^o_k y_j$$

$$\Delta v_{ij} = \eta \delta^y_j x_i$$

So $\delta^o_k$ and $\delta^y_j$ are very important for weight adjustment. Expand formulas $\delta^o_k$, $\delta^y_j$ respectively:

$$\delta^o_k = -(d_k - o_k)$$

$$\delta^y_j = \sum_{k=1}^{l}(d_k - o_k)f'(\text{net}_k)\omega_{jk}$$

Derivation of (6) and (7): 

$$\frac{\partial E}{\partial o_k} = -(d_k - o_k)$$

$$\frac{\partial E}{\partial y_j} = -\sum_{k=1}^{l}(d_k - o_k)f'(\text{net}_k)\omega_{jk}$$

Derivative of (5): 

$$f'(x) = f(x)[1 - f(x)]$$

Expand (17) and (18) according to (19), (20) and (21):

$$\delta^o_k = (d_k - o_k) o_k(1 - o_k)$$

$$\delta^y_j = \sum_{k=1}^{l}(d_k - o_k)f'(\text{net}_k)\omega_{jk} f'(\text{net}_j)$$

$$= (\sum_{k=1}^{l} \delta^o_k \omega_{jk}) y_j (1 - y_j)$$
According to (11), (12), (22) and (23), get the weight error adjustment formula as follows:

\[
\begin{align*}
\Delta \omega_{jk} &= \eta \delta_k^p y_j = \eta(d_k - o_k) a_k (1 - a_k) y_j \\
\Delta v_{ij} &= \eta \delta_{l_k}^p x_i = \eta(\sum_{k=1}^{N} \delta_{k}^p \omega_{jk}) y_j (1 - y_j) x_i
\end{align*}
\]  

(24)

2.4. Genetic Algorithm to Improve BP Neural Network

2.4.1. Chromosome coding. Because of the large number of nodes and weights, if binary coding is used, the chromosome length will increase dramatically, which will bring serious pressure to the calculation and computer. Therefore, in order to reduce the calculation pressure caused by the length of chromosome and reduce the number of iterations, all chromosomes in this paper adopt real coding, because real coding has less calculation than binary coding, is easy to realize, has lower requirements on computer performance, and can achieve the purpose of Optimizing BP neural network. In this paper, the length of all chromosomes is the same, and the corresponding position of each chromosome has the same meaning, so it can be crossed. The coding method proposed in this paper solves the problem that chromosomes of different lengths cannot cross. Table 1 shows how the chromosome is divided into several fragments and the coding method and value range corresponding to each fragment. The chromosomes were divided into 8 segments: A, B, C, D, E, F, G, H. The number of neural network nodes is in the interval [1, 16], and the weights and thresholds are in the interval [-2, 2].

2.4.2. Fitness function. The fitness function in this paper is MSE function of neural network, and its formula is as follows:

\[
fitness_i = MSE = \sum_{n=1}^{N} \sum_{k=1}^{i} (a_k^n - d_k^n)^2, \quad (i = 1, 2, \cdots, \text{population size})
\]

(25)

In batch mode, “n” represents the number of data groups and “k” represents the output of the k-th neuron. The fitness of the i-th chromosome. The value of “population size” is 10, as in the Table 2.

| Chromosome fragment | Corresponding neural network parameters | Coding requirements | Range of gene values in fragments |
|---------------------|----------------------------------------|---------------------|----------------------------------|
| A                   | Actual number of neurons in hidden layer | Real coding         | [1, 16]                          |
| B                   | Effective weight of hidden layer       | Real coding         | [-2, 2]                          |
| C                   | Invalid weight fragment for hidden layer | Real coding         | [-2, 2]                          |
| D                   | Effective weight of output layer       | Real coding         | [-2, 2]                          |
| E                   | Invalid weight of output layer         | Real coding         | [-2, 2]                          |
| F                   | Hidden layer effective threshold       | Real coding         | [-2, 2]                          |
| G                   | Invalid threshold for hidden layer     | Real coding         | [-2, 2]                          |
| H                   | Output layer threshold                 | Real coding         | [-2, 2]                          |

2.4.3. Selection. The selection step is an important step of genetic algorithm, which plays an important role in the genetic algorithm. The selection of better chromosomes and the elimination of worse ones are directly related to the effect of genetic algorithm. The selection step in this paper adopts the roulette model. Assuming that the fitness value of k chromosome is fitness_k.

| Population size | Number of iterations | Cross rate | Mutation rate |
|-----------------|----------------------|------------|--------------|
| 10              | 30                   | 0.2        | 0.1          |

\[
p_i = \frac{\text{fitness}_i}{\sum_{i=1}^{\text{pop.size}} \text{fitness}_i}
\]

(26)
The $p_i$ is the selection probability of the $i$-th chromosome. The `pop_size` is Population size in Table 2, so the value of `pop_size` is 10.

In Table 2, the Population size represents the total number of chromosomes, the Number of iterations represents the total number of iterations, the Cross rate represents the probability of chromosomes crossing, and the Mutation rate represents the probability of chromosome mutation.

2.4.4. Crossover. Crossover occupies an important position in the genetic algorithm, its main role is to generate new chromosomes, and the crossover rate has a great influence on the speed of evolution of the genetic algorithm. The method is based on a certain point as the center, crossing two chromosomes with each other, the choice of this point is random.

2.4.5. Mutations. Mutation is the only source of new genes in genetic algorithms and plays an important role in the evolution of populations. Usually the selected chromosome changes one gene or multiple genes. In this paper, multiple gene mutations are used, and the positions of the mutations are random. The probability of mutation in this article is 0.1, as in Table 2.

Fig.3 shows the flow chart of Genetic Algorithm Optimizing BP neural network. Firstly, according to the number of input nodes, the size of training set and the number of output nodes, the maximum number of hidden nodes is estimated. The maximum number of nodes is 16. Secondly, the hidden layer nodes of the neural network are coded (decimal coding, the result must be an integer greater than 0 and less than or equal to the maximum number of nodes). Thirdly, the BP neural network is initialized to get the weight threshold of the neural network. Fourthly, the weight threshold is coded in decimal (the weight value and the threshold must be between $[-2, +2]$). Then the fitness value of chromosome was calculated. Then, genetic operations such as selection, crossover and variation are carried out to calculate the fitness value and get a new generation of population. When the number of iterations is met, the loop stops, otherwise it returns to perform genetic operation until the condition is met. Then the optimal number of nodes and the weight threshold of the neural network are obtained by decoding. The neural network is constructed according to the number of nodes, and the threshold weight is given to the neural network. The neural network is trained and tested, and the simulation results are obtained.

![Flow chart of genetic algorithm optimization neural network](image-url)
3. Simulation and result

3.1. Experimental environment and data

3.1.1. Experimental environment. This paper uses MATLAB software to simulate, the software version is MATLAB R2014a, and the computer configuration is (CPU: Intel (R) core (TM) i3-2301m 2.10GHz; RAM: 2.0GB).

3.1.2. Vessel data. First look for ship data resources. The original data comes from the website of China port network. Then the data is preprocessed, and the data that meets the requirements is made into data set, which is divided into training set, test set and verification set. Then, BP neural network is constructed, and then genetic algorithm is used to optimize BP neural network. BP neural network is trained to input test set into neural network to get output. The output of neural network and test set are compared to get mean square error results and ship prediction trajectory.

3.2. Simulation result

Fig.4 is a comparison between the ship trajectory predicted by the two methods and the actual ship trajectory. In Fig.4, the horizontal axis is longitude (degrees) and the vertical axis is latitude (degrees). The trajectory in the upper right corner of the time sequence is earlier, that is, the ship is from the upper right corner moves to the lower left corner. Observing the graph, the predicted trajectory gradually shifts from the actual trajectory over time, but the genetic algorithm optimized neural network path is closer to the actual path, and the unoptimized neural network has a larger deviation from the actual path. Whether it is at the upper right corner of the starting position or the lower left corner of the ending position, it is better than the prediction path of the BP neural network.

Fig.5 is a time-based diagram describing the deviation between the predicted trajectory and the actual trajectory. Abscissa is the time (minutes) corresponding to the test set, and ordinate is the mean square error of latitude and longitude.

It can be seen from Fig.5 that, with the increase of time, the mean square deviation of GABP and BP are increasing.

This is because with the ship's navigation on the sea, the marine environment such as wind speed, ocean current and sea wave has changed, the ship's motion model has changed gradually, while the
GABP and BP neural network model have not changed, so the prediction error has increased gradually. But the error of the improved neural network is smaller than that of the unoptimized neural network.

**Figure 5.** Mean square error of ship predicted trajectory

4. Conclusion
Genetic algorithm optimizes the weight and threshold of BP neural network and the number of hidden layer nodes. The simulation results show that the GA optimized neural network GABP has better prediction effect than BP neural network.

Genetic algorithm optimizes the weight and threshold of BP neural network and the number of hidden layer nodes. The simulation results show that the GA optimized neural network GABP has better prediction effect than BP neural network. GABP is more suitable for ship trajectory prediction.

To sum up, the prediction of GABP is more accurate than that of BP, which reduces the prediction error to a certain extent. What needs to be improved is to simplify the program and reduce the time-consuming of the program.

Acknowledgments
Thank you, Mr. Meng Xin and Prof. Ma Shexiang for your guidance, and thank you for the support of National Natural Science Foundation of China. The corresponding author of this paper is Chen Xingqiang. You are welcome to criticize and correct the email address chenxingqiang22@163.com

Fund Project: National Natural Science Foundation of China (No.61601326, No.61371180).

References
[1] X.Tie,C.Fengjun,H.Qinyou,and Y.Chun, “Research on estimation of AIS vessel trajectory data based on Kalman filter algorithm”. Modern Electronics Technique, vol.5: pp.97-100, 2014.
[2] Z.Shuaibing,T.Cheng,L.Shan,and W.Dejun, “Track prediction of vessel in controlled waterway based on improved Kalman filter”. Journal of Computer Applications, vol.11, pp.3247-3250, 2012.
[3] L.Xiling,R.Qunsheng, and G.Ziqiang, “New Prediction Model of Ships GPS Navigation Positioning Trajectory”. Journal of Jiangnan University (Natural Science Edition), vol.6, pp.686-692, 2014.
[4] M.Chenchen,P.Chen,Y.Bo,L.Yining, and X.Xunqi, “Ship trajectory prediction based on Gaussian process regression”. Technology Innovation and Application, vol.31, pp.28-29, 2017.
[5] X.Tingting,L.Xiaoming, and Y.Xin, “BP neural network-based ship track real-time prediction”. Journal of Dalian Maritime University, vol.01, pp.9-11, 2012.

[6] Z.Rong,J.Yongxing,H.Qinyou,S.Chaojian, and W.Shengzheng, “Vessel behavior prediction based on AIS data and BP Neural Network”. Navigation of China, vol.2, pp.6-10, 2017.

[7] Z.Qingqing, and H.Xingshi, “An improved method of selecting node number of BP neural network and its application”. Journal of Xi’an Poly technic University. Vol.4, pp.502-505, 2008.

[8] X.Guangnan,C.Runwei. “Genetic algorithms and engineering design”. China: Science Press, 2000, pp.8-11.

[9] C.Qiong,Q.Feng, and W.Jun, “An effective multitarget track tracking algorithm”. Fire Control Radar Technology, vol.1:43-46, 2008

[10] G.Daqi, “On structures of supervised linear basis function feedforward three-layered neural networks”. Chinese J.Computers, vol.1, pp.80-86, 1998.

[11] Z.Peng,C.Jinhai,L.Qianfeng,M.Qiang, and H.Pengfei, “Research on trajectory interpolation method based on AIS”. Journal of Jimei University (Natural Science), vol.06:443-447, 2018.

[12] H.Li, “Theory, design and application of artificial neural network”. China: Chemical Industry Press, 2007, pp.47-67.