Predicting Stock Closing Price with Stock Network Public Opinion Based on AdaBoost-IWOA-Elman Model and CEEMDAN Algorithm

Lianghe Kang, Xiaoqiang Li, Lili Wu, Yue Li, Xia Zhao

***Abstract*—This paper proposes AdaBoost-IWOA-Elman algorithm to predict the stock closing price based on network public opinion. Firstly, stock opinion data obtained by Python Spyder is cleaned and quantified by various text mining algorithms. Then, the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) algorithm is used to decompose attributes serial selected by the Boruta and text mining algorithm. The improved WOA algorithm is used to optimize the weight and threshold of Elman. Finally, the AdaBoost algorithm is used to promote the forecasting performance of the IWOA-Elman. Experiments show that the root mean square error (RMSE) of AdaBoost-IWOA-Elman model reduces from 381.2384 to 147.7438. The prediction results are better than those obtained by other predicted algorithms, SVR, BPNN, Elman, and GA-Elman, which verifies the feasibility and effectiveness of the proposed method of anticipating stock price trends.**

***Index Terms*-AdaBoost, CEEMDAN, Elman, Text mining WOA**

I. INTRODUCTION

SINCE its establishment of the stock market, the prediction of stock closing price has been an intractable and knotty problem due to the many diverse features prevailing in the stock market, such as irregularities, instability, political influence, daily market trends and noise[1]. Researchers in many disciplines, including economics, physics, mathematics, and computer science, have realized stock price predictions based on historical trading data, such as opening price, trading volume, and price index. With the increasement in data amounts and types, traditional simplified and static historical stock market data has gradually failed to meet the needs of analysis and prediction [2-4]. Therefore, stock prediction is crucial for researchers and investment planners.

There are many studies to understand the predictive power of investor emotions and its impact on the stock market. Fluctuation in the stock market is governed by various macro-economic and micro-economic factors like political stability, government policies, general economic status, organization’s growth, global economic conditions, and investor’s psychology. Mehta et al. [5] predicted the stock market through the Twitter mood and got better results. WEI et al. [6] used Granger analysis to find social media sentiment has a significant impact on future closing prices. Thomas et al. [7] conducted an essential study on the combination of Natural Language Processing (NLP) and financial time series analysis. Nahil and Lyhyaoui [8] attempted to improve the prediction capacity of the stock price via an integrated prediction model based on Kernel Principal Component Analysis (KPCA) and Support Vector Machines for Regression (SVR). Kapanova et al. [9] applied GA to develop an automatic approach that searched for the optimal network architecture of ANN for fitting a specific function. Yang et al. [10] combined the improved Particle Swarm Optimization (PSO) algorithm and neural network to improve the stock price prediction effect. HU et al. [11] used the WOA algorithm to optimize BP neural network and forecasted the direction of stock Google Trend. Wang et al. [12] used the CEEMDAN algorithm to optimize the WOA algorithm, and achieved good prediction results in various economic and financial time series.

The stock market is a complex and changeable system with chaotic and non-stationary data. In traditional stock price forecasting, researchers use some simple linear models to analyze stock data. Although these models perform well in linear and stable time series, they perform poorly in nonlinear and non-stationary data. This paper uses an improved WOA algorithm to optimize the initial weight and threshold of the Elman neural network. It not only combines the advantages of Elman neural network wireless approximation, but also overcomes the problems of slow learning speed and low prediction accuracy. Regarding the attribute construction, the Boruta algorithm is used to select import attributes. To enhance forecasting accuracy and stability, the CEEMDAN algorithm is used to decompose

Manuscript received February 12, 2022; revised October 26, 2022. This work is supported by the Sheng Tong sheng innovation fund of Gansu Agricultural University (No. GSAU-STS-2021-15), National Natural Science Foundation (NO. 32060437), Postgraduate Education Project of Gansu Agricultural University (NO. 2020-19).

Lianghe Kang is an Assistant at the Department of Information Science and Technology, Gansu Agricultural University of Data Science and Data Technology, Lanzhou 730070, China (e-mail: kanglianghe@126.com)

Xiaoqiang Li is an Engineer at the General Station of Technical Extension of Agricultural Mechanization Gansu Province, Lanzhou 730046, China (e-mail: 499489560@qq.com).

Lili Wu is an Associate Professor at the Department of Information Science and Technology, University of Gansu Agricultural University of Information Science and Information Management, Lanzhou 730070, China (e-mail: 67676376@qq.com)

Yue Li is an Associate Professor at the Department of Information Science and Technology, Gansu Agricultural University of Data Science and Data Technology, Lanzhou 730070, China (e-mail: 249035844@ qq.com)

Xia Zhao is an Associate Professor at the Department of Information Science and Technology, Gansu Agricultural University of Data Science and Data Technology, Lanzhou 730070, China (e-mail: 58892778@ qq.com)

features selected by the Boruta algorithm, which solves the mode aliasing problem of EMD algorithm. Finally, to improve the prediction accuracy, the AdaBoost algorithm formed a strong predictor by combining five weak IWOA-Elman predictors. The results on the prediction of stock network public opinion show that AdaBoost-IWOA-Elman model significantly enhances the prediction performance in terms of forecasting accuracy and stability.

II. DATA DESCRIPTION AND ALGORITHM INTRODUCTION

A. Data Description

SSE 180 Index on Oriental Wealth Network is the research object, which can be accessed from this URL(http: //guba.eastmoney.com/list).The research data from January 4 to December 31, 2016is divided into two parts: the first part is the opinions of the stocks crawled by Python Spider from Oriental Wealth Network; the second part is the closing price downloaded from the CSMAR database.

B. Algorithm Introduction

(1) Text Mining

To improve the quality of predicted data, data cleaning technologies are used, including Chinese word segmentation, removing stop words, text representation, etc. The detailed flow is shown in Fig.1.

Step1: Python Spyder program [13] is used to get stock text information of SSE 180 Index on Oriental Wealth Network.

Step2: Python Jieba segmentation [14] is used to segment the title of comments received by the Python Spyder program into individual words.

Step3: Eliminate the stop words through a stop list combining current mainstream several stop lists [15], such as the Baidu stop list, the stop list of Harbin Institute of Technology, and the stop list of Machine Intelligence Laboratory of Sichuan University.

Step4: Use Term frequency–inverse document frequency (TF-IDF) [16] algorithm to evaluate the weight of a word in a file set or a document.

Step5: Use vector space model (VSM) [17] to represent the final text information, $\boldsymbol{D}=\boldsymbol{D}(\boldsymbol{T}_1,\boldsymbol{W}_1;\boldsymbol{T}_2,\boldsymbol{W}_2;...;\boldsymbol{T}_n,\boldsymbol{W}_n)$, $\boldsymbol{T}_i$ is a different word group from each other in the document.

(2) Boruta Algorithm

Because the time series data contains a lot of noise and high dimensions, Boruta algorithm is used to reduce dimensions. Boruta algorithm [18] is a fully correlated encapsulated attribute selection method. Due to high information prediction and regardless of the degree of correlation, the Boruta algorithm is used to select important attributes. Find all relevant attributes based on information prediction, which is more suitable for selecting essential attributes. The process and result of Boruta algorithm are shown in Fig.2.

Step1: Form a new matrix stitching the original attribute matrix with a shadow attribute matrix.

Step2: Train and output important attribute models using the new attribute matrix as input.

Step3: Calculate Z value of original and shadow attributes.

Step4: Find out the largest Z value in the shadow attribute, denoted as Z_{max} value.

Step5: Mark the original attribute with "Important", if Z value is more excellent than Z_{max} value, mark with "Unimportant" and delete the original attribute.

Step6: Repeat Step1-Step5 until all attributes are marked as "Important" or "Unimportant".

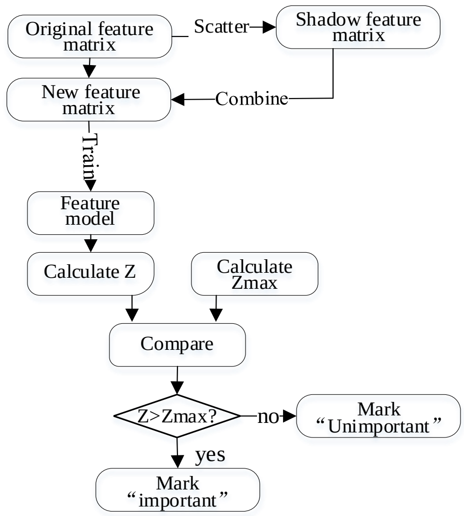


Fig. 2. Process of the Boruta algorithm

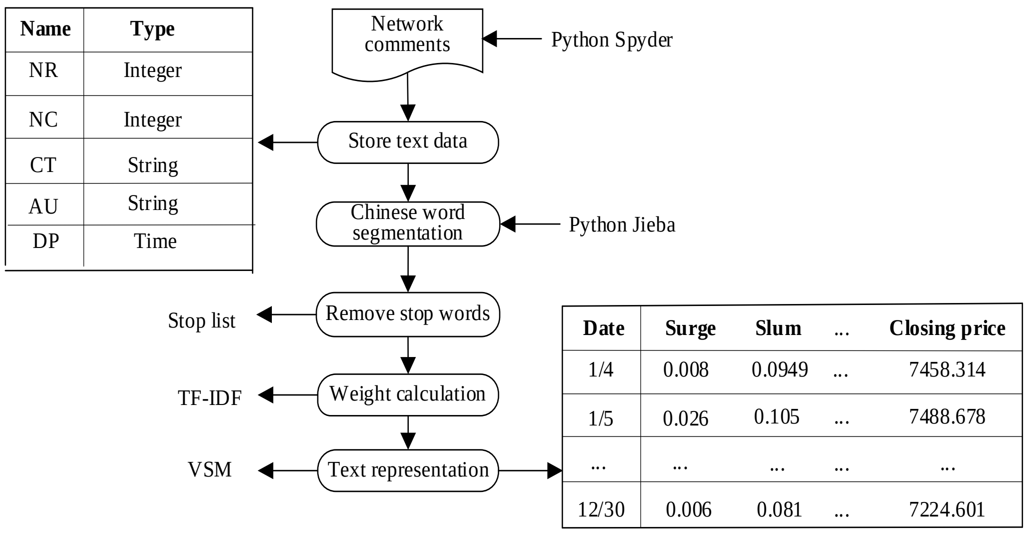


Fig. 1. Process of text mining

(3) CEEMDAN Algorithm

EMD algorithm separating original time series into a sum of intrinsic mode functions (IMFs) and one residue is a method of processing unsteady signals. Each IMF component represents different features of the original signal at different time scales. The EMD algorithm has a mode mixing problem. By adding Gaussian white noise to the original attributes, the EEMD algorithm has eliminated a mode mixing problem of the EMD algorithm [19]. However, to make up for the errors and low computational efficiency caused by overlapping signals in the EEMD algorithm, the CEEMDAN algorithm is used to improve the version of EEMD [20]. The realization of CEEMDAN is described in Fig.3.

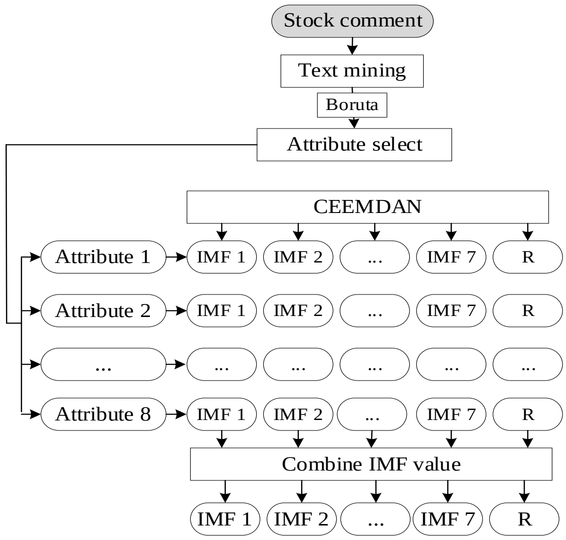


Fig. 3. Process of the CEEMDAN algorithm

Step1: Add Gaussian white noise $\omega(i)$ to original attribute data series $x_0(i)$ to obtain several new attribute series $x(i)$.

$$x_i = x_0(i) + \varepsilon_0 \omega(i) \tag{1}$$

Where ε_0 is noise coefficient.

Step2: Use the EMD algorithm to decompose each $x(i)$ and obtain the first $IMF_1(i)$ component.

$$\begin{cases} ml = mean(\max(x(i)) + \min(x(i))) \\ IMF_1(i) = x(i) - ml(x(i)) \end{cases} \tag{2}$$

Step3: Calculate the remainder of the attribute.

$$r_1(i) = x(i) - IMF_1(i) \tag{3}$$

Step4: Use the EMD algorithm to decompose each $x(i)$ again and get other IMF components.

$$\begin{cases} IMF_j(i) = r_{j-1}(i) - ml(r_{j-1}(i)) \\ r_j(i) = r_{j-1}(x) - IMF_j(i) \end{cases} \tag{4}$$

Step5: Calculate $TIMF1$ by adding IMF_1 to each attribute.

$$TIMF1 = \sum_{i=1}^{L} IMF_1(i) \tag{5}$$

Step6: Calculate the other $TIMF$.

$$TIMFj = \sum_{i=1}^{L} IMF_j(i) \tag{6}$$

Elman algorithm is a typical local-feedback recursive neural network, which introduces a feedback link based on BP neural network to enhance dynamic characteristics [21]. The additional context layer, also known as one-step time delays, can store the previous value of the hidden layer. Due to its dynamic memory and time-varying ability, Elman has been widely used in time series prediction since it was first proposed by Elman in 1990. Its structure is shown in Fig.4. State space equation of Elman neural network is shown in equation 7.

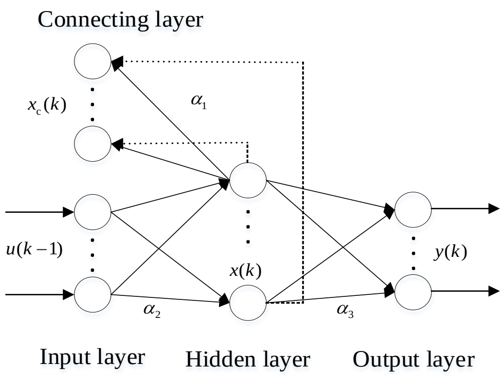


Fig. 4. The structure of Elman

$$\begin{cases} x(k) = F(\alpha_1 x_c(k) + \alpha_2 u(k-1) + b_1) \\ x_c(k) = x(k-1) \\ y(k) = G(\alpha_3 x(k) + b_2) \end{cases} \tag{7}$$

Where $x(k)$ and $x_c(k)$ are output vectors of the hidden and connecting layers at time k. $y(k)$ is the output vector. ∂_1, ∂_2 ∂_3 are weight, b_1 and b_2 are threshold of hidden layer and output layer. Function $F(\bullet)$ and $G(\bullet)$ are transfer functions in hidden layer and output layer.

(4) WOA Algorithm

Whales are biggest mammals in the world, and they have spindle cells in some areas of whales' brains, they can distinguish whales from other creatures. Therefore, whales can learn, think, judge and communicate like humans. The humpback whales are one of the most giant baleen whales, most of them like prey krill and small fish herds near the surface [22]. According to research findings, humpback whales adopt a special hunting strategy called the bubble-net feeding method, this maneuver technique is called 'upward-spirals' in the bubble-net feeding method. In the last maneuver, humpback whales swim fast toward the surface, and this behavior is called 'double loops', which includes three different stages: coral loop, lob-tail, and capture loop. Fig. 5 is the schematic diagram of the WOA algorithm. WOA uses a set of random candidate solutions and three rules to update the position of candidate solutions in each step, which is called encircling prey by spiral updating position and searching for prey. The description of relevant parameters can be found in Table I.

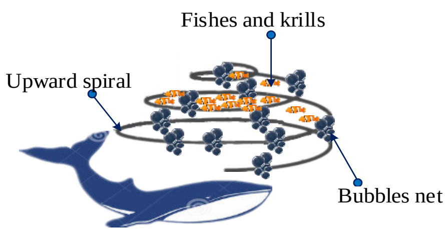


Fig. 5. Schematic diagram of WOA algorithm

Table I
PARAMETERS OF WOA ALGORITHM

Name	Meaning
W_i	The position of the whales
t	The current iterations
X^*	The location of prey
X	The current whale position
a	A variable in [2,0]
r	A random vector in [0, 1]
b	a constant
l	A random number in [-1,1]
A	A variation used to search for prey

Step1: Encircling Prey

The best location is the best candidate solution or is near the optimum [23]. After the best search agent is defined, the other search agents will try to update their positions toward the best solution or position. This behavior is shown in equations (8) and (9).

$$D = \left| C \bullet X(t) - X(t) \right| \tag{8}$$

$$X(t-1) = X^*(t) - A \bullet D \tag{9}$$

t is the current iterations, X^* is the location of prey, X is the current whale position, A and C are random coefficient.

$$A = 2a \bullet r - a \tag{10}$$

$$C = 2 \bullet r \tag{11}$$

a is linearly decreased from 2 to 0 throughout iterations (in both exploration and exploitation phases) and r is a random vector in [0, 1].

Step2: Spiral Updating Position

In this method, the distance between the whale located at (X, Y), and the prey located at (X^*, Y^*). The helix-shaped movement of humpback whales is simulated by creating a spiral equation as follows:

$$\vec{W}(t+1) = \vec{D}' \bullet e^{\mathrm{bl}} \bullet \cos(2\pi \mathrm{l}) + \vec{W}^*(t) \tag{12}$$

$$\vec{D}' = \left| \vec{W}^*(t) - \vec{W}(t) \right| \tag{13}$$

Where b is a constant defined by the logarithmic spiral shape, l is a random number in $[-1,1]$.

Step3: Search for Prey

Humpback whales search for prey by updating their position according to randomly selected candidates rather than the best candidates. $\vec{A}$ vector variation is used to search for prey. If $\left|\vec{A}\right| > 1$, the position will be updated randomly according to a randomly chosen search agent as in equation (11) and (12), if $\left|\vec{A}\right| < 1$, the position will be updated according to the best solution.

$$\vec{D} = \left| \vec{C} \bullet \vec{X}_{rand} - \vec{X}(t) \right| \tag{14}$$

$$\vec{X}(t+1) = \vec{X}_{rand}(t) - \vec{A} \bullet \vec{D} \tag{15}$$

(5) Improved WOA Algorithm

Weight is an essential parameter in the WOA algorithm, and constant weight will reduce the efficiency of the WOA algorithm, which is not conducive to global optimization. The more significant weight is beneficial to global optimization, and the smaller weight is useful to local mining. Based on this, an adaptive weight is proposed to ensure that the algorithm has a sizeable non-linear weight at the beginning of iteration with different weights [24]. The adaptive weight is defined as equation 16.

$$w = w_{\min} + (w_{\max} - w_{\min}) \bullet \mathrm{r} \bullet \mathrm{e}\left(-\frac{t}{\mathrm{Max\_iter}}\right) \tag{16}$$

Where $\mathrm{w}_{\min}$ is the minimum value of the weight, $\mathrm{w}_{\max}$ is the maximum value of the weight, r is a random number in $[0,1]$, t is the number of current iterations, $\mathrm{Max\_iter}$ is the maximum number of iterations. The improved position equation is as follows:

Encircling prey:

$$\vec{X}(t-1) = w \bullet \vec{X}^*(t) - \vec{A} \bullet \vec{D} \tag{17}$$

Spiral updating position:

$$\vec{W}(t+1) = \vec{D}' \bullet e^{bl} \bullet \cos(2\pi l) + w \bullet \vec{W}^*(t) \tag{18}$$

Search for prey:

$$\vec{X}(t+1) = w \bullet \vec{X}_{rand}(t) - \vec{A} \bullet \vec{D} \tag{19}$$

The improvement WOA algorithm, called IWOA, its pseudo code is presented in Algorithm 1 IWOA.

Algorithm 1 IWOA

```
Input: Max_iter , m
Output: w , b
Max_iter : the maximum number of iterations
m : the number of whales
w_min : the minimum value of weight
w_max : the maximum value of weight
function IWOA ( Max_iter , m ):
    Max_iter = 30
    m = 30
    w_min = 0.1
    w_max = 0.55
    for ( 1 ≤ t ≤ Max_iter ):
        /*get D , calculate the initial position
        a = 2 − t*(2 / Max_iter)
        a_2 = −1 + t*((−1) / Max_iter)
        for( 1 ≤ i ≤ size(D,1) ):
            r_1, r_2 = rand()
            A = 2*a*r_1 − a
            C = 2*r_2
            w = w_min + (w_max − w_min)*r*(−t / Max_iter)
            if p < 0.5 :
                if |A| < 1 :
                    X(t+1) = w · X*(t) − A · D
                else:
                    X(t+1) = w · X(t+1) − A · D
                end if
            else:
    X(t+1) = ω · X(t+1) − D · e^(lb) · cos(2πl)
            end if
        end for
        calculate the weight w and b
    return w , b
end function
```

(6) AdaBoost Algorithm

The AdaBoost algorithm is an iterative algorithm. To form a final strong predictor, AdaBoost continuously trains several weak predictors by adjusting the weights of weak predictors [25]. The detailed flow is shown in Fig.6.

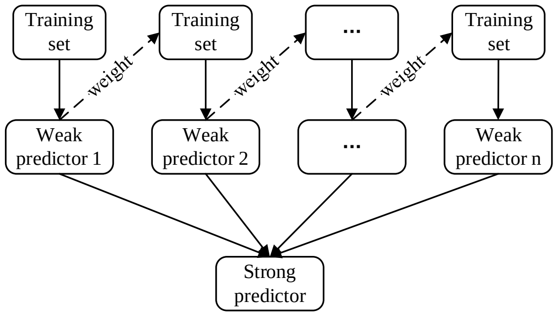


Fig. 6. Process of the AdaBoost algorithm

Step1: Select m samples to form training groups and assign initial equal weights to each sample in the original training set:

$$w_i^1 = \frac{1}{\mathrm{m}},\ i = 1,2\cdots\mathrm{m} \tag{20}$$

Where m is the number of samples.

Step2: Calculate the error of weak predictor. Get the error sum of the prediction sequence $g(t)$ obtained by weak predictor Elman.

$$e_t = \sum_{i=1}^{m} \left| g(t) - y_i \right| \tag{21}$$

Where y_i is the actual closing price.

Step3: Calculate weight of the prediction sequence，which AdaBoost uses e_t to calculate weight α_t.

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1\text{-}e_t}{e_t}\right) \tag{22}$$

Step4: Adjust weights. According to α_t, the next weight is adjusted as follows:

$$D_{t+1}(i) = \frac{D_t(i)}{B_t} * \exp[-\alpha_t y_i g_t(x_i)] \tag{23}$$

Where B_t is normalization factor.

Step5: Form a strong predictor function. After T iterations, the strong predictor function $F(x)$ is generated from the weak predictor function $f(g_t, a_t)$ of T group.

$$F(x) = \frac{\alpha_t}{\sum_{t=1}^{\mathrm{T}} \alpha_t} g(t) \tag{24}$$

(7) AdaBoost-IWOA-Elman Algorithm

To improve the performance and lower convergence speed of Elman, IWOA algorithm is used to optimize the initial weights and thresholds of Elman. At the same time, AdaBoost algorithm combines 5 IWOA-Elman predictors to form a strong IWOA-Elman predictor [26]. The process is shown in Fig.7.

Step1: Data Preprocess

This paper obtained 9 million pieces of data through Python Spider algorithm. Boruta algorithm selects 20 important attributes cleaning and quantifying by text mining algorithms. At the same time, to reduce data noise, CEEMDAN algorithm is used to decompose and reconstitute import attributes.

Step2: Get Optimal Initial Weights and Thresholds.

Initialization of Elman to determine the network structure, initial connection weight and threshold of the network. IWOA algorithm is initialized, including the number of whales m, the total number of iterations, the minimum and maximum value of weight, $w_{\min}$ and $w_{\max}$. Use encircling prey, spiral updating position, and search for prey of IWOA algorithm to select optimal the weights and thresholds. The fitness function is as follow:

$$F = \frac{1}{n}\sum_{i=1}^{n} \left| y_i - t_i \right| \tag{25}$$

y_i is the real stock closing price, $\hat{y}_i$ is the predicted price, n is the number or measure of predicted times.

Step3: Train Weak Predictor and Get Strong Predictor

Train weak predictor and adjust the initial weight of AdaBoost algorithm. We choose 5 AAFSA-Elman as the weak predictors.

Step4: Get a strong predictor

Combine five weak predictors to one strong predictor. Predict closing price through the strong predictor and estimate AdaBoost-IWOA-Elman model.

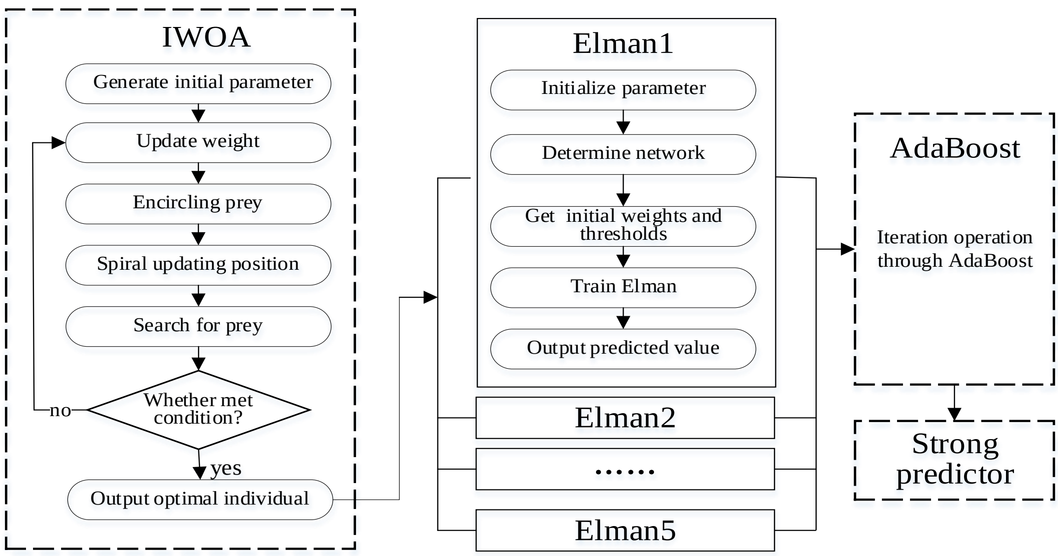


Fig.7. The process of AdaBoost-IWOA-Elman

III. EXPERIMENTS AND RESULTS

This study chose the closing price of SSE 180 Index from January 4 to December 30 in 2016. A total dataset is divided into 229 training datasets and 15 testing datasets.

A. Evaluation Indicator and Parameter Design.

To compare the predicted results of different models, we select indicator Mean Absolute Error (MAE) as evaluation index. In order to reduce the influence of outliers, Root Mean Square Error (RMSE) is used to calculate the error. Considering the ratio between the error and the actual value, we used the mean absolute percentage Error (MAPE) indicator [27].

$$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2} \tag{26}$$

$$MAE=\frac{1}{N}\sum_{i=1}^{N}\left|y_i-\hat{y}_i\right| \tag{27}$$

$$MAPE=\frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i-\hat{y}_i}{y_i}\right| \tag{28}$$

Where y_i is the actual stock closing price, $\hat{y}_i$ is the predicted price, N is the number of predicted times.

B. Parameter Design

In Elman algorithm, the topological structure is four layers, the activation function of hidden layer is $\mathrm{logsig}(x)=2/(1+\exp(-2x))-1$, and the activation function of output layer is $\mathrm{purelin}(x)=x$. The number of hidden layer is calculated with equation the $m=\sqrt{n+l}+\partial$, m, n and l are the numbers of hidden, input, and output layer, ∂ is a constant between 1 to 10. The maximum training times are 100, the error target is 0.0001, and the learning rate is 0.1.

In WOA algorithm, the total number of whales is $m=30$, the maximum number of iterations is $\mathrm{Max\_iter}=30$, about adaptive weights, $w_{\min}=0.1$, $w_{\max}=0.55$.

C. Result of Experiment

(1) Important Attribute and Data Characteristics

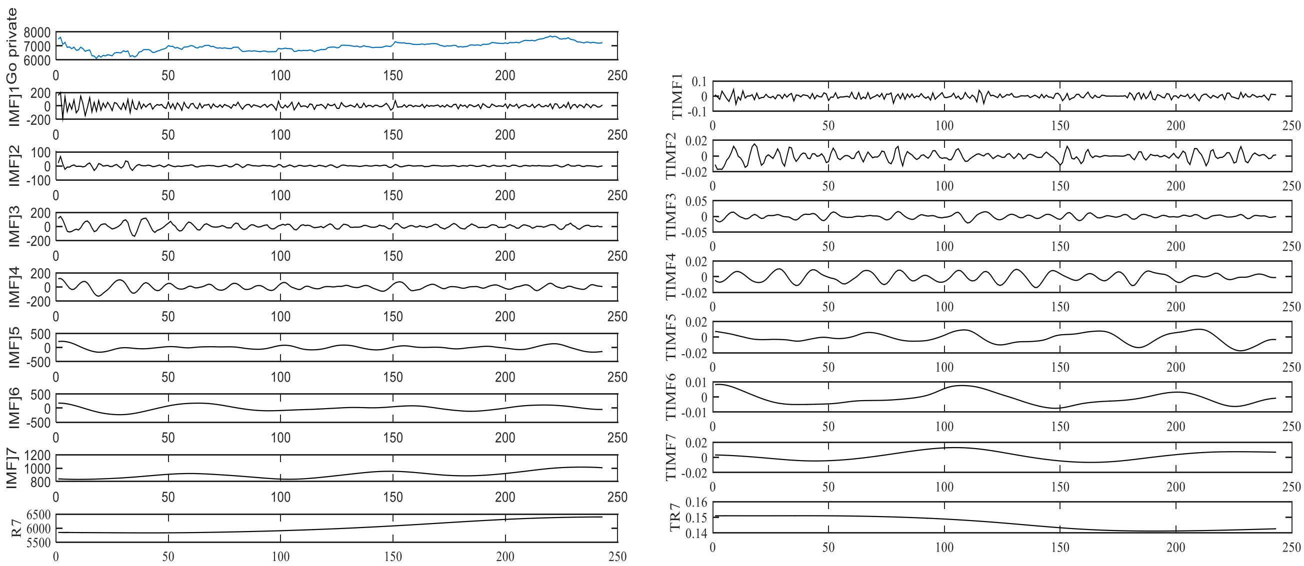
According to the contribution rate of each feature to the closing price, the Boruta algorithm selects twenty important attributes, from which we pick the first eight as modeling attributes. They are Go Private, Slump, Bull Market Stop Trading, Surge, Drop, Bullish, and Lock. Table II is the data characteristics of eight attributes, including max, min, mean, Q1 (25%), Q3 (75%), and interquartile range.

(2) Decompose Attributes

Figure 8(a) is the seven IMF components and one residue decomposed from the first attribute series by CEEMDAN. Each IMF is arranged from high frequency to low frequency. The First Few IMFS represent high-frequency components or noises in the original attribute series. Figure 8(b) is TIMF1 calculated by IMF1 of each attribute.

Table II
DATA CHARACTERISTICS OF ATTRIBUTES

Statistics	Go Private	Slump	Bull Market	Stop Trading	Surge	Drop	Bullish	Lock
Mean	0.014371	0.014108	0.013449	0.031923	0.006097	0.025254	0.038782	0.061008
std	0.007014	0.008225	0.008890	0.012946	0.002649	0.008283	0.009318	0.014495
min	0.003776	0.001696	0.000944	0.007452	0.000605	0.004924	0.005141	0.031833
25%	0.009468	0.008062	0.008337	0.023398	0.004069	0.020149	0.033088	0.051108
50%	0.013081	0.013016	0.011218	0.029304	0.006055	0.024158	0.037997	0.059231
75%	0.017219	0.017519	0.017382	0.037879	0.007656	0.028680	0.045004	0.066888
max	0.042362	0.048643	0.097807	0.099020	0.014766	0.068050	0.064089	0.113533



(a) (b)

Fig. 8. Decomposition and reorganization of CEEMDAN algorithm. (a) Decompose Go Bear Market by CEEMDAN. (b) Reorganize attribute date set.

(3) Influence of Weight on IWOA Algorithm

Figure 9(a) is the MAPE of IWOA-Elman model when weight is constant and adaptive weight in WOA algorithm. We can see MAPE of the IWOA algorithm with constant weight is higher than adaptive weight, and its fluctuation is also larger. we can find that weight has a significant influence on WOA algorithm. Figure 9(b) compares RMSE, MAE and MAPE when $w_{min} = 0.1$, $w_{max} \in [0.2, 0.9]$. Through analysis of Figure 8(b), the RMSE, MAE, and MAPE fluctuate are greater when w_{max} equals different values. After testing, the results of RMSE, MAE, and MAPE are the smallest when $w_{min} = 0.1$ and $w_{max} = 0.55$.

(4) Fitness Analysis

Figure 10(a) is the fitness of the optimized algorithm, including GA, WOA, and IWOAS. In Figure 10(a), we can see the fitness of GA is above 160 and starting to convergence when $trynum = 20$, it means GA algorithm has low convergence accuracy and slow convergence speed. The WOA algorithm has a faster convergence speed, but the convergence accuracy is worse than IWOA algorithm. Figure 10(b) is the fitness value of the weak predictor of AdaBoost algorithm. The fitness of IWOA- Elman1 is obviously high when $Max\_iter = 9$. When the first iteration finished, according to the error of IWOA-Elman1, AdaBoost algorithm adjusts the initial weight of IWOA-Elman 2, so the fitness value of IWOA-Elman 2 is smaller than IWOA-Elman 1. The other weak predictors follow the same rule. We can see the error is decreasing with iterations, so optimization of the AdaBoost algorithm is effective.

(5) Predicted Result of AdaBoost-IWOA-Elman

In Figure 11, Actual represents the real closing price, and the others are closing price predicted by Elman and other optimized algorithms. After analysis, we can find that the predicted value of AdaBoost-IWOA-Elman is the closest to the real closing price. The predicted closing price of WOA-Elman and IWOA-Elman is better than Elman. The most predicted value of AdaBoost-IWOA-Elman is more comparable to real value than WOA-Elman and IWOA-Elman models.

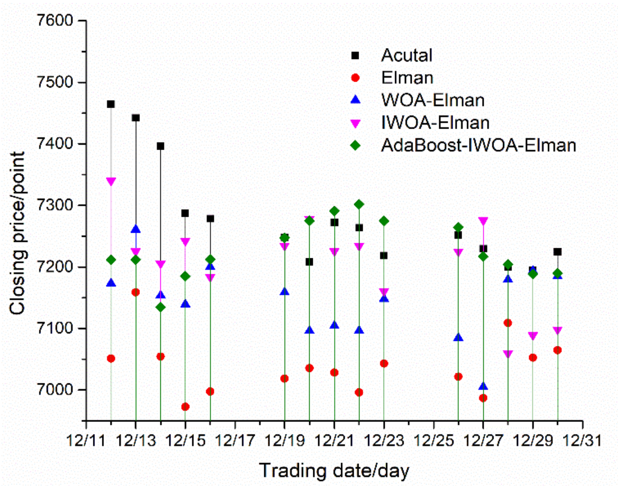


Fig. 11. Result of IWOA-Elman

A. Comparative Discussion

(1) Comparison of Predicted Results

This study uses Support Vector Regression (SVR), Back Propagation Neural Network (BPNN), and Genetic Algorithms (GA) to compare with the performance of the AdaBoost-IWOA-Elman model. Table III is the predicted result based on the original attribute set, Table IV and Table V based on EMD and CEEMDAN attribute set.

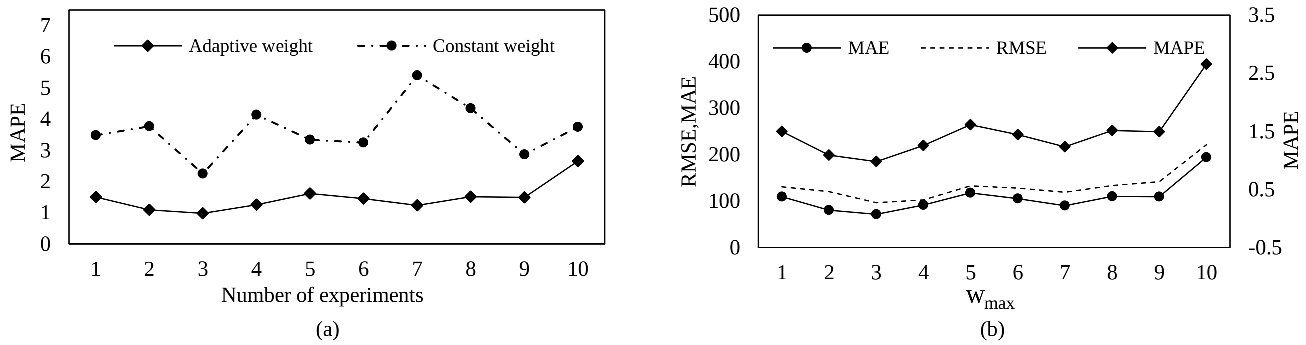


Fig. 9. Influence of weight on IWOA algorithm. (a) Influence of weight on IWOA algorithm. (b) WOA algorithm.

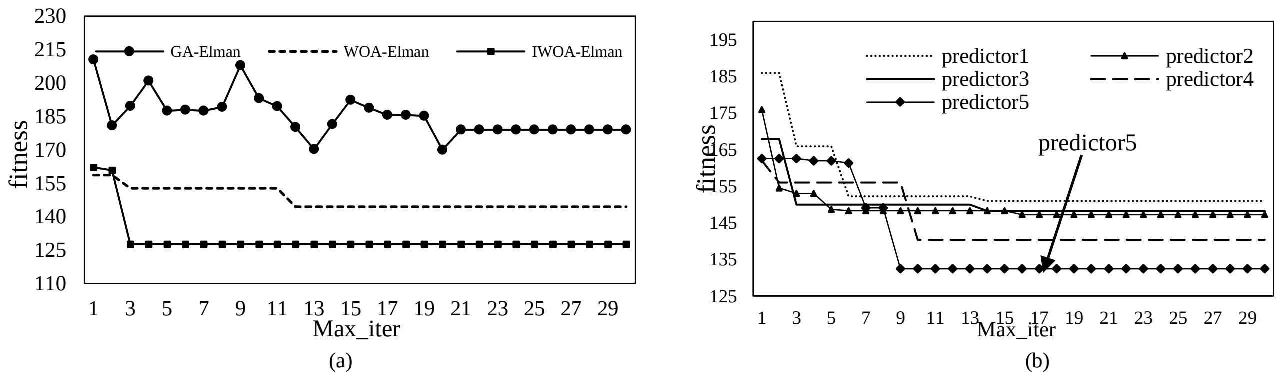


Fig. 10. Fitness of optimized algorithm and AdaBoost algorithm. (a) Fitness of optimized algorithm. (b) Fitness of AdaBoost algorithm

Table III
PREDICTED RESULT OF DIFFERENT MODELS BASED ON ORIGINAL ATTRIBUTE SET

Date	Actual	BPNN	SVR	Elman	GA-Elman	WOA-Elman	IWOA-Elman	AdaBoost-IWOA-Elman
12/2	7464.64	7432.344	7334.745	7064.244	6941.442	7045.931	7108.215	7046.065
12/13	7441.98	7315.179	7303.293	7153.734	7050.737	7069.310	7190.619	7075.249
12/14	7395.85	7246.114	7149.763	7151.517	6948.719	7065.600	7084.206	7058.861
12/15	7287.03	7153.901	7073.214	7086.971	7002.846	7089.690	7166.549	6919.636
12/16	7278.28	7035.953	7033.664	6982.646	6946.907	7096.146	7174.620	7023.355
12/19	7248.14	6957.461	7002.325	6942.381	6987.9	7104.189	7219.887	7014.071
12/20	7208.2	6955.168	6990.204	6980.848	7017.785	7086.236	7193.800	7102.374
12/21	7272.17	7032.699	6973.954	7049.398	6914.652	7029.199	7022.941	7298.845
12/22	7263.44	7022.041	6993.566	7042.541	7024.657	7032.343	7106.219	7211.886
12/23	7218.32	7033.79	7021.832	6969.205	7036.853	7019.826	7139.618	7187.085
12/26	7251.94	7001.503	6989.138	6987.375	7012.994	7002.337	7102.757	6983.475
12/27	7229.39	7011.642	6951.012	7029.151	6882.905	6957.227	6959.619	6954.928
12/28	7199.79	6983.551	6917.453	7059.151	7206.777	7022.107	7210.064	6917.692
12/29	7194.7	6912.788	6937.541	7013.575	7090.297	7004.220	7106.906	7121.138
12/30	7224.6	6884.199	6946.165	6960.308	6977.931	6965.850	7052.310	7035.856

Table IV
PREDICTED RESULT OF DIFFERENT MODELS BASED ON EMD ATTRIBUTE SET

Date	Actual	BPNN	SVR	Elman	GA-Elman	WOA-Elman	IWOA-Elman	AdaBoost-IWOA-Elman
12/12	7464.64	7519.905	6883.669	7002.249	6931.511	6952.873	7123.787	7132.695
12/13	7441.98	7528.857	6921.831	6960.506	6895.405	6997.915	7134.462	7078.928
12/14	7395.85	7490.549	7169.796	6962.969	6874.632	6970.401	7154.934	7199.492
12/15	7287.03	7522.379	6916.501	6926.167	6852.218	6845.825	7125.651	7276.336
12/16	7278.28	7532.624	6787.01	6917.315	7030.513	6973.252	7103.594	6945.173
12/19	7248.14	7554.291	6790.423	6895.217	7047.53	7024.137	7079.838	7050.925
12/20	7208.2	7542.024	6650.549	6905.012	6619.774	6988.807	7085.435	6962.393
12/21	7272.17	7465.606	7383.861	6958.599	6645.778	7139.386	7134.073	6896.166
12/22	7263.44	7486.025	6787.62	6943.858	6577.924	7154.802	7098.836	7018.278
12/23	7218.32	7496.745	7123.266	6941.51	7010.715	7195.111	7067.004	7232.235
12/26	7251.94	7474.171	6820.037	6951.361	6863.264	6888.724	7080.015	7103.606
12/27	7229.39	7427.483	7122.991	6982.251	6910.495	7028.330	7139.392	7021.617
12/28	7199.79	7535.449	6808.297	6892.423	7248.296	7024.702	7062.91	7126.755
12/29	7194.7	7489.887	6761.664	6924.419	6956.898	7027.284	7068.606	6778.193
12/30	7224.6	7453.082	6854.588	6950.146	7189.904	7119.184	7076.442	6782.005

Table V
PREDICTED RESULT OF DIFFERENT MODELS BASED ON CEEMDAN ATTRIBUTE SET

Date	Actual	BPNN	SVR	Elman	GA-Elman	WOA-Elman	IWOA-Elman	AdaBoost-IWOA-Elman
12/12	7464.6	**7551.673**	7314.961	7051.1	7575.562	7173.1	7340.3	7211.5
12/13	7442.0	7626.309	6963.869	7158.8	**7621.077**	7260.5	7225.4	7212.2
12/14	7395.9	7668.369	6990.827	7054.5	7648.103	7153.8	**7205.5**	7134.7
12/15	7287.0	7654.779	7024.032	6972.8	7647.480	7139.2	**7242.7**	7185.0
12/16	7278.3	7567.464	7018.191	6997.6	7574.734	7200.3	7183.6	**7212.4**
12/19	7248.1	7508.472	6966.493	7018.4	7516.944	7158.9	7234.3	**7247.6**
12/20	7208.2	7474.393	6975.981	7035.4	7514.017	7096.2	7278.0	**7274.7**
12/21	7272.2	7489.993	7021.538	7028.3	7512.228	7104.5	7225.9	**7291.3**
12/22	7263.4	7448.941	7015.410	6996.0	7465.949	7096.3	7234.5	**7301.6**
12/23	7218.3	7438.134	7051.444	7043.0	7404.769	7147.6	7160.4	**7274.7**
12/26	7251.9	7420.649	7012.878	7021.9	7372.857	7084.7	7225.1	**7264.5**
12/27	7229.4	7437.302	6931.581	6986.9	7366.610	7004.9	7275.9	**7216.9**
12/28	7199.8	7454.481	6939.266	7109.0	7310.992	7179.2	7059.7	**7204.3**
12/29	7194.7	7434.989	6939.899	7052.7	7257.374	**7193.6**	7089.2	7188.5
12/30	7224.6	7398.709	6943.001	7064.7	**7242.391**	7184.9	7098.2	7189.8

In Table V, BPNN, SVR, and GA models are used to compare with predicted effect of AdaBoost-IWOA-Elman model based on datasets selected by CEEMDAN algorithm. Of the 15 predicted values, there are 12 predicted values of AdaBoost-IWOA-Elman closer to the true value. Compared with SVR, BPNN, Elman, GA-Elman, WOA-Elman, and IWOA-Elman algorithm, the predicted effect of AdaBoost-IWOA-Elman algorithm is obviously better. We can find that the same rule in Table III and Table IV. Compared with the same algorithm, the prediction result based on CEEMDAN datasets is better than EMD and the original datasets.

(2) Comparison of Predicted Effect

To further evaluate the performance of IWOA-Elman, use Figure 12 to compare with the predicted effect of different models and datasets.

Figure 12(a) shows the comparison of Elman with BPNN and SVR. We can see that the results of SVR have a significant fluctuation, so the performance of SVR is inferior to the Elman and BPNN. The predicted results of Elman and BPNN are not significant. Figure 12(b) is the comparison of GA, WOA, WOA, and AdaBoost-IWOA. The error of GA is the biggest, the predicted effect of IWOA is obviously better than WOA. Although the predicted effect of IWOA is better than AdaBoost-IWOA in 12/12-12/15, we can find that AdaBoost-IWOA is better than IWOA in the whole prediction cycle. Figure 12(c) is the comparison of WOA based on different datasets, including datasets gained from EMD, CEEMDAN and the original closing price. We can find that the predicted effect of WOA based on the original datasets has a significant fluctuation. The predicted value of IWOA based on the CEEMDAN datasets is better than WOA based on EMD datasets. Figure 12(d) compares IWOA based on three datasets. After analysis, it is found that Figure 11(d) has the same rule as Figure 13(c). The result indicates that model IWOA-Elman based on the CEEMDAN algorithm has higher predicted accuracy and better predicted effect.

(3) Comparison of predicted error

Table VI evaluates the predicted error, including RMSE, MAE, and MAPE of different models. It shows the error rate of AdaBoost-IWOA-Elman based on the CEEMDAN dataset is smaller than others. Compared with SVR, BPNN, and Elman, the MAPE of AdaBoost-IWOA-Elman model based on the CEEMDAN dataset reduced by 2.4913%, 1.8828%, and 2.0848% respectively. Compared to the optimized algorithm GA, WOA, and IWOA, the MAE of AdaBoost-IWOA-Elman proposed in this paper reduced by 99.295, 62.00934 and 25.30457 respectively. At the same time, the predicted result of the same model based on the CEEMDAN dataset is obviously superior to the EMD dataset and the original dataset. Compared to the EMD dataset and the original dataset, the RMSE of AdaBoost-IWOA-Elman based on CEEMDAN dataset reduced to 41.8465 and 105.446. In summary, the model AdaBoost-IWOA-Elman based on CEEMDAN is effective in stock network public opinion forecasting.

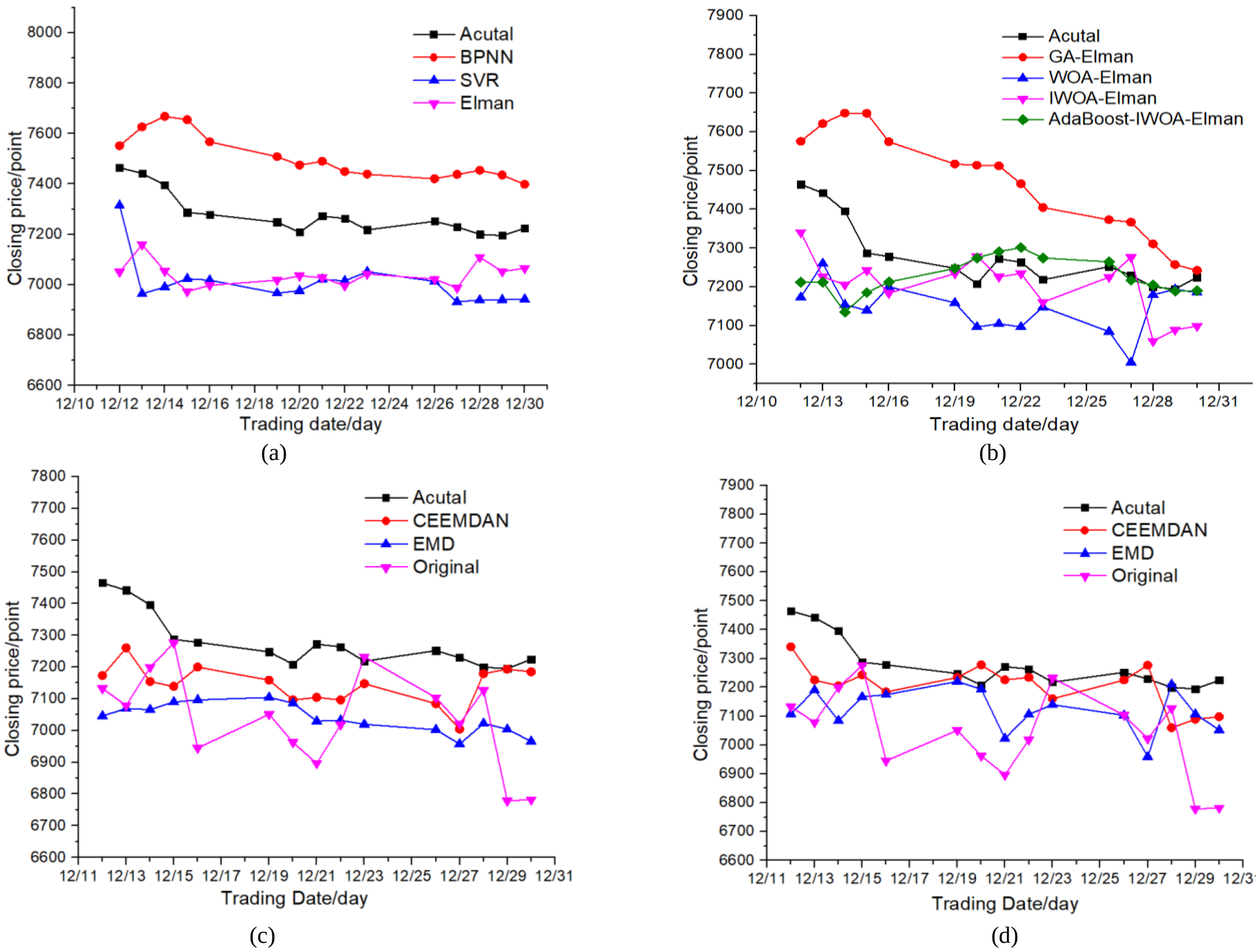


Fig. 12. Compare of different predicted models. (a) Comparison of single models. (b) Comparison of different optimization algorithm. (c) Comparison of WOA-Elman based on different datasets. (d) Comparison of IWOA-Elman based on different datasets

Table VI
PREDICTED ERROR OF DIFFERENT MODELS

Models	CEEMDAN			EMD			Original		
	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE
SVR	282.2788	271.2732	3.7229	242.2766	237.3734	3.267	406.7571	374.6498	5.1416
BPNN	234.9613	226.4124	3.1144	226.0411	213.3425	2.9407	393.7889	366.804	5.0393
Elman	250.4869	241.9297	3.3164	254.1273	247.0283	3.3882	**381.2384**	**358.812**	**4.9432**
GA-Elman	212.1418	190.1745	2.6116	304.1921	276.6027	3.7843	355.7306	341.3112	4.6822
WOA-Elman	175.9695	152.8888	2.0921	175.9695	239.2172	3.2765	294.1927	256.5157	3.5071
IWOA-Elman	145.6567	113.0553	1.44531	149.6023	157.3793	2.1501	274.7717	240.1002	3.2948
AdaBoost-IWOA-Elman	**147.7438**	**90.8795**	**1.2316**	189.5903	2.1501	157.3793	253.1893	218.7535	2.993

IV. CONCLUSION

This study takes the stock network public opinions on Oriental Wealth Network as the research object, crawls, cleans, and quantifies opinions through text mining technology. Boruta and CEEMDAN algorithms are used to select and reconstruct import attributes. Through adaptive weight to improve WOA algorithm, the IWOA algorithm is used to optimize the initial weights and thresholds of Elman. At the same time, the AdaBoost algorithm forms a strong predictor to predict the stock closing price. The experiment shows that the model AdaBoost-IWOA-Elman proposed has more evident advantages in prediction accuracy than BPNN, SVR, and GA, which proves the reliability and effectiveness of the model proposed.

REFERENCES

[1] Yanni Tang, Jiamou Liu, Wu Chen, “Exchange, adopt, evolve: Modeling the spreading of opinions through cognition and interaction in a social network,” *Information Sciences*, vol. 551, pp. 1-22, 2021.

[2] Kumar, J. Deepak, Prasanthi, B. and Venkatesh, J, “An intelligent cognitive-inspired computing with big data analytics framework for sentiment analysis and classification,” *Information Processing& Management*, vol. 59, no. 1, pp. 1-15, 2022.

[3] Srivastava P R, Zhang Z P, Eachempati P, “Deep Neural Network and Time Series Approach for Finance Systems: Predicting the Movement of the Indian Stock Market,” *Organizational and End User Computing*, vol. 33, no. 5, pp. 204-226, 2021.

[4] Dami S, Esterabi M, “Learning Sentimental and Financial Signals with Normalizing Flows for Stock Movement Prediction,” *IEEE Signal Processing Lettere*, vol. 29, pp. 414-418, 2022.

[5] Mehta P, Pandya S, Kotecha K, “Harvesting social media sentiment analysis to enhance stock market prediction using deep learning,” *Peerj Computer Science*, 2021.

[6] WEI G Z, Xu Y L, et al, “Model and forecast stock market behavior integrating investor sentiment analysis and transaction data,” *Cluster Comput*, vol. 20, pp. 789-803, 2017.

[7] Thomas Renault, “Intraday online investor sentiment and return patterns in the U.S. stock market,” *Journal of Banking & Finance*, vol. 84, pp. 25-40, 2017.

[8] Nahil A, Lyhyaoui A, “Short-term stock price forecasting using kernel principal component analysis and support vector machines: the case of Casablanca stock exchange,” *Procedia Comput Sci*, vol. 127, pp. 161–169, 2018.

[9] Kapanova K G, Dimov I,Sellier J M,“A genetic approach to automatic neural network architecture optimization,” *Neural Computing and Applications*, vol 29, no. 5, pp. 1481-1492, 2018.

[10] Yang F W, Chen JJ, Liu Y C, “Improved and optimized recurrent neural network based on PSO and its application in stock price prediction,” *Soft Computing*, 2021.

[11] Hu H, Tang L, Zhang S, et al, “A Combined Prediction Model for Hog Futures Prices Based on WOA-Light GBM-CEEMDAN,” *Complexity*, 2022.

[12] Wang X, Gao S, Guo YB, Zhou SY, Duan YH, Wu DY, “A Modified Whale Optimization Algorithm and Its Application in Seismic Inversion Problem,” *Mobile Information Systems*, 2022.

[13] Brunekreef, Tammo E, Otten, Henny G, “Text Mining of Electronic Health Records Can Accurately Identify and Characterize Patients with Systemic Lupus Erythematosus,” *ACR Open Rheumatology*, vol. 3, no. 2, pp. 65-71, 2021.

[14] Rahaman, Tariq, “Discovering New Trends & Connections: Current Applications of Biomedical Text Mining,” *Medical Reference Services Quarterly*, vol. 40, no. 3, pp. 329-336, 2021.

[15] Wang LX, Jiang M, Zhou Q, “Short Text Mining Framework with Specific Design for Operation and Maintenance of Power Equipment,” *Journal of Power and Energy Systems*, 2021, vol. 7, no. 6, pp. 1267-1277.

[16] Alammary A S, “Arabic Questions Classification Using Modified TF-IDF,” *IEEE Access*, vol. 9, pp. 95109-95122, 2021.

[17] Parra-Royon M, Atemezing G, Benitez JM, “Semantics of Data Mining Services in Cloud Computing,” *IEEE Transactions on Services Computing*, vol. 15, pp. 945-955, 2022.

[18] Qu JH, Ren K, Shi XY, “Binary Grey Wolf Optimization-Regularized Extreme Learning Machine Wrapper Coupled with the Boruta Algorithm for Monthly Stream flow Forecasting,” *Water Resource Manage*, vol. 35, pp. 1029-1045, 2021.

[19] Liu Y, Wang L H, “Drought Prediction Method Based on an Improved CEEMDAN-QR-BL Model,” *IEEE Access*, vol. 9, pp. 6050-6062, 2021.

[20] Shang Y, Miao L, Shan YP, Gnyawali KR Zhang, “A Hybrid Ultra-Short-Term and Short-Term Wind Speed Forecasting Method Based on CEEMDAN and GA-BPNN,” *Weather and Forecasting*, vol. 37, no. 4, pp. 415-428, 2022.

[21] Baranilingesan I, “Optimization algorithm-based Elman neural network controller for continuous stirred tank reactor process model,” Current Science, vol. 120, no. 8, pp. 1324-1333, 2021.

[22] Li Q, Li MQ, Wang R, “WOA-BP algorithm for crack detection in coal bed methane reservoirs and its application,” *Chinese Journal of Geophys Ice-Chinese Edition*, vol. 65, no. 2, pp. 773-784, 2022.

[23] Duan L J, LIAN Z Y, Qiao Y H, “ELM Algorithm Optimized by WOA for Motor Imagery Classification,” *International Conference on Extreme Learning Machine*, vol. 14, pp. 51-60, 2021.

[24] Pankaja K, Suma V, “Plant Leaf Recognition and Classification Based on the Whale Optimization Algorithm (WOA) and Random Forest (RF),” *Journal of The Institution of Engineers (India): Series B*, vol. 101, pp. 597–607, 2020.

[25] Wang W Y, Sun D C, “The improved AdaBoost algorithms for imbalanced data classification,” *Information Sciences*, vol. 563, pp. 358-374, 2021.

[26] Xia T B, Zhuo P C, Xiao L, “Multi-stage fault diagnosis framework for rolling bearing based on OHF Elman AdaBoost-Bagging algorithm,” *Neurocomputing*, vol. 443, pp. 237-251, 2021.

[27] Sun G L, Shang Y L, Zhang ROX, “An Efficient and Robust Improved Whale Optimization Algorithm for Large Scale Global Optimization Problems,” *Electronics*, vol. 119, no. 4, 2022.