Received March 8, 2020, accepted March 23, 2020, date of publication March 26, 2020, date of current version April 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983588

Hybrid Method for Short-Term Time Series Forecasting Based on EEMD

YUJUN YANG AND YIMEI YANG

School of Computer Science and Engineering, Huaihua University, Huaihua 418008, China

Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, Huaihua 418000, China

Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province, Huaihua 418000, China

Corresponding author: Yimei Yang (yym1630@163.com)

This work was supported in part by the Scientific Research Fund of Hunan Provincial Education under Grant 19C1472 and Grant 17C1266, in part by the Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province under Grant ZNKZ2018-5, in part by the Key Scientific Research Projects of Huaihua University under Grant HHUY2019-08, in part by the Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, and in part by the Constructing Program of the Key Discipline in Huaihua University.

ABSTRACT To improve the prediction effect of time series, we make a systematic study of various time series prediction methods based on statistics and machine learning in this paper. In the experiment, we compare the prediction results of several prediction methods. In particular, much research has been done on the selection of experimental data because representative time series data can better test the effectiveness and practicability of the prediction method. Based on the idea of divide and conquer of complex problems and the strategy of continuous optimization of machine learning, we proposed the prediction methods of LSTM-TFE, LR-TFE, and BR-TFE combined the EEMD, LSTM, LR, and BR methods in this paper. These methods use EEMD to decompose complex time series into several relatively milder, more regular and stable subsequences. Then the prediction model of each subsequence based on machine learning is carried out by using the LSTM, LR, or BR methods. We use these prediction models to predict the value of each subsequence. Finally, the value of multiple subsequences is fused to form the prediction results of the original complex time series. To verify the proposed method comprehensively, we select three representative time series data to test this paper. From the experimental results, we found that the proposed method has a good effect.

INDEX TERMS LSTM, short-term forecasting, EEMD, price prediction, time series.

I. INTRODUCTION

Time series forecasting (TSF) is a hot research field. The practical application of time series analysis method also develops with the time series analysis method. In many fields, time series prediction plays an important role, especially when social organizations, enterprises or individuals make critical strategic decisions. Prediction is the basis of the decision, and the decision is the continuation of prediction. Therefore, accurate prediction is crucial to make the correct decision. An accurate prediction can produce the correct decision. Without accurate prediction, it leads to the wrong decision. For example, accurate bankruptcy forecast and credit score [1] can help financial institutions reduce financial risks or avoid a financial crisis. Forecasting power load [2] can help power system planning and deployment effectively. Network traffic forecasts [3] can help service providers improve their service quality. Forecasts of physical phenomena in nature, such as weather [4] and earthquake [5], can help people take necessary measures to prevent disasters, and so on.

The researchers proposed many linear time series prediction methods based on statistics. The common goal of these linear methods is to use a time series analysis to predict the future value of time series. Among them, the more successful methods are Holt winters exponential (HWE) smoothing [6], autoregressive integrated moving average (ARIMA) [7], and linear regression (LR) [8] based on machine learning. In recent years, with the rapid development of computing technology, artificial neural network (ANN) [9], fuzzy comprehensive evaluation (FCE)[10]–[12], wavelet analysis (WA) [13]–[16] and support vector machine (SVM) [17]–[19] are widely used in short-term time series prediction. Many researchers successfully applied artificial neural networks in the field of classification [20]–[24] and regression [25]–[29],

The associate editor coordinating the review of this manuscript and approving it for publication was Chaitanya U. Kshirsagar.

but it is easy to fall into the dilemma of local optimum. In 2006, Huang *et al.* [30] proposed the concept of deep neural networks (DNN). Since then, deep learning (DL) has gradually become the most popular method in the field of machine learning [31]–[39]. In recent years, deep learning methods and deep learning platforms have developed rapidly. They have been widely used in various fields, such as Google's machine translation, Facebook's chat robot, Google's Alphago zero machine chess player, Google and Oxford University's lip recognition system, etc.

The prediction of time series is an open and complex problem, which involves a wide range of fields and is difficult to predict accurately. For complex problems, people usually use the divide and conquer method to solve them. In various integration methods, the divide and conquer method is widely used [40]–[41]. The divide and conquer method is also often used in the prediction of time series, which is to decompose time series into multiple subsequences, predict many subsequences respectively, and finally merge the prediction results of many subsequences. There are two main methods to decompose time series: wavelet transform decomposition and empirical mode decomposition. Wavelet transform decomposition is the first method applied to time series decomposition. Through time-frequency analysis, the original time series is decomposed into several orthogonal subsequences. Many scholars use the adaptive wavelet neural network model with feed-forward neurons to predict short-term time series. Empirical mode decomposition (EMD) is another decomposition method suitable for time series prediction. It is a part of the Hilbert Huang transform (HHT) [42]. EMD and wavelet transform decomposition is different. Wavelet transform decomposition is based on the frequency domain, while the EMD process is based on the time domain. All the above solutions are based on the prediction of the time series itself, without processing the time series itself. How to improve the prediction effect by combining the existing prediction methods and decomposing the time series has become a challenge. As we all know, the change of time series has uncertainty and complexity. Individual prediction methods are often only valid for some time series, but not for other time series. At the same time, the hybrid methods obtained better prediction results to some extent. Besides, the EEMD method decomposes the original complex time series into several sub time series with relatively stable changes. Combined with the current effective prediction methods, the prediction results of the sub time series with relatively stable changes are better in theory. Therefore, in this paper, we proposed some hybrid methods for short-term time series forecasting based on an improved EMD method EEMD (Ensemble Empirical Mode Decomposition) [43], [47], [48] and machine learning algorithm.

The rest of this paper is organized as follows: Section 2 briefly reviews the related concepts, terms, the LSTM, and the EEMD. Section 3 demonstrates the hybrid methods for short-term time series forecasting based on EEMD and the framework of the proposed methods. In section 4, we mainly introduce the experimental data in our study. Section 5 presents the simulation experiment and empirical analyzes of the proposed methods. Finally, some conclusions are given, and some future works are pointed out in the final section.

II. RELATED WORKS

To improve the prediction effect of time series, we proposed three hybrid methods for short-term time series forecasting using EEMD and three methods that are long short term memory (LSTM) neural network, linear regression (LR), and Bayesian ridge regression (BR). The three proposed methods are based on the idea of divide and conquer of complex problems and the strategy of continuous optimization of machine learning. In this paper, we call them LSTM time series forecasting based on EEMD (LSTM-TFE), LR time series forecasting based on EEMD (LR-TFE), and BR time series forecasting based on EEMD (BR-TFE) respectively. Below we first discuss the related concepts and terms, and then review the ensemble empirical mode decomposition and long short term memory, as related to this paper.

A. EMPIRICAL MODE DECOMPOSITION

Empirical mode decomposition (EMD) was proposed by Xu *et al.* [43] in 1998, which is a very efficient time series decomposition method. Based on the local characteristics of time series data, the EMD can effectively extract the original time-series data from the time series with noise. The EMD can also achieve better results in the decomposition of other complex time series. Therefore, it has been successfully applied in many practical fields. However, the main disadvantage of EMD is that it is easy to produce pattern mixing, that is, scale separation. The so-called pattern mixing is that there are signals of different frequencies or scales in the same sub time series, or the signals of the same frequency or scale are decomposed into different sub time series.

Because of the above problems, many researchers put forward some solutions. In 2009, Xu *et al.* [43] proposed an adaptive empirical mode decomposition method (EEMD) based on the statistical characteristics of white noise signals. This decomposition method decomposes the original time series into several sub time series. The sum of all sub time series is the original time series. Each sub time series is called Intrinsic Mode Function(IMF). Compared with wavelet transform decomposition and other decomposition techniques, the EEMD decomposition method has the characteristics of intuitionistic, direct, adaptive, and experiential. The EEMD no need to set decomposition series and can automatically complete multi-level decomposition according to the time series itself. The practical application in many fields proves that EEMD is an effective method to decompose time-frequency data from non-stationary and non-linear time series. The following is a brief introduction to the process of the EEMD decomposition time series.

Suppose EEMD decomposes a time series X into n subsequences IMF_i $(i = 1, 2, \ldots, n)$. According to the regulation of the EEMD decomposition time series, the last subsequence

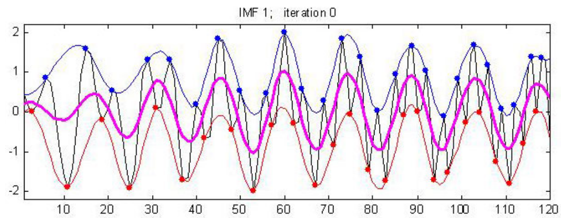


FIGURE 1. The process chart of EEMD decomposition time series.

IMF_n is called residual subsequence R_N. In fact, EEMD decomposes a time series X into n-1 subsequence IMF component C_i ($i = 1, 2, \ldots, n-1$) and a residual subsequence R_N. The steps of EEMD decomposition algorithm are as follows:

(1) For a given time series $x(t)$ is shown in the black line in Fig. 1, the three-degree spline interpolation method of local maximum and minimum values is used to create its upper and lower envelope lines, which are shown in the blue line and red line in Fig. 1.

(2) Calculate the mean value $m(t)$ of the upper and lower envelope lines, as shown in the thick purple-red line in Fig. 1.

(3) By subtracting the mean value $m(t)$ of the upper and lower envelope from the original time series $x(t)$, the first subsequence IMF_1 component $h(t) = x(t)$ - $m(t)$ is obtained.

(4) Taking the subsequence component $h(t)$ as a new time series $x(t)$, and repeating steps (1) to (3) until the stop condition is met. The stop conditions are as follows:

(a) The mean value $m(t)$ of the upper and lower envelope lines is approximately equal to zero;

(b) The number of extreme points of component $h(t)$ is equal to or at most different from the number of zero-crossing points;

(c) The predefined maximum number of iterations has been reached.

(5) $h(t)$ is taken as a subsequence IMF component C_i ($i = 1, 2, \ldots, n-1$), and calculate the remaining subsequence R $R(t) = x(t) - h(t)$.

(6) Use the remaining subsequence $R(t)$ as the new time series $x(t)$ to calculate the next IMF, and repeat steps (1) to (5) until all the IMF is obtained or the maximum decomposition level is reached.

Finally, EEMD has decomposed time series X into n subsequences, which are expressed by Eq.1.

$$x(t) = \sum_{i=1}^{n-1} (C_i) + R_n, \quad i = 1, 2, \ldots, n-1 \tag{1}$$

where the number n of sub time series depends on the complexity of the original time series. Fig. 2 shows a decomposition process of the EEMD for the time series Eq. 2.

$$x(t) = sin(2\pi * 10t) + sin(2\pi * 100t) \tag{2}$$

where t = 0, f, 2f, ..., 300f, $f = 0.001$.

B. LSTM NEURAL NETWORK

Long short term memory (LSTM) neural network was proposed by Hochreiter and Schmidhuber [44] in 1997. It is a typical and efficient recurrent neural network structure, which can better deal with long-term dependent time series data and widely used in many practical fields. At present, it has become a very prevalent time series prediction model. Before introducing the LSTM neural network, we first introduce the recurrent neural network.

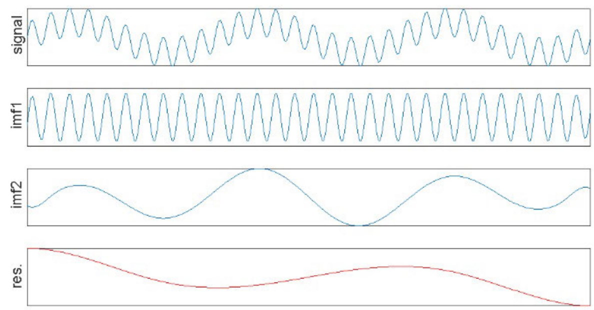


FIGURE 2. The process chart of EEMD decomposition of Sin mixing time series.

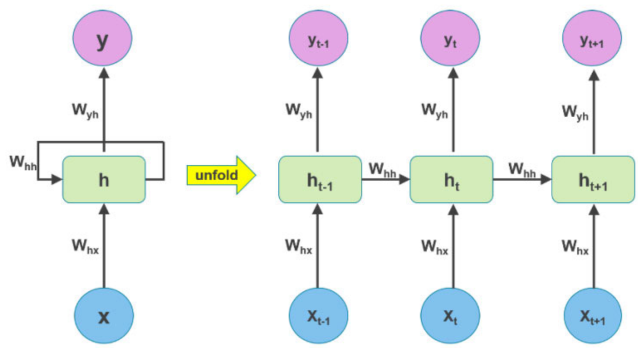


FIGURE 3. The basic structure of recurrent neural network.

The design of the recurrent neural network (RNN) is mainly used to deal with nonlinear time-varying problems. The internal connection of RNN allows data to be pushed forward and fed back. The design of feedback connection is very convenient for the algorithm to update the residual value or weight value in the forward step. This feature is very suitable for time-series prediction and can extract rules from the historical data of the time series to predict the future value of time series. The basic structure diagram of the recurrent neural network is shown in Fig. 3. In Fig. 3, the first column on the left is the overall structure of the RNN. The three columns on the right are an extension of the overall structure of the RNN. The RNN neural network module h reads an input data x_t and outputs a value y_t at time t. Compared with the traditional neural network, the parameters W_{hx}, W_{hh}, W_{yh} of each network layer in the RNN neural network are shared. Each input layer shares the network parameters W_{hx}, W_{hh}, W_{yh}. It is felt that every step of the RNN neural network is doing the same thing, but the input data x_t and output data y_t are different. In this way, the parameters of the RNN neural network need to learn are significantly reduced. After the RNN neural network is expanded, it becomes a multilayer neural network. In the traditional multilayer neural network,

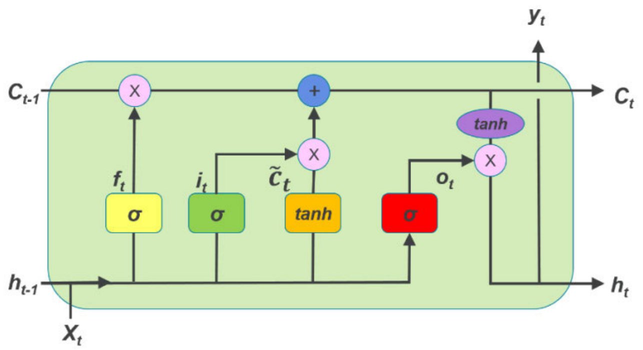


FIGURE 4. The basic structure of the LSTM neural network.

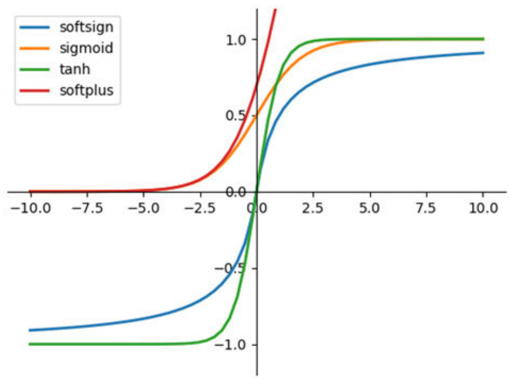


FIGURE 5. Four commonly used activation function curves.

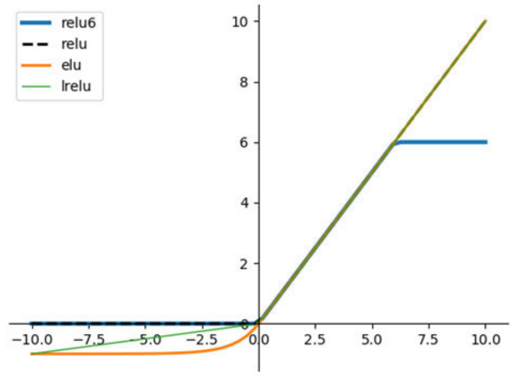


FIGURE 6. ReLU series activation function curve.

the parameters of each layer are different. For example, in the traditional multilayer neural network, the W_{hh} parameter between layer h_t -1 and layer h_t is different from that between layer h_t and layer h_t+1. In the RNN neural network, the parameters of each layer are the same. Similarly, the parameters of W_{hx} between layer x_t and layer h_t and W_{yh} between layer h_t and layer y_t are the same.

Each step of the RNN neural network has input and output, but the input and output of each step are not necessary. For example, when predicting the value of a mathematical expression, we only need to care about the output after the input of the last expression symbol, rather than knowing the output after the input of each expression symbol. The core of the RNN neural network is the hidden layer, which can capture the long and short term information in the time series.

However, the RNN neural network cannot solve the problem of gradient disappearance in network training. When the gradient disappears, the training time of the network increase infinitely, which eventually leads to network paralysis. Therefore, the simple RNN neural network is not the ideal choice to predict the long-term dependent time series.

The design of the LSTM neural network aims to overcome the problem of gradient disappearance when simple RNN neural network processes long-term dependent time series. Input gate, output gate, and forgetting gate are added to the network model of LSTM based on a simple RNN neural network. As shown in Fig. 4, there are three gate modules with σ coincidence in the box, which can effectively eliminate gradient. Therefore, LSTM neural network is very suitable for solving the problem of long-term dependence. The design of memory neurons is the most significant innovation of the LSTM neural network, which is used to store state information. Besides, generally, each door module needs to select an activation function to make nonlinear conversion or tradeoff for the information passing through the door module. For example, forgetting gate f_t is used to determine which state information of neurons to clear.

C. ACTIVATION FUNCTION

The activation function plays an essential role in the neural network. When the input data is weighted and summed, it also needs to be applied to a function, which is the activation function. The activation functions are all nonlinear. The activation function introduces nonlinear factors to the neural network, which can approach any nonlinear function arbitrarily so that the neural network can solve the problems that the linear model can not solve. If the activation function is not used, the output of each layer of the neural network is the linear combination of the input of the previous layer. If the activation function is not used, the output of the neural network is a linear combination of input, no matter how many layers the neural network has.

The commonly used activation functions are Sigmoid function, Tanh function, ReLU function, Softsign function, Softplus function, and Softmax function, etc. The curves of eight activation functions are shown in Fig. 5 and Fig. 6. Several typical activation function formulas are introduced here. The formulas of Sigmoid function, Tanh function, ReLU function, and Softmax function are shown in Eq.3, Eq.4, Eq.5, and Eq.6, respectively. Softmax function is mainly used in a multi-classification neural network, with slightly different formulas.

$$f\left(z\right) = \frac{1}{1+\exp(-z)} \tag{3}$$

$$f\left(z\right) = tanh\left(z\right) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}} \tag{4}$$

$$f(z) = \max(0, z) \tag{5}$$

$$f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{6}$$

In this paper, the ReLU function is selected as the activation function of each gate of the LSTM neural network. For example, the function of forgetting gate f_t is shown in formula Eq.7.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{7}$$

The workflow of the LSTM neural network is shown in Fig. 4, which is briefly described as follows, where W_i, W_f, W_c and W_o involved in each formula respectively represent the corresponding weight vector, b_i, b_f, b_c and b_o respectively represent the corresponding deviation vector.

(1) The external input data x_t flows into the LSTM neural network, connects the output data h_{t-1} of the upper layer to form the input data $x_t' = [h_{t-1}, x_t]$. The input data x_t' first flows into the forgetting gate f_t. After the activation function calculating by Eq.7, the output data of the forgetting gate f_t is obtained. The forgetting gate f_t can determine which state information of the neuron to clear.

(2) The input data x_t' flows into the input gate at the same time. After the activation function calculating by Eq.8, the output data i_t of the input gate is obtained. The input gate i_t determines which information needs to be updated and which information needs to be stored in the memory neuron. The function of the input gate i_t is shown in Eq.8.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{8}$$

(3) The input data x_t' flows into the *tanh* gate at the same time. After the activation function calculating by Eq.9, the output data $\tilde{c}_t$ of the *tanh* gate is obtained. $\tilde{c}_t$ is the candidate vector created by the *tanh* gate, which is used to update the state data of neurons. The function of the *tanh* gate is shown in Eq.9.

$$\tilde{c}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{9}$$

(4) The new state information c_t is obtained by updating the old state information c_{t-1} of the neuron. The new state information c_t is calculated by the function of Eq.10. The candidate vector $\tilde{c}_t$ determines how much state information to update.

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \tag{10}$$

(5) The input data x_t' flows into the output gate at the same time. After the activation function calculating by Eq.11, the output data o_t of the output gate is obtained. The function of the output gate o_t is shown in Eq.11.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{11}$$

(6) For the calculation of the output data y_t, the activation function is used to filter the state information of the neuron first, then input it to the *tanh* gate, and then multiply the output data o_t of the output gate to get the output data h_t, as shown in Eq.12.

$$y_t = h_t = o_t * tanh(c_t) \tag{12}$$

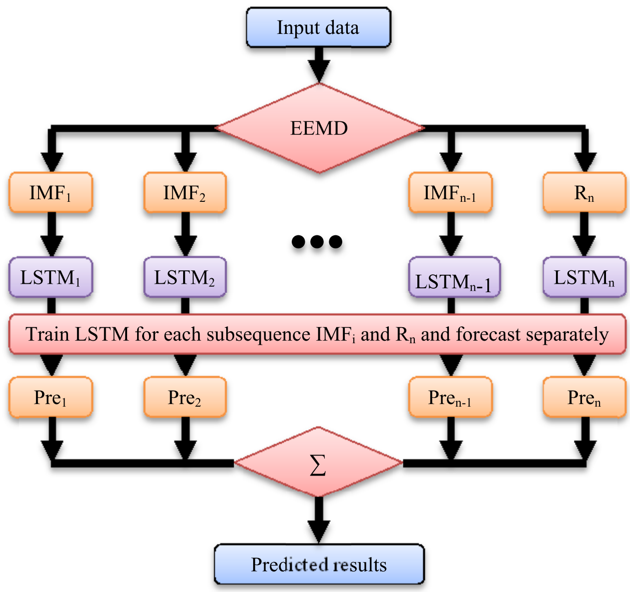


FIGURE 7. Model structure of LSTM-TFE.

III. METHODOLOGY

This chapter mainly introduces the theoretical basis of LSTM-TFE, LR-TFE, and BR-TFE prediction methods, EEMD, LSTM, LR, and BR methods firstly. Then the model structure of LSTM-TFE, LR-TFE, and BR-TFE prediction methods, which provides a solid theoretical basis for our proposed prediction method, be introduced. The proposed LSTM-TFE, LR-TFE, and BR-TFE methods use EEMD to decompose complex time series into several relatively simple subsequences firstly. LSTM neural network, LR, and BR are used to predict the value of each subsequence. Finally, the prediction results of multiple subsequences are summed to form the prediction results of the original complex time series.

The models of LSTM-TFE, LR-TFE, and BR-TFE are shown in Fig. 7, Fig. 8, and Fig. 9, respectively. The flowchart of the proposed prediction method is shown in Fig. 10. The model structure of the three methods is the same, but different machine learning methods are used in the model to train and predict the subsequence decomposed by the time series EEMD. The following is a brief introduction to the models of the three methods.

These models are divided into five steps: data collection and data preprocessing, EEMD decomposition to obtain multiple subsequences, one by one training machine learning model, and produce prediction results. Finally, the prediction results of multiple subsequences are summed to form the prediction results and evaluation model.

(1) Generate simulation data and collect real data in the real world, preprocess the original time series data, make its data format meet the format requirements of the EEMD decomposition algorithm, and form the input data X of EEMD decomposition algorithm.

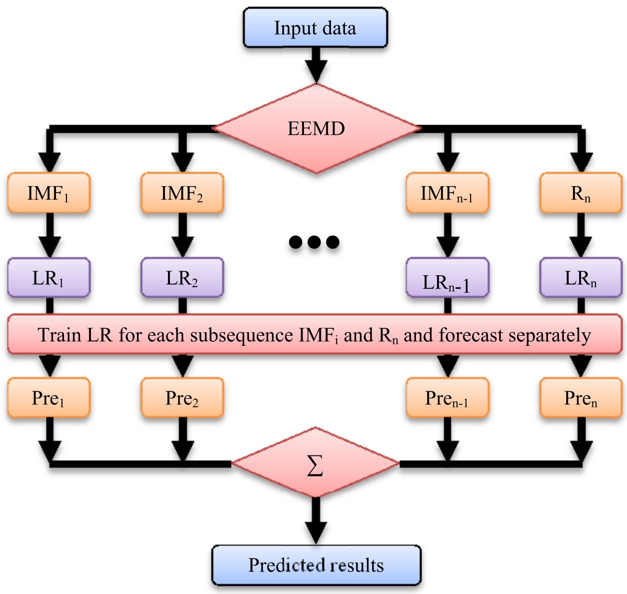


FIGURE 8. Model structure of LR-TFE.

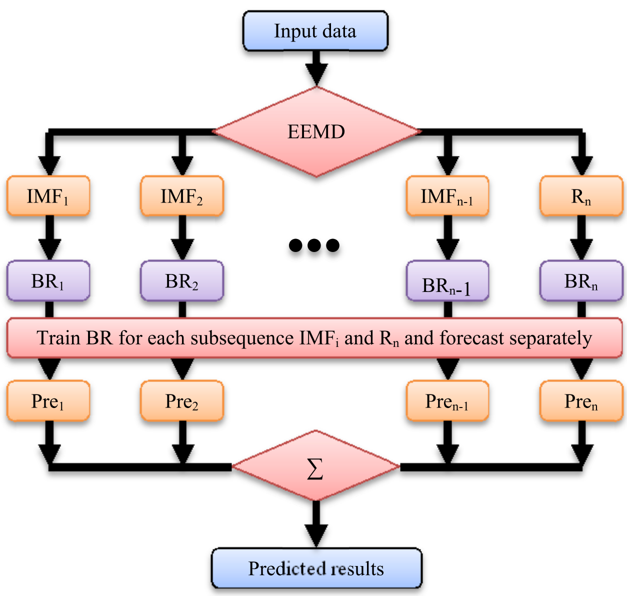


FIGURE 9. Model structure of BR-TFE.

(2) EEMD decomposition algorithm decomposes input data X to form n subsequences. According to the regulation of the EEMD decomposition time series, the last subsequence is generally called residual subsequence R. Therefore, n subsequences include $n-1$ IMF subsequence and a residual subsequence R, which are respectively expressed as IMF_1, IMF_2, IMF_3, …, IMF_{n-1}, R_n. Each subsequence data is divided into training data and test data. The training data is used to train the machine learning model, and the test data is used to test the trained model.

(3) Use the training data of each subsequence to train the machine learning model corresponding to the subsequence

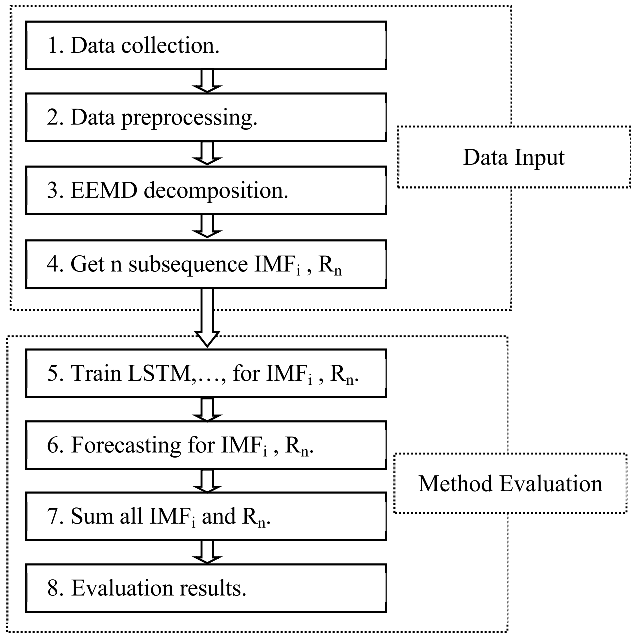


FIGURE 10. The flowchart of the proposed prediction method.

one by one. For example, train the LSTM machine learning model in the LSTM-TFE method, train the LR model in the LR-TFE method, and train the BR model in the BR-TFE method. Each training process is independent and does not affect each other. After the model is trained, form the model $LSTM_k$, LR_k, BR_k ($k = 1, 2, 3, \ldots, n-1, n$), using these trained models to predict the corresponding subsequence test data, and get the prediction results Pre_k ($k = 1, 2, 3, \ldots, n-1, n$).

(4) There are many fusion methods to fuse the prediction results of multiple subsequences to form the final prediction results. The fusion method used in this paper is summation, which is to accumulate the prediction results of each subsequence test data to form the final prediction results.

(5) Finally, comparing the prediction results with the actual results of the test data, three evaluation criteria are used to calculate the prediction errors RMSE, MAE and R^2, and evaluate the advantages and disadvantages of the model.

IV. EXPERIMENTS DATA

In this section, we mainly introduce the experimental data in our study. To test the prediction effect of the proposed method better, we used two kinds of experimental data in our study. The first one is the artificial simulation experimental data, which is generated automatically by a computer algorithm and is mainly used to verify the correctness and validity of the proposed method. In many literatures, the simulation data is used to test the proposed method. The second is the real stock indices data in the real world and the temperature time series in the meteorological field. Only using the proposed method to analyze the real social data has practical significance, and it is also the most effective test to apply the proposed prediction method to the actual field.

A. ARTIFICIAL SIMULATION EXPERIMENTAL DATA

To test the effectiveness of the proposed method, we used an artificial simulation experimental data in experiment. To obtain sufficient and effective experimental results, the artificial simulation experimental data generated manually should not be too short. Therefore, the length of the artificial simulation experimental data generated manually is 10000. The manually generated data is based on sin function, which is automatically generated by the computer program by Eq.13.

$$X = x(t) = sin(2\pi * 100t) + sin(2\pi * 10t) \quad (13)$$

where $t = 0, f, 2f, \ldots, 1000f; f = 0.0001$ in Eq.13.

B. REAL SOCIAL EXPERIMENTAL DATA

To test the practicability of the proposed method, we need to use the real data in the social field to test. In this paper, we use two real data in the real world, which are the temperature time series in the meteorological field and the SP500 stock index time series in the financial market. These two time-series have good representativeness. One time series in the field of Meteorology reflects the natural phenomenon without human intervention, and the temperature time series changes relatively smoothly. The time series of the SP500 stock index in the financial market reflects the social phenomenon, which is complicated, unstable, and is severely affected by human factors.

The temperature time series used in the experiment is the temperature data of an air quality data set, which is the weather and pollution conditions collected by the US embassy in Beijing within five years (from January 1, 2010 to December 31, 2014). The data set is recorded in hours, including PM2.5 data, weather information, date and time, weather information including dew point, temperature, pressure, wind direction, wind speed, and accumulated snowfall hours. The original data for this dataset can be downloaded from the UCI machine learning repository.

The SP500 stock index used in the experiment is the abbreviation of the American Standard & Poor's stock index. The SP500 index is one of the primary stock indexes in the American stock market, which has the significance of representing the American stock market. The original experimental data of the SP500 stock index can be obtained free of charge from Yahoo Finance (http://finance.yahoo.com). The data of each stock index includes five attributes: the opening price, the closing price, the highest price, the lowest price, and the trading volume. According to the results of the literature [45]–[46] analysis on SP500, the closing price of the SP500 stock index on that day is highly representative. Therefore, the experimental data selects the closing price time series of the SP500 stock index on that day, including all data from July 9, 2001 to December 24, 2015.

C. DATA PREPROCESSING

Given the above data are continuous numerical data, this paper uses these data to build a prediction problem. The prediction problem is to predict the value of the next sequence according to the value of 11 consecutive previous time series, that is, according to the 11 consecutive previous (hours or days) data of the above data, we can predict the value of the 12th (hours or days) data of time series. The length of these data is shown in Table 1. In Table 1, sin, temperature, and stock represent respectively the artificially generated sin simulation time series, the temperature time series of the actual air quality data set collected by the US embassy in Beijing, and the SP500 stock index time series of the financial market.

TABLE 1. Length of data used in the experiment.

Category	Sin	Temperature	SP500
Total length of selected original data	10000	10000	3761
Total length of experimental data	9988	9988	3749
Training data length	5992	5992	2249
Test data length	3996	3996	1500

To achieve better experimental results of our proposed prediction method, we do our best to let the experimental data should be as much as possible and remove all the noise data of the experimental data as far as we could. Because there are some incorrect data in the original experimental data sets of the four indices, such as the records of two consecutive days with exactly equal data, the record of the transaction volume being equal to zero, we firstly should remove all noise or incorrect data in the experimental data sets of the four stock indices.

Because the prediction problem is to predict the value of the next sequence according to the value of the 11 consecutive previous sequences of the time series, it needs to preprocess the original time series data. The original time series is stored in a one-dimensional array. The experiment needs to process the original one-dimensional data into a two-dimensional array with 12 data as a group (row), which causes the first part of the data to be unusable. It causes the total length of selected original data (the first row in Table 1) is 12 longer than the total length of experimental data (the second row in Table 1). To reduce the operation time, we selected the first 10000 pieces of original data in the experiment because the length of the temperature time series is very long. The length of the generated sin simulation time series is also 10000. The actual length of the SP500 stock index time series is 3761. Therefore, the first row of Table 1 shows the "total length of selected original data" of the three experimental data, which is respectively 10000, 10000, and 3761. The total length of the experimental data after preprocessing is shown in the second row of Table 1, which is 9988, 9988, and 3749, respectively. When training the machine learning model, we need to divide the data into training data and test data. In the experiment, the ratio of 6:4 is used to segment the data. 60% of the total length of the experimental data is used as the training data, and the rest is used as the test data. The specific length of training data and test data of each time series is shown in the third and fourth rows of Table 1, respectively.

V. EXPERIMENTS

The experimental data used in the experiment come from two fields: one is the artificial simulation experimental data, the others are the real data in the real world. Besides, to compare with other methods conveniently, we selected seven prediction methods to test the same experimental data. Therefore, this section explains and analyzes the experimental results from three aspects.

A. EVALUATION CRITERION

There are many evaluation criteria for the evaluation model, most of which are targeted. The typical problems are clustering, regression, classification, ranking, double clustering, and pairing. To evaluate the proposed method correctly and effectively, we selected the evaluation criterion of RMSE, R^2, and MAE to calculate the prediction error and evaluate the model.

Root mean square error (RMSE), also known as standard error, is a widely used evaluation index, which is very sensitive to extremely small or significant errors in test data. Its calculation formula is shown in Eq.14.

$$RMSE\left(y, \tilde{y}\right) = \sqrt{\frac{1}{n}\sum\nolimits_{k=1}^{n}\left(y_k - \tilde{y_k}\right)^2} \tag{14}$$

where y and $\tilde{y}$ in Eq.14 represent the real value sequence and predicted value sequence of test data respectively, y_k and $\tilde{y_k}$ respectively represent the real value and predicted value of the k th in the test data sequence, and n represents the length of test data.

R square (R^2) is the ratio of the sum of regression squares to the sum of total deviation squares. The larger the ratio is, the better the effect of the model is, and the more accurate the effect is. At present, it is an indicator used to measure the prediction ability of the model. The best score is 1. The closer it is to 1, the better the prediction effect of the model is. In general, the model over 0.9 is better, and its value may be negative, indicating that the model is inferior, not as good as the average prediction. The calculation formula is shown in Eq.15.

$$R^2\left(y, \tilde{y}\right) = 1 - \frac{\sum_{k=1}^{n}\left|y_k - \tilde{y_k}\right|}{\sum_{k=1}^{n}\left|y_k - \overline{\overline{y_k}}\right|} \tag{15}$$

In Eq.15, y, $\overline{\overline{y_k}}$, and $\tilde{y}$ respectively represent the true value, average value, and predicted value of test data, n represents the length of test data, and the calculation formula of average value $\overline{\overline{y_k}}$ is shown in Eq.16.

$$\overline{\overline{y_k}} = \frac{1}{n}\sum\nolimits_{k=1}^{n} y_k \tag{16}$$

Mean absolute error (MAE) is the average value of absolute error, which can better reflect the actual situation of predicted value error, and its calculation formula is shown in Eq. 17.

$$MAE\left(y, \tilde{y}\right) = \frac{1}{n}\sum\nolimits_{k=1}^{n}\left|y_k - \tilde{y_k}\right| \tag{17}$$

where y and $\tilde{y}$ in Eq.17 respectively represent the real value and predicted value of test data, and n represents the length of test data.

TABLE 2. Prediction results of three methods for three time-series.

METHOD		Sin	Temperature	SP500
SVR	RMSE	0.130662	1.947225	425.013873
	MAE	0.117780	1.442153	335.804235
	R2	0.981994	0.960849	-0.558416
LR	RMSE	**0.000035**	**1.392644**	**14.172792**
	MAE	**0.000021**	**0.978153**	**10.186058**
	R2	**1.000000**	**0.979974**	**0.998267**
RFR	RMSE	0.005341	1.464543	270.957771
	MAE	0.003571	1.058784	174.903846
	R2	0.999970	0.977853	0.366596
KR	RMSE	0.838632	8.860991	599.048490
	MAE	0.700675	7.788972	464.725006
	R2	0.258231	0.189280	-2.096002
BR	RMSE	**0.000035**	**1.392694**	**14.177517**
	MAE	**0.000021**	**0.978235**	**10.194094**
	R2	**1.000000**	**0.979973**	**0.998266**
KNR	RMSE	0.011126	1.885286	278.261273
	MAE	0.005177	1.419670	182.983170
	R2	0.999869	0.963300	0.331990

B. ANALYSIS OF EXPERIMENTAL RESULTS OF OTHER METHODS

To compare the effects and characteristics of various prediction methods, we selected seven prediction methods to predict the same experimental data. The experimental results are shown in Table 2 and Table 3. The LSTM method is described in detail above, and the other six methods are briefly introduced below.

1) SVR

SVR is short for Support Vector Regression, which is the most basic support vector regression method. According to libsvm, machine learning is controlled by penalty function and loss function. In the experiment, the penalty parameter and loss parameter are 1 and 0.2, respectively.

2) LR

LR is short for linear regression, which is a standard least square regression method. It is widely used at present.

3) RFR

RFR is short for random forest regression. A random tree is a basic estimate. A large number of classification decision

TABLE 3. RMSE evaluation of prediction results of LSTM and LSTM-TFE.

Method	Iteration times	Sin	Temperature	SP500
LSTM	50	0.650849	8.455165	1414.209806
	500	0.042390	1.545091	559.593267
	5000	0.038733	1.682006	559.588074
Proposed LSTM-TFE	50	**0.389935**	**3.030954**	**1359.775809**
	500	**0.038895**	**0.922435**	**478.802265**
	5000	**0.020879**	**0.738811**	**489.776767**

trees are used to fit the subsamples of the data set. Finally, it is averaged to improve prediction accuracy and control overfitting.

4) KR

KR is short for kernel ridge regression, combined with the skills of ridge regression and kernel, adopts the linear least square L2 norm regularization. In the data space of the linear kernel, it corresponds to the linear function in the original space. For the nonlinear kernel, it corresponds to the nonlinear function in the original space.

5) BR

BR is short for Bayesian ridge regression, which uses the regularized parameters λ (accuracy of weight) and α (accuracy of noise) to fit the model.

6) KNR

KNR is short for K-nearest Neighbors Regression, which is a kind of regression method based on the k-nearest neighbor algorithm. It can predict the target by local interpolation of the target related to the nearest neighbor in the training set.

Table 2 shows the prediction results of six methods of three-time series: sin, temperature, and stock. The data used by all methods are the data preprocessed by these three time-series. The specific length of the data in the experiment is shown in the second row of Table 1. The prediction problem is to predict the value of the next time series according to the value of 11 consecutive previous time series. The experimental results in Table 2 are calculated by RMSE, MAE, and R^2 based on the predicted and real results. The smaller the results of RMSE and MAE, the better the prediction effect. However, the larger the result of R^2 evaluation index, the better the prediction effect of the method.

For the convenience of comparison, the results of the two best prediction methods for each time series data are shown in bold, as shown in Table 2. It is found that LR and BR are the best methods to predict sin time-series data. The LR and Br methods are the best methods to predict the time series data of temperature and SP500. Therefore, LR and BR are the best methods to predict the three time-series data.

Through careful observation of Table 2, it is found that for the prediction of sin time series data, except the KR method,

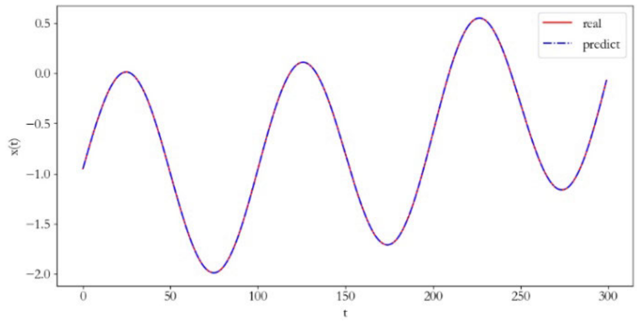


FIGURE 11. Prediction results of BR method for sin time series.

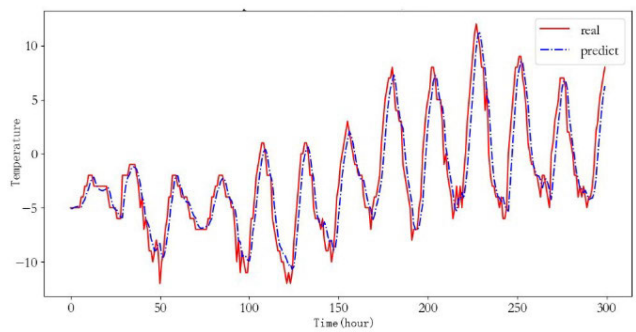


FIGURE 12. Prediction results of SVR method for temperature time series.

the other five methods have a good prediction effect. The R^2 evaluation index of the other five methods is more than 0.98, which shows that the prediction effect of these methods is better for the time series with regular and stable changes. Here, we only select the result of one method to display the prediction effect in the figure, as shown in Fig. 11. It is found that the overall prediction effect of the KR method inferior. To make the result graphs clear and distinguishable, all the result graphs in this section only show the last 300 predicted results, unless otherwise specified.

Observing the SVR experimental results in Table 2, it is found that the prediction effect of this method for the time series data of temperature and sin is better than that of others. Figure 12 shows the SVR prediction result of the time series data of temperature, but the prediction effect for the time series data of stock is inferior. The change of the time series of temperature and sin is more regular and stable, which shows that the SVR method is more suitable for regular and stable time series prediction. Although it belongs to the category of support vector regression, many improvements have been made on the traditional SVR method. It is more flexible in the selection of penalty function and loss function, and it is suitable for the situation with a more significant sample number. From the experimental results of SVR in Table 2, the prediction effect of the three time-series data is not the best.

By observing the experimental results of RFR and KNR in Table 2, it is found that the prediction effect of this

TABLE 4. Prediction results of LR, BR and the proposed method for three time-series.

METHOD		Sin	Temperature	SP500
LR	RMSE	0.000035	1.392644	14.172792
	MAE	0.000021	0.978153	10.186058
	R2	1.000000	0.979974	0.998267
Proposed LR-TFE	RMSE	**0.000027**	**0.713150**	**7.302429**
	MAE	0.000021	**0.512700**	**5.307596**
	R2	1.000000	**0.994749**	**0.999540**
BR	RMSE	0.000035	1.392694	14.177517
	MAE	**0.000021**	0.978235	10.194094
	R2	1.000000	0.979973	0.998266
Proposed BR-TFE	RMSE	**0.000033**	**0.712299**	**7.295944**
	MAE	0.000027	**0.512609**	**5.310422**
	R2	**1.000000**	**0.994761**	**0.999541**

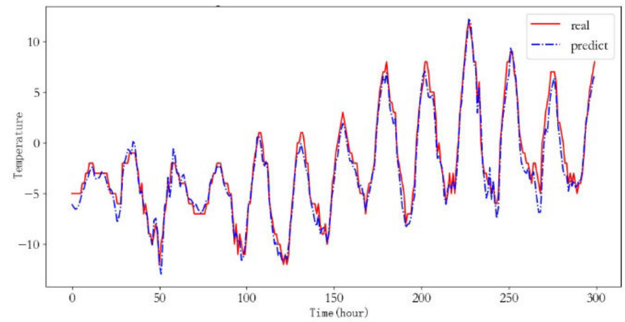
method for temperature and sin time series data is better than that of others, but the prediction effect for stock time series data is inferior. Human factors greatly influence the stock time series. Since the data composition, which changes with the most irregular and unstable, is complicated, it is not suitable to use the KNR method to predict the stock time series. The experimental results of RFR and KNR show that RFR and KNR are not suitable for time series prediction with irregular and unstable changes. LR and BR methods can achieve better results for the stock time series. Besides, the other methods are not suitable for the prediction of financial markets, which are greatly influenced by human factors.

Observing Table 3, it is found that the prediction effect of the LSTM method on stock time series data is inferior. The prediction effect of temperature and sin time series data is slightly better. With the rapid development of deep learning algorithms, LSTM should achieve better results, but the results obtained in the experiment are not good, which may be related to the LSTM model constructed in the experiment, the parameters set, and the number of iterations during training. It is concluded that the main reason for such experimental results is that human factors greatly influence the stock time series, and its changes are the most irregular and unstable.

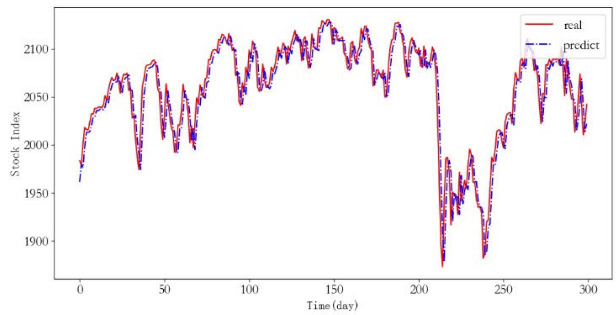
C. ANALYSIS OF EXPERIMENTAL RESULTS BASED ON ARTIFICIAL SIMULATION DATA

To test the effectiveness of the proposed method, we used an artificial simulation experimental data. To obtain sufficient and effective experimental results, the artificial simulation experimental data generated manually should not be too short. Therefore, the length of the generated sin simulation time series is 10000.

Before discussing the experimental results of real data in the actual field, the proposed LSTM-TFE, LR-TFE, and



(a) Predict and real value of temperature



(b) Predict and real value of the stock index

FIGURE 13. Prediction results of LSTM-TFE method for temperature time series and stock time series.

BR-TFE methods are used to predict the Sin artificial simulation time series data. The artificial simulation experiments can verify the correctness and effectiveness of the three methods.

Observe the sin data column in Table 3, the RMSE evaluation index of the LSTM-TFE method is better than that of the LSTM method under the same iteration times. Observe the sin data column in Table 4, the evaluation indexes of the LR-TFE method are equal to or better than those of the LR method. The RMSE and R^2 of the BR-TFE method are equal to or better than those of the BR method. Although the MAE evaluation index of the BR-TFE method is slightly worse than that of the BR method, it is almost negligible. In conclusion, the proposed LSTM-TFE, LR-TFE, and BR-TFE methods have a better overall effect than the original methods, which shows that the three prediction methods are correct and effective.

D. ANALYSIS OF EXPERIMENTAL RESULTS BASED ON REAL DATA

To test the practicability of the proposed LSTM-TFE, LR-TFE, and BR-TFE prediction methods, we used three methods to predict the temperature time series in the meteorological field and a real stock index time series in the financial market. These two real-time-series can be divided into two categories: temperature time series belong to natural phenomenon, while stock index time series belong to social

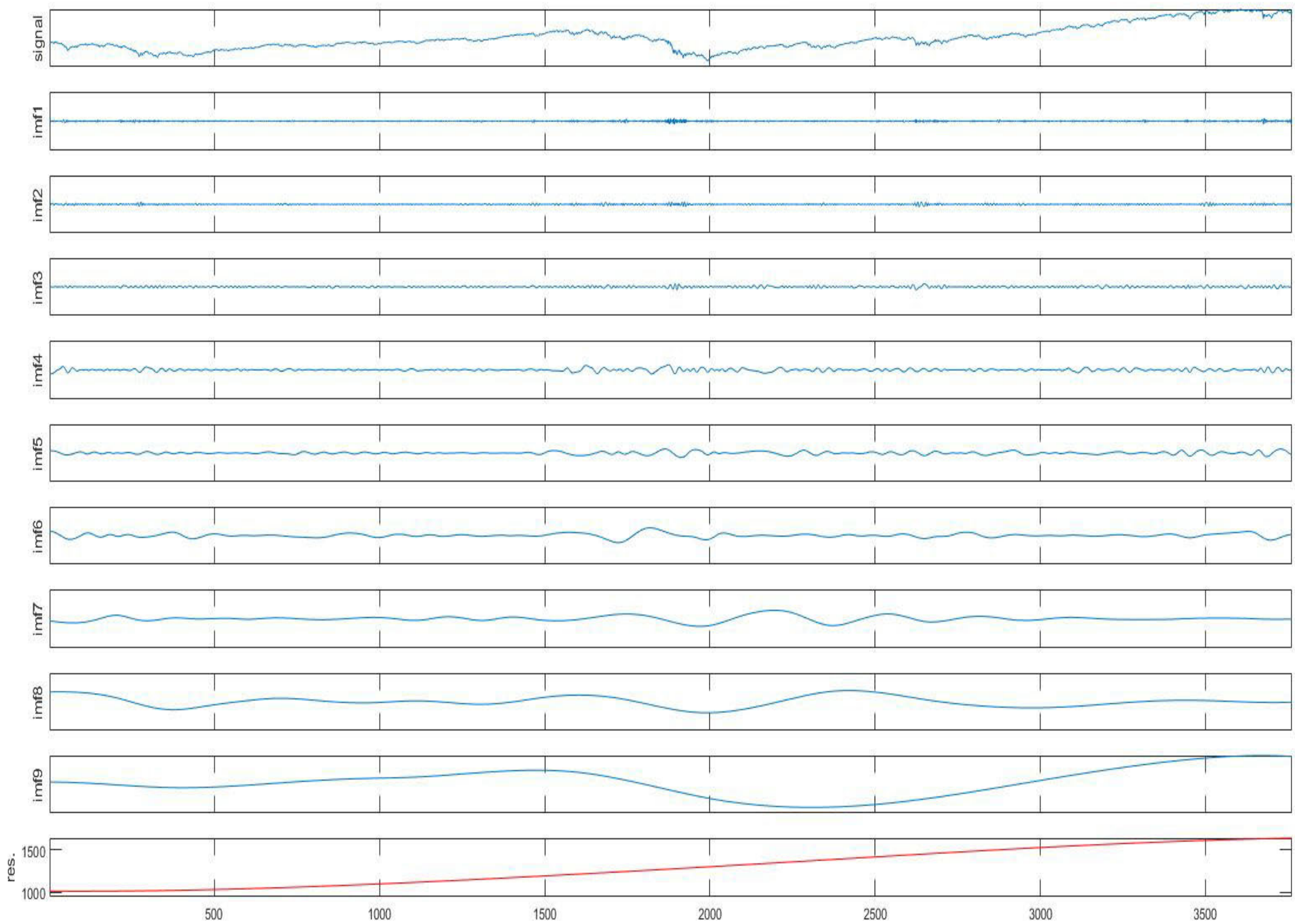


FIGURE 14. SP500 stock time-series change and EEMD decomposition.

phenomenon. These two time-series have good representativeness. The temperature time series are not interfered with by a human, but the change of temperature time series is relatively stable. The time series of stock index in the financial market is complex and unstable, which is severely affected by human factors.

Observing RMSE evaluation indexes of LSTM and LSTM-TFE prediction results of temperature and stock time series in Table 3, it is found that the RMSE evaluation index of LSTM-TFE method is better than that of LSTM method in the time series of temperature and stock. Fig. 13 shows the LSTM-TFE prediction results of the temperature time series.

By observing Table 3 and Table 4, it is found that the results of the proposed LSTM-TFE, LR-TFE and BR-TFE methods on the two time series of temperature and stock are much better than those of the original LSTM, LR and BR methods. There are two main reasons for such good results. On the one hand, the three methods proposed in this paper decompose the time series of temperature and stock by EEMD, which makes the complex original time series decompose into more regular and more stable subsequences, and the prediction results of subsequences are more accurate than that of the original sequence. On the other hand, although the changes in temperature and stock time series are complex, they can be decomposed into a regular and stable time series. To clearly understand the change of temperature and stock time series and EEMD decomposition, Fig. 14 and Fig. 15 show all the changes of these two time-series and EEMD decomposition, respectively. The first subgraph at the top of Fig. 14 and Fig. 15 is the original change of the two time-series, and the other subgraphs are the EEMD decomposition subgraphs of the two time-series. It is easy to see that the lower EEMD decomposition subgraph is milder, more regular, more stable, easier to predict, and the better prediction effect.

In this paper, we put forward three prediction methods based on EEMD. The experimental results show that the fusing predicts results is better than that of using the existing prediction methods to predict the original time series directly. Although we choose three classical methods in this paper, such as BR, LR, and LSTM, when someone applies them in different fields, they can choose one to try in a specific field. In application, if the effect of the selected method is

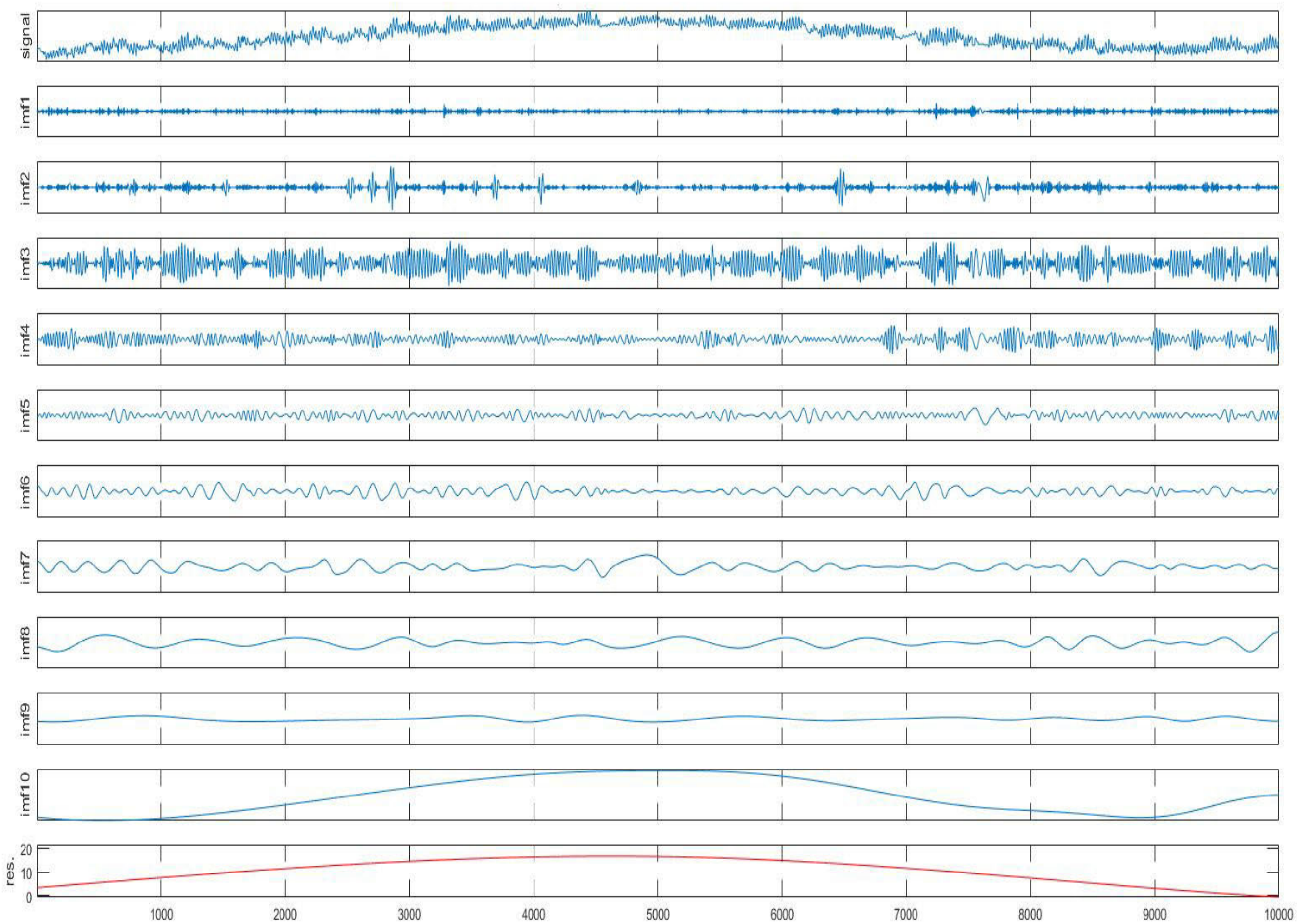


FIGURE 15. Temperature time-series change and EEMD decomposition.

not as expected, the other two methods can be selected one by one.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed three hybrid methods for short-term time series forecasting based on EEMD, which are LSTM-TFE, LR-TFE, and BR-TFE. These methods are all based on the idea of divide and conquer of complex problems and the strategy of machine learning optimization step by step, combined with EEMD, Long-term and Short-Term Memory neural network, Linear Regression and Bayesian Ridge Regression. EEMD is used by LSTM-TFE, LR-TFE, and BR-TFE to decompose complex time series into several relatively milder, more regular, and stable subsequences. Using the LSTM neural network, LR, and BR methods, each subsequence is trained and predicted by machine learning. The value of each subsequence is predicted firstly. The prediction results of multiple subsequences are fused to form the prediction results of the original complex time series. To verify the proposed method comprehensively, we select three representative time series data to test in this paper. From the experimental results, it can be found that the proposed method has an excellent overall effect. But there are also some defects, and the prediction effect of time series with more drastic changes is not good.

The research of time series analysis and prediction method has developed rapidly, but its effect can't meet the high requirements of practical application in some fields in many aspects, and there are many problems to be solved. Based on the research work in this paper, there are still many contents to study in the future. In the future, the EEMD method needs to be improved or combined with other methods, to solve the problem of poor prediction effect of such time series. We will continue to study the application of deep learning algorithm in time series prediction, try to train the model with the help of high-performance computer, increase the number of training iterations, and study the construction of deep learning model for different time series, to provide an important reference for the application in the actual field. To improve the decomposition effect of time series, we will make a comparative study on various time series decomposition methods, such as wavelet decomposition, VMD, EEMD, etc., and find their advantages and disadvantages and limitations in use, so that they can be normally applied in different fields.

REFERENCES

[1] W.-Y. Lin, Y.-H. Hu, and C.-F. Tsai, "Machine learning in financial crisis prediction: A survey," *IEEE Trans. Syst., Man, Cybern., C (Appl. Rev.)*, vol. 42, no. 4, pp. 421–436, Jul. 2012.

[2] M. Q. Raza and A. Khosravi, "A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings," *Renew. Sustain. Energy Rev.*, vol. 50, pp. 1352–1372, Oct. 2015.

[3] N. Meade and T. Islam, "Forecasting in telecommunications and ICT—A review," *Int. J. Forecasting*, vol. 31, no. 4, pp. 1105–1126, Oct. 2015.

[4] I. Maqsood, M. Khan, and A. Abraham, "An ensemble of neural networks for weather forecasting," *Neural Comput. Appl.*, vol. 13, no. 2, pp. 112–122, Jun. 2004.

[5] J. Reyes, A. Morales-Esteban, and F. Martínez-Álvarez, "Neural networks to predict earthquakes in chile," *Appl. Soft Comput.*, vol. 13, no. 2, pp. 1314–1328, Feb. 2013.

[6] C. C. Holt, "Forecasting seasonals and trends by exponentially weighted moving averages," *Int. J. Forecasting*, vol. 20, no. 1, pp. 5–10, Jan. 2004.

[7] S. M. Tabatabaie Nezhad, M. Nazari, and E. A. Gharavol, "A novel DoS and DDoS attacks detection algorithm using ARIMA time series model and chaotic system in computer networks," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 700–703, Apr. 2016.

[8] S.-J. Kim, C.-H. Kim, S.-Y. Jung, and Y.-J. Kim, "Shape optimization of a hybrid magnetic torque converter using the multiple linear regression analysis," *IEEE Trans. Magn.*, vol. 52, no. 3, Mar. 2016, 8102504.

[9] M. I. Saripan, W. H. Mohd Saad, S. Hashim, A. T. A. Rahman, K. Wells, and D. A. Bradley, "Analysis of photon scattering trends for material classification using artificial neural network models," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 2, pp. 515–519, Apr. 2013.

[10] C. Li, J. Yang, Y. Xu, Y. Wu, and P. Wei, "Classification of voltage sag disturbance sources using fuzzy comprehensive evaluation method," *CIRED-Open Access Proc. J.*, vol. 2017, no. 1, pp. 544–548, Oct. 2017.

[11] X. Wei, X. Luo, Q. Li, J. Zhang, and Z. Xu, "Online comment-based hotel quality automatic assessment using improved fuzzy comprehensive evaluation and fuzzy cognitive map," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 1, pp. 72–84, Feb. 2015.

[12] K. Zhou, Z. Li, W. Gong, S. Zhao, C. Wen, and Y. Song, "Influence of magnetic field generated by air core reactors in SVC-based substation and an optimal suppression method based on fuzzy comprehensive evaluation," *IEEE Trans. Electromagn. Compat.*, early access, Oct. 8, 2019, doi: 10.1109/TEMC.2019.2942435.

[13] M. P. Hobson, A. W. Jones, and A. N. Lasenby, "Wavelet analysis and the detection of non-gaussianity in the cosmic microwave background," *Monthly Notices Roy. Astronomical Soc.*, vol. 309, no. 1, pp. 125–140, Oct. 1999.

[14] J. Gruber, Z. Sekeresova, J. Hlina, and J. Sonsky, "Analysis of thermal plasma dynamics in 3-D using tomographical reconstruction and wavelet analysis," *IEEE Trans. Plasma Sci.*, vol. 39, no. 11, pp. 2850–2851, Nov. 2011.

[15] H. Kim and H. Ling, "Wavelet analysis of electromagnetic backscatter data," *Electron. Lett.*, vol. 28, no. 3, p. 279, 1992.

[16] F. Li, B. Wu, N. Liu, Y. Hu, and H. Wu, "Seismic Time–Frequency analysis via adaptive mode separation-based wavelet transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 696–700, Apr. 2020.

[17] H. Zhanlin, Z. Chen, and W. Jingning, "A new multi-node cooperative interference detection and recognition algorithm based on svm," in *Proc. Int. Conf. Inf. Commun. Technol. (ICT)*, Nanjing, China, 2014, pp. 1–5.

[18] M. H. Selamat and H. Md Rais, "Image face recognition using hybrid multiclass SVM (HM-SVM)," in *Proc. Int. Conf. Comput., Control, Informat. Appl. (ICINA)*, Oct. 2015, pp. 159–164.

[19] X. Wu, W. Zuo, L. Lin, W. Jia, and D. Zhang, "F-SVM: Combination of feature transformation and SVM learning via convex relaxation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5185–5199, Nov. 2018.

[20] T. Hee Oong and N. A. M. Isa, "Adaptive evolutionary artificial neural networks for pattern classification," *IEEE Trans. Neural Netw.*, vol. 22, no. 11, pp. 1823–1836, Nov. 2011.

[21] M. Ilkucar, A. H. Isik, and A. Cifci, "Classification of breast cancer data with harmony search and back propagation based artificial neural network," in *Proc. 22nd Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2014, pp. 762–765.

[22] J. Kaur and A. Kalra, "Hybrid artificial neural network for data classification problem," in *Proc. 4th Int. Conf. Signal Process., Comput. Control (ISPCC)*, Sep. 2017, pp. 66–71.

[23] X. Qin, T. Hu, H. Zou, W. Yu, and P. Wang, "Polsar image classification via complex-valued convolutional neural network combining measured data and artificial features," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 3209–3212.

[24] H. Madani, N. Ouerdi, A. Palisse, J.-L. Lanet, and A. Azizi, "Classification of ransomwaresusing artificial neural networks and Bayesian networks," in *Proc. 3rd Int. Conf. Intell. Comput. Data Sci. (ICDS)*, Marrakech, Morocco, Oct. 2019, pp. 1–6.

[25] A. Upadhyay, A. Upadhyay, and S. Maurya, "Regression and artificial neural network based improved classification of LISS-III satellite image," in *Proc. Int. Conf. Current Trends Comput., Electr., Electron. Commun. (CTCEEC)*, Sep. 2017, pp. 917–921.

[26] A. E. Tumer and S. Edebali, "Prediction of wastewater treatment plant performance using multilinear regression and artificial neural networks," in *Proc. Int. Symp. Innov. Intell. Syst. Appl. (INISTA)*, Sep. 2015, pp. 1–5.

[27] R. Elkadiri, M. Sultan, A. M. Youssef, T. Elbayoumi, R. Chase, A. B. Bulkhi, and M. M. Al-Katheeri, "A remote sensing-based approach for debris-flow susceptibility assessment using artificial neural networks and logistic regression modeling," *IEEE J. Sel. Topics Appl. Earth Observat., Remote Sens.*, vol. 7, no. 12, pp. 4818–4835, Dec. 2014.

[28] A. Y. Alanis, "Electricity prices forecasting using artificial neural networks," *IEEE Latin Amer. Trans.*, vol. 16, no. 1, pp. 105–111, Jan. 2018.

[29] P. Musikawan, K. Sunat, Y. Kongsorot, P. Horata, and S. Chiewchanwattana, "Parallelized Metaheuristic-ensemble of heterogeneous feedforward neural networks for regression problems," *IEEE Access*, vol. 7, pp. 26909–26932, 2019.

[30] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. London. A, Math., Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, Mar. 1998.

[31] J. Cho and M. Lee, "Estimation of user-indoor spatial information using deep neural networks selective ventilation for living area estimated by deep neural network," in *Proc. IEEE 8th Int. Conf. Consum. Electron. (ICCE-Berlin)*, Berlin, Germany, Sep. 2018, pp. 1–4.

[32] S. Sotirov, "A generalized net model of the deep learning neural network," in *Proc. Adv. Neural Netw. Appl.*, St. Konstantin Elena Resort, Bulgaria, 2018, pp. 1–4.

[33] E. Nishani and B. Cico, "Computer vision approaches based on deep learning and neural networks: Deep neural networks for video analysis of human pose estimation," in *Proc. 6th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2017, pp. 1–4.

[34] S. Tezuka, G. Maeda, M. Baba, and N. Baba, "PM-05 Applications of a particle extraction method with deep neural networks using improved autoencoders to biological ultra-thin section images," *Microscopy*, vol. 65, no. 1, p. i34, Nov. 2016.

[35] S. Pathak, X. Cai, and S. Rajasekaran, "Ensemble deep TimeNet: An ensemble learning approach with deep neural networks for time series," in *Proc. IEEE 8th Int. Conf. Comput. Adv. Bio Med. Sci. (ICCABS)*, Oct. 2018.

[36] J. Townsend, T. Chaton, and J. M. Monteiro, "Extracting relational explanations from deep neural networks: A survey from a neural-symbolic perspective," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2019.

[37] L. Liu, "Human face expression recognition based on deep learning-deep convolutional neural network," in *Proc. Int. Conf. Smart Grid Electr. Autom. (ICSGEA)*, Aug. 2019, pp. 221–224.

[38] J. Zhang, C. Lee, C. Liu, Y. S. Shao, S. W. Keckler, and Z. Zhang, "SNAP: A 1.67—21.55 TOPS/W sparse neural acceleration processor for unstructured sparse deep neural network inference in 16nm CMOS," in *Proc. Symp. VLSI Circuits*, Kyoto, Japan, Jun. 2019, pp. C306–C307.

[39] C.-Y. Low, J. Park, and A. B.-J. Teoh, "Stacking-based deep neural network: Deep analytic network for pattern classification," *IEEE Trans. Cybern.*, early access, Apr. 22, 2019, doi: 10.1109/TCYB.2019.2908387.

[40] R. Bhowmick, M. I. Sadek Bhuiyan, M. Sabir Hossain, M. K. Hossen, and A. Sadee Tanim, "An approach for improving complexity of longest common subsequence problems using queue and Divide-and-Conquer method," in *Proc. 1st Int. Conf. Adv. Sci., Eng. Robot. Technol. (ICASERT)*, Dhaka, Bangladesh, May 2019, pp. 1–5.

[41] Y. Pan, R. Xia, J. Yin, and N. Liu, "A Divide-and-Conquer method for scalable robust multitask learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3163–3175, Dec. 2015.

[42] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: A noise-assisted data analysis method," *Adv. Adapt. Data Anal.*, vol. 1, no. 1, pp. 1–41, Jan. 2009.

[43] Y. Xu, C. Zhou, J. Geng, S. Gao, and P. Wang, "A method for diagnosing mechanical faults of on-load tap changer based on ensemble empirical mode decomposition, volterra model and decision acyclic graph support vector machine," *IEEE Access*, vol. 7, pp. 84803–84816, 2019.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[45] Y. Yang, J. Li, and Y. Yang, "The cross-correlation analysis of multi property of stock markets based on MM-DFA," *Phys. A, Stat. Mech. Appl.*, vol. 481, pp. 23–33, Sep. 2017.

[46] Y. Yujun, L. Jianping, and Y. Yimei, "Multiscale multifractal multiproperty analysis of financial time series based on Rényi entropy," *Int. J. Mod. Phys. C*, vol. 28, no. 02, Feb. 2017, Art. no. 1750028.

[47] T. Zhongda, L. Shujiang, and W. Yanhong, "A prediction approach using ensemble empirical mode decomposition-permutation entropy and regularized extreme learning machine for short-term wind speed," *Wind Energy*, vol. 23, no. 2, pp. 177–206, 2020.

[48] T. Zhongda, R. Yi, and W. Gang, "Short-term wind power prediction based on empirical mode decomposition and extreme learning machine," *J. Elect. Eng. Technol.*, vol. 13, no. 5, pp. 1841–1851, 2018.

YUJUN YANG received the B.S. and M.S. degrees in computer science and technology from the Central South University of Forestry Science and Technology, Changsha, China, in 2002 and 2007, respectively, and the Ph.D. degree in computer science and technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2018.

He is currently an Associate Professor with the School of Computer Science and Engineering, Huaihua University, a Senior Research Professional with the Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, and a Research Assistant with the Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province. His research interests include machine learning, time series analysis, financial data mining, prediction and recommendation, and artificial intelligence techniques.

YIMEI YANG received the M.S. degree in computer science and technology from the Central South University of Forestry Science and Technology, Changsha, China, in 2007.

She is currently an Associate Professor with the School of Computer Science and Engineering, Huaihua University, a Senior Research Professional with the Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, and a Research Assistant with the Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province. Her research interests include machine learning, time series analysis, financial data mining, prediction and recommendation, and artificial intelligence techniques.