

Research Article

A Hybrid Prediction Method for Stock Price Using LSTM and Ensemble EMD

Yang Yujun ^{1,2,3}, Yang Yimei ^{1,2} and Xiao Jianhua^{1,2}

¹School of Computer Science and Engineering, Huaihua University, Huaihua 418008, China

²Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province, Huaihua 418000, China

³Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, Huaihua 418000, China

Correspondence should be addressed to Yang Yimei; yym1630@163.com

Received 24 July 2020; Revised 13 September 2020; Accepted 21 November 2020; Published 4 December 2020

Academic Editor: Cheng Lu

Copyright © 2020 Yang Yujun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The stock market is a chaotic, complex, and dynamic financial market. The prediction of future stock prices is a concern and controversial research issue for researchers. More and more analysis and prediction methods are proposed by researchers. We proposed a hybrid method for the prediction of future stock prices using LSTM and ensemble EMD in this paper. We use comprehensive EMD to decompose the complex original stock price time series into several subsequences which are smoother, more regular and stable than the original time series. Then, we use the LSTM method to train and predict each subsequence. Finally, we obtained the prediction values of the original stock price time series by fused the prediction values of several subsequences. In the experiment, we selected five data to fully test the performance of the method. The comparison results with the other four prediction methods show that the predicted values show higher accuracy. The hybrid prediction method we proposed is effective and accurate in future stock price prediction. Hence, the hybrid prediction method has practical application and reference value.

1. Introduction

According to the statistics of China Securities Depository and Clearing Corporation, as of March 2020, there are 163.3 million securities investors in China. Stock price forecasting is a difficult and meaningful task for financial institutions and private investors. In order to effectively reduce investment risks and obtain stable returns on investment, many scholars put forward a large number of prediction models [1–9]. With the speedy development of big data application technology, especially the application of machine learning and deep learning in the financial field, it has a profound impact on investors. Research directions include low-frequency data and high-frequency data [10]. The previous research studies are mainly divided into two kinds of methods: fundamental analysis and technical analysis [11].

On the one hand, in technical analysis, people widely use mathematical statistical techniques to analyze historical stock price trends and predict recent stock prices. In recent years, many researchers have applied a variety of machine learning algorithms to analyze and predict stock prices, such as neural networks, multicore learning [12], stepwise regression analysis [13], and deep learning [14, 15]. Although many algorithms have achieved good results in certain aspects, there are many parameter configurations and data selection problems in the use of machine learning, which is still an important area of research. On the other hand, in the fundamental analysis [16–18], people mainly use natural language processing to analyze the company's financial news and financial statements to predict the future stock price trend.

The long-short term memory (LSTM) is a very good method in dealing with time series. Stock price data belongs

to time-series data. Therefore, many researchers use LSTM [19–26] to analyze and predict stock prices. Many studies have analyzed the correlation between time-series data [27–30], and the results show that LSTM has advantages in time-series data. In the literature [31], researchers used LSTM to predict the coding unit split, and the experimental results proved the advantages of LSTM in terms of efficiency. Time-series data trend research is also a new form of time-series data prediction, so LSTM is a natural choice.

Empirical mode decomposition (EMD) technology is usually applied to nonstationary and nonlinear signals. EMD can decompose nonlinear signals into several inherent modal functions (IMF) adaptively. EMD can effectively suppress continuous noise, such as Gaussian noise [32]. However, EMD cannot suppress intermittent noise and mixed noise. Ensemble empirical mode decomposition (EEMD) technology can solve the problem of noise-mode mixing [33]. In the EEMD algorithm, a group of white noise is first added to the original signal, and then it is decomposed into several IMF. The average value of the corresponding IMF set is regarded as the correct result. EEMD will separate the noise in different IMF from the original signal components [34], thus eliminating the noise-mode mixing phenomenon. In recent years, the application of EEMD has attracted the attention of many researchers and scholars [27, 35–44]. In order to solve the problem that noise in practical applications makes interference term retrieval difficult, Zhang et al. [35] proposed a technique based on EEMD and EMD to achieve automatic interference term retrieval from the spectral domain low-coherence interferometry. The proposed algorithm uses EEMD technology to make the relative error of coupling strength less than 2%. To solve the problem that Gaussian noise and non-Gaussian noise seriously hinder the detection of rolling bearing defects by traditional methods, Jiang et al. [36] proposed a new rolling bearing inspection method that combines bispectrum analysis with improved integrated EMD. This method uses ensemble empirical mode decomposition technology to have superior performance in reducing multiple background noises and can more effectively detect defects in rolling bearings. To solve the problem of the influence of the authenticity of the partial discharge signal on the evaluation accuracy of the transformer insulation performance, Wang et al. [37] proposed a method to suppress white noise in PD signals based on the integration of EMD and the combination of high-order statistics. This method uses EEMD decomposition to the threshold and reconstructs each IMF to suppress the white noise in each component. To solve the problem that most existing measurement methods only focus on mathematical values and are affected by measurement errors, interference, and uncertainty, Wei et al. [38] proposed a new time-history comparison for vehicle safety analysis by the integrating empirical mode decomposition method. This method uses EEMD decomposition to make the trend signal to reflect the overall change and is not affected by high-frequency interference. To solve the difficult problem of wind speed prediction, Yang and Yang [39] proposed a hybrid BRR-EEMD short-term prediction method for wind speed based on the EMD and Bayesian

ridge regression (BRR). This hybrid method uses the Bayesian regression method and the EEMD to perform regression prediction on each subsequence decomposed by the EEMD and obtains good results in wind speed prediction. In order to find potential profit arbitrage opportunities when the returns of stock index futures contracts and stock index futures contracts continue to deviate from fair prices in irrational and nonefficiently operating markets, Sun and Sheng [40] proposed a time-series analysis method based on integrated EMD. This method uses EEMD to analyze the stock futures basis sequence and extracts a monotonically decreasing trend from the sequence to discover business opportunities. To improve the problem that a single method of predicting complex and nonlinear stock prices cannot achieve good results, Al-Hnaity and Abbod [41] proposed a hybrid integrated model based on ensemble empirical mode decomposition and backpropagation neural network to predict the closing price of the stock index. The researchers [41] have proposed five hybrid prediction models: ensemble EMD-NN, ensemble EMD-Bagging-NN, ensemble EMD-Crossvalidation-NN, ensemble EMD-CV-Bagging-NN, and ensemble EMD-NN. The experimental results show that the performance of the ensemble EMD-CV-Bagging-NN, ensemble EMD-Crossvalidation-NN, and ensemble EMD-Bagging-NN models based on ensemble EMD are all a grade higher than that of the ensemble EMD-NN model and significantly higher than the single neural network model.

The typical forecasting scheme is based on the forecast of the time-series data itself and does not deal with the time-series data itself. It has become a challenge that how to combine the existing forecasting methods to improve the forecasting effect by decomposing the time-series data. The above methods use EEMD to decompose the time-series data to improve the performance of the algorithm. How to effectively decompose complex and nonlinear stock time-series data for prediction has been puzzled by many researchers. Due to the uncertainty and nonlinearity of the stock time series, the deviation of a single method to predict stock prices is generally relatively large. The abovementioned hybrid method does indeed improve the algorithm significantly. Therefore, it can be boldly guessed that the hybrid method generally can get better prediction results than the single specific method. Besides, the original complex time series was decomposed by the EEMD method into several relatively stable subsequence time series. By effectively combining several current effective forecasting methods, the forecasting results of relatively stable subsequence time series are theoretically better. Combining the features of the EEMD method based on the improved empirical mode decomposition method and the LSTM machine learning algorithm, this paper proposes a hybrid LSTM-EEMD method for stock index price prediction.

The rest of this paper is organized as follows. Three related terminology such as EMD, EEMD, and BRR are presented in Section 2, while Section 3 briefly introduces the flowchart of our proposed LSTM-EEMD method and the structure of our proposed hybrid LSTM-EEMD method. In Section 4, we describe the experiment data collection,

experiment data preprocessing, and modeling processing. In Section 5, we describe the experimental results of our proposed hybrid LSTM-EEMD method and analyze the results of simulation experiment of our proposed hybrid LSTM-EEMD method for prediction. Finally, the conclusion of this paper and some future works are described in Section 8.

2. Related Works

Over the years, many studies in the financial field have focused on the problems of stock price prediction. These studies mainly focus on three important research directions: (1) based on the machine learning method; (2) based on the time-series analysis method; and (3) based on the hybrid method. Below, we first briefly introduce related terminology. Then, we introduce the LSTM and EEMD related to this study.

2.1. Stock Price Predicting. Stock prices are a highly volatile time series. Stock prices are affected by various factors such as national policies, interest rates, exchange rates, industry news, inflation, monetary policies, temporary events, investor sentiment, and human intervention. Predicting stock prices on the surface requires the establishment of a model of the relationship between stock prices and these factors. Although these factors will temporarily change the stock price, in essence, these factors will be reflected in the stock price and will not change the long-term trend of the stock price. Therefore, stock prices can be predicted simply with historical data.

This paper believes that there are many studies using a single analysis method to predict stock market trends, but the results are not good. Need to consider a variety of factors or use a variety of techniques to build a hybrid model to further explore the prediction of stock prices. In-depth systematic research is required to answer the following research questions:

RQ1 Which factors or combinations of factors most affect the trend of the stock?

RQ2 What kind of analysis technology combination is most suitable for stock trend prediction?

RQ3 Do we need to use deep learning methods to mine data in order to better discover the internal relationship between the stock market and influencing factors?

RQ4 Whether the predictability of the analysis model depends on specific stock company characteristics, such as the domain, shareholder background, and policies?

RQ5 In the context of stock market forecasting, have we developed some effective forecasting methods?

RQ6 We should focus on the analysis of the specific nature of the stock price itself, rather than solving general relationship problems. Whether the price analysis driven by influencing factors can be more effective?

2.2. LSTM. The long-short term memory neural network is generally called LSTM. The LSTM was proposed by Hochreiter and Schmidhuber [27] in 1997. The LSTM is a special type of recurrent neural network (RNN). The biggest feature of the RNN which was improved and promoted by Alex Graves is that long-term dependent information of data can be obtained. LSTM has been widely used in many fields and has achieved considerable success in many problems. Since LSTM can remember the long-term information of data, the design of LSTM can avoid the problem of long-term dependence. Currently, the LSTM is a very popular time-series forecasting model. Below, we first introduce the RNN network, followed by the LSTM neural network.

3. Preliminaries

3.1. Recurrent Neural Network. When we deal with problems related to the timeline of events, such as speech recognition, sequence data, machine translation, and natural language processing, traditional neural networks are powerless. RNN is specifically proposed to solve these problems. Because the correlation between the contexts of the text needs to be considered in the word processing, the weather conditions of consecutive days and the relationship between the weather conditions of the day and the past days need to be considered when predicting the weather.

The RNN has a chain form of repeating neural network modules. In the standard RNN, this repeated structural module has only a very simple structure, such as a tanh layer. The simple structure of the recurrent neural network is shown in Figure 1.

The design intent of RNN is to solve nonlinear problems with timelines. The way of internal connection of the recurrent neural network generally only feeds forward the data, but in the bidirectional recurrent neural network, it allows the forward and backward directions to feedback the data. The RNN has designed a feedback mechanism, so the RNN can easily update the weight or residual value of the previous step. The design of the feedback mechanism is very suitable for time-series forecasting. The RNN can extract rules from historical data and then use the rules to predict time series. Figure 1 shows the simple structure of the RNN, and Figure 2 shows the expanded diagram of the basic structure of the RNN. The left side of the arrow with unfold label in Figure 2 is the basic structure of the recurrent neural network. The right side of the arrow with unfold label in Figure 2 is a continuous 3-level expansion of the basic structure of the recurrent neural network. An input data x_t is input into module h of the RNN. The y_t is an output of module h of the RNN values at time t . Like other neural networks, the recurrent neural network shares all parameters of each layer, such as W_{hx} , W_{hh} , and W_{yh} in Figure 2. As shown in Figure 2, the RNN shares two input parameters W_{hx} and W_{hh} , and one output parameter W_{yh} . As we all know, the number of parameters in each layer of a general multilayer neural network is different. Looking at Figure 2, we feel that the operation of each step of the recurrent neural network is the same on the surface. In fact, the output y_t and the input x_t are different. The number of parameters of the

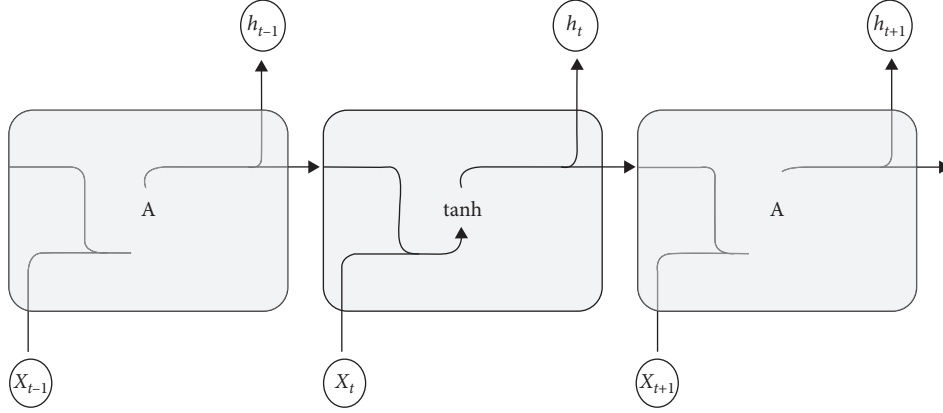


FIGURE 1: The simple structure of the RNN.

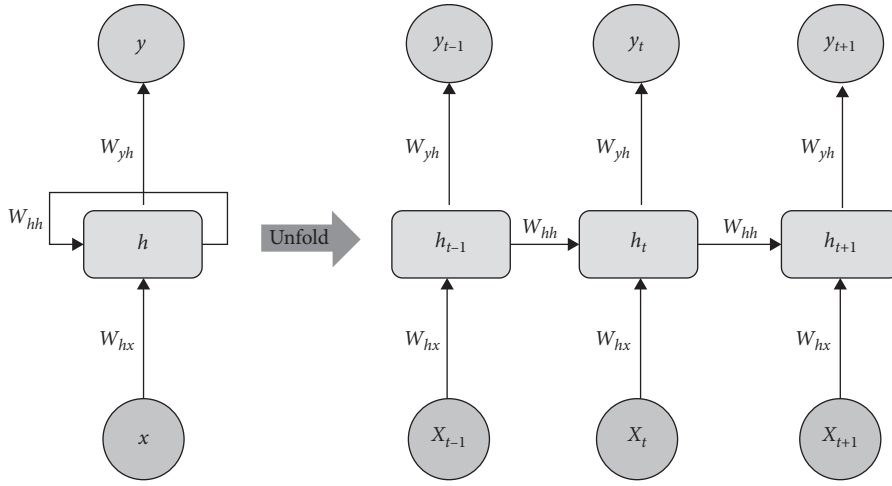


FIGURE 2: The expanded diagram of the basic structure of the RNN.

recurrent neural network will be significantly reduced during training. After multilevel expansion, the recurrent neural network becomes a multilayer neural network. Looking closely at Figure 2, we find that W_{hh} between layer h_{t-1} and h_t is the same as W_{hh} between layer h_t and h_{t+1} in form. In value and meaning, W_{hh} between layer h_{t-1} and h_t is different as W_{hh} between layer h_t and h_{t+1} . Similarly, W_{hx} and W_{yh} have similar situations.

Although each layer of the RNN neural network has output and input modules, the output and input modules of some layers can be omitted in specific application scenarios. For example, in language translation, we only need the overall language symbol output after the last language symbol is input and do not need to know the language symbol output after each language symbol is input. The main feature of RNN is self-expanding, with multiple hidden layers.

As we all know, during network training, the recurrent neural network models are prone to disappearing gradients. Once the gradient of the model disappears completely, the algorithm enters an endless loop, and the network training will not end, eventually leading to RNN paralysis. Therefore, simple RNNs are prone to gradient disappearance problems and are not suitable for long-term predicting.

The purpose of designing LSTM is to avoid or reduce the appearance of the problem of vanishing gradient while dealing with long-term correlation time series by simple RNN. Based on the simple RNN, the LSTM adds the output gates, the input gates, and the forget gates. In Figure 3, all three gates are replaced by σ , which can effectively prevent the gradient from being eliminated. Therefore, LSTM can solve long-term dependence problems. The purpose of designing memory neurons is to store some LSTM important information for state information. In addition, in general, each gate has an activation function. This function performs nonlinear transformations or trade-offs on data. Generally, the forget gate f_t can filter some status information. Equations (1)–(6) associated with the LSTM neural network is shown below. They are for the forget gate, input gate, inverse of the memory cell, memory cell, output gate, and output, respectively:

$$f_t = \sigma(W_f \cdot (h_{t-1} \| x_t) + b_f), \quad (1)$$

$$i_t = \sigma(W_i \cdot (h_{t-1} \| x_t) + b_i), \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot (h_{t-1} \| x_t) + b_C), \quad (3)$$

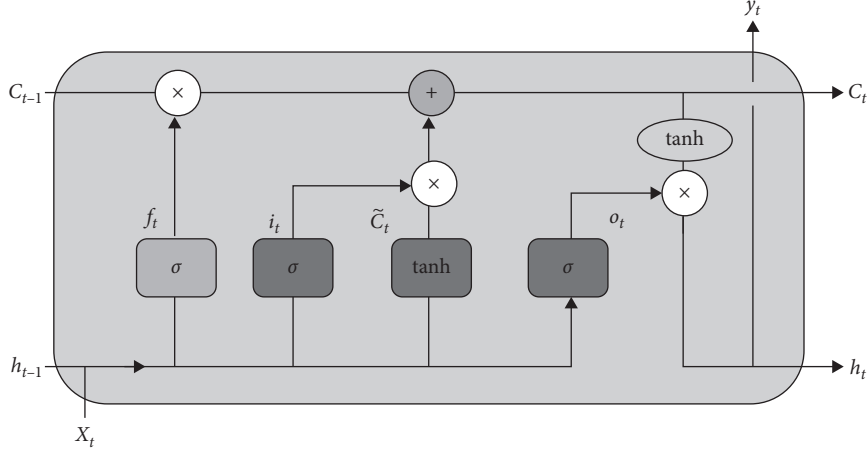


FIGURE 3: The basic structure of the LSTM neural network.

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t, \quad (4)$$

$$o_t = \sigma(W_o \cdot (h_{t-1} \| x_t) + b_o), \quad (5)$$

$$h_t = o_t \times \tanh(C_t). \quad (6)$$

3.2. *EMD*. The empirical mode decomposition (EMD) proposed by Huang et al. in 1998 [42] is a widely used adaptive time-frequency analysis method. Empirical mode decomposition is an effective decomposition method for time-series data. Due to the common local features of time-series data, the EMD method has extracted the required data from them and obtained very good results in applications fields. Hence, many scholars apply the EMD method in many fields successfully. The prerequisite for EMD decomposition is the existence of the following three assumptions [43]:

The following briefly introduces the decomposition process:

- (1) Suppose there is a signal $s(i)$ with the black line in Figure 4. The extreme value of the signal is shown in red and blue dots. Form the upper wrapping line through all the blue dots and form the lower wrapping line through all the red dots.
- (2) Calculate the average value of the lower and upper wrapping lines to form an average purple-red line $m(i)$. Here, **define the discrepancy line $d(i)$** as

$$d(i) = s(i) - m(i). \quad (7)$$

- (3) Judge whether the **discrepancy line $d(i)$ is an IMF according to IMF judgment rules**. If the discrepancy line $d(i)$ comply with IMF judgment rules, the discrepancy line $d(i)$ is the i th IMF $f(i)$. Otherwise, the discrepancy line $d(i)$ is considered the signal $s(i)$ and

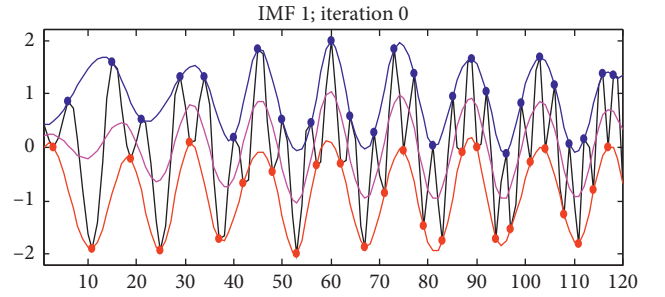


FIGURE 4: The decomposition process chart of sequences.

repeat two steps above until $d(i)$ complies with IMF judgment rules. After this, the IMF $f(t)$ is defined as

$$f(t) = d(t), \quad t = 1, 2, 3, \dots, n-1. \quad (8)$$

- (4) Calculate and get the IMF $f(t)$ by

$$r(t) = s(i) - c(t), \quad (9)$$

where $r(t)$ is considered the residual signal.

- (5) Repeat execution the four steps above N times until running status meets stop conditions. Obtain the N IMFs which meet with

$$\begin{cases} r_1 - c_2 = r_2, \\ \vdots \\ r_{N-1} - c_N = r_N. \end{cases} \quad (10)$$

- (1) Next level signal has to contain more than two extreme values: one is the minimum value and the other is the maximum value

- (2) Determine the time scale of signal characteristic based on the time difference between two extreme values
- (3) If the data has only an inflection point and no extremum, more times judgments are needed to reveal the extremum

Finally, the following equation (11) expresses the composition of the original signal $s(i)$:

$$s(i) = \sum_{j=1}^N f_j(t) + r_j(t). \quad (11)$$

3.3. EEMD. A classical EMD has mode mixing problems when decomposing complex vibration signals. To solve the above problem, Wu and Huang [44] proposed the EEMD method in 2009. The EEMD method is short for ensemble empirical mode decomposition method. EEMD is commonly used for nonstationary signal decomposition. However, the EEMD method is significantly different from WT transform and FFT transform. Here, WT is the wavelet transform and FFT is the fast Fourier transform. Without the need for basis functions, the EEMD method can decompose any complex signal. At the same time, the EEMD method can decompose any signal into many IMFs. Here, IMF is the intrinsic modal function. The decomposed IMF components contain local different feature signals. EEMD can decompose nonstationary data into multiple stable subdata and then use Hilbert transform to get the time spectrum, which has important physical significance. Comparative analysis with FFT and WT decomposition, the EEMD has the characteristics of intuitive, direct, posterior, and adaptive. The EEMD method has adaptive characteristics because of the local features of the time-series signal. The following briefly introduces the process of EEMD decomposing data.

Assume that the EEMD will decompose the sequence X . According to the steps of EEMD decomposition, n subsequences will be obtained after decomposition. These n subsequences include $n-1$ IMFs and one remaining subsequence R_n . Here, these $n-1$ IMFs are $n-1$ component subsequence $C_i (i=1, 2, \dots, n-1)$ of the original sequence X . These $n-1$ IMFs are named $IMF_i (i=1, 2, \dots, n-1)$. The remaining subsequence R_n is sometimes named residual subsequence. The detailed steps of using EEMD to decompose the sequence are introduced as follows:

- (1) Suppose there is a signal $s(i)$ with the black line in Figure 4. The extreme value of the signal is shown in red and blue dots. Form the upper wrapping line through all the blue dots and form the lower wrapping line through all the red dots. The lower wrapping line is shown in the red line in Figure 4. And the upper wrapping line is the blue line in Figure 4.
- (2) Calculate the average value of the lower and upper wrapping lines to form a mean line $m(i)$ which is shown in purple-red line in Figure 4.

- (3) Obtain the first component IMF_1 of the signal $s(i)$. The IMF_1 is obtained by calculation formula $d(i) = s(i) - m(i)$. This formula means the difference of the original signal $s(i)$ minus the mean line $m(i)$ of the lower and upper wrapping lines.
- (4) Take the first component IMF_1 $d(i)$ as a new signal $s(i)$ and repeat execution the three steps above until running status meets stop conditions. The following describes the stop condition in detail:
 - (a) The average $m(i)$ is approximately 0
 - (b) The number of signal lines $d(i)$ passing through zero points is greater than or equal to the number of extreme points
 - (c) The number of iterations reaches the set maximum
- (5) Take subsequence $d(i)$ as the i th IMF $fi (i=1, 2, \dots, n-1)$. Obtain the residual R by calculating the formula $r(i) = s(i) - d(i)$.
- (6) Take the residual $r(i)$ as the new signal $s(i)$ to calculate the $(i+1)$ th IMF, and repeat execution the five steps above until running status meets stop conditions. The following describes the stop condition in detail:
 - (a) The signal $s(i)$ has been completely decomposed and has obtained all IMF
 - (b) The number of decomposition level reaches the set maximum

Finally, the following equation (12) expresses the composition of the original sequences and n subsequences decomposed by the EEMD:

$$x(t) = \sum_{i=1}^{n-1} (C_i) + R_n, \quad i = 1, 2, \dots, n-1, \quad (12)$$

where the number n of subsequences depends on the complexity of the original sequences. Figure 5 shows the sin time series represented by equation (13) and the IMF diagram of it which is decomposed by the EEMD:

$$x(t) = \sin(20\pi t) + 2^* \sin(200\pi t) + 3t, \quad (13)$$

where $t = 0, f, 2f, \dots, 1000f, f = 0.001$.

4. Methodology

The principle of our hybrid prediction method LSTM-EEMD for stock price based on LSTM and EEMD is introduced in detail in this section. These theories are the theoretical formation of our forecasting methods. The following first introduces the flowchart, the basic structure, and the process of the LSTM-EEMD hybrid stock index prediction method based on the ensemble empirical mode decomposition and the long-short term memory neural network. Our proposed hybrid LSTM-EEMD prediction method first uses the EEMD to decompose the stock index sequences into a few simple stable subsequences. Then, the predict result of each subsequence is predicted by the LSTM

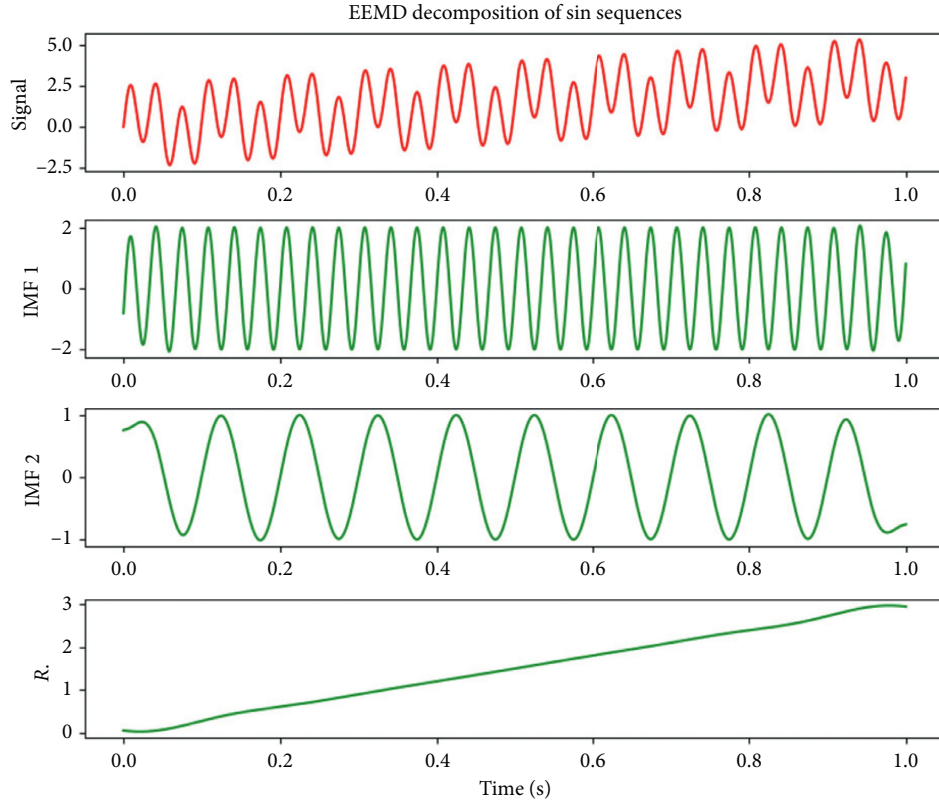


FIGURE 5: The sequences of sin signal decomposed by the EEMD.

method. Finally, the LSTM-EEMD obtains the final prediction result of the original stock index sequence by fusing all LSTM prediction results of several stock index subsequences.

Figure 6 shows the structure and process of the LSTM-EEMD method. The basic structure and process of the LSTM-EEMD predict method include three modules. The three modules are the EEMD decomposition module, LSTM prediction module, and fusion module. Our proposed LSTM-EEMD prediction method includes three stages and eight steps. Figure 7 shows three stages and eight steps of our proposed method. The three stages of the proposed hybrid LSTM-EEMD prediction method are input data, model predict, and evaluate model. The model evaluation stage and the data input stage each include 4 steps, and the model prediction stage includes 3 steps. The hybrid LSTM-EEMD prediction method is introduced in detail as follows:

- (1) The simulation data is generated. And the real-world stock index time-series data are collected. Then, the original stock index time-series data are pre-processed to make the data format of stock index time series satisfy the format requirements for decomposition of the EEMD. Finally, the input data X of the LSTM-EEMD hybrid prediction method is formed.
- (2) The input data X is decomposed into a few sequences by the EEMD method. If n subsequences are obtained, then there are one residual subsequence R_n and $n-1$ subsequences. These n subsequences are

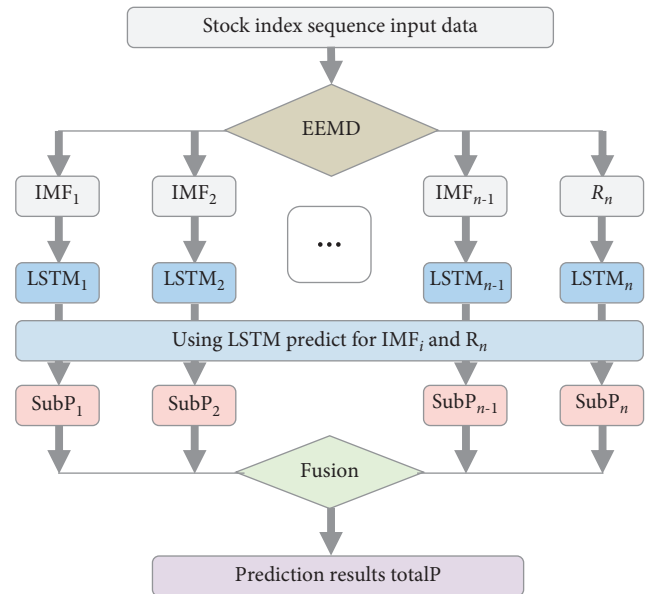


FIGURE 6: The structure and process of the LSTM-EEMD method.

expressed as R_n , IMF_1 , IMF_2 , IMF_3 , ..., IMF_{n-1} , respectively.

- (3) The prediction process of any subsequence is not affected by the prediction process of other subsequences. A LSTM model is established and trained for each subsequence. Hence, we need to build and

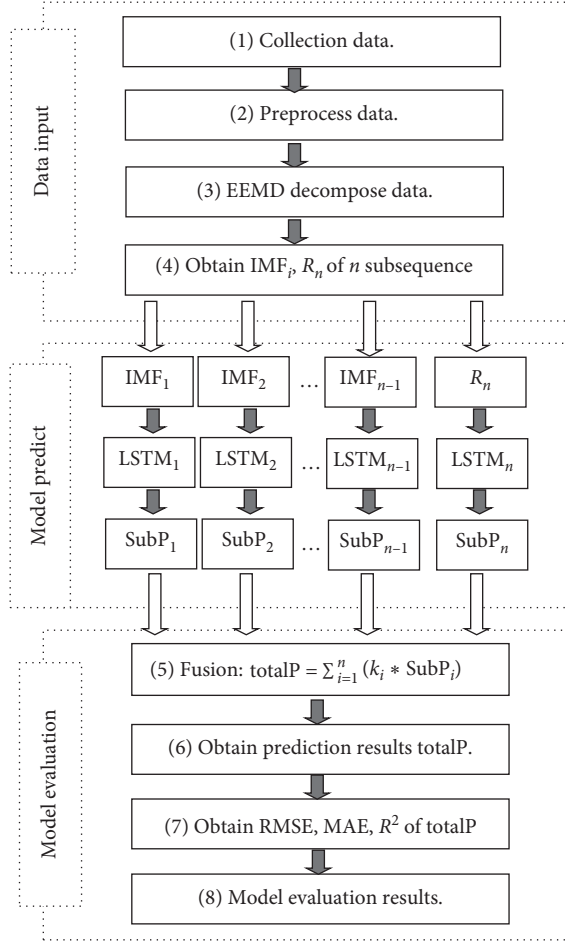


FIGURE 7: The data flowchart of our proposed hybrid method.

train n LSTM for n independent subsequences. These n independent LSTM models are named $LSTM_k (k = 1, 2, \dots, n-1, n)$, respectively. We use the n LSTM models to predict these n independent subsequences and get n prediction values of the stock index time series. These n prediction values are named $SubP_k (k = 1, 2, \dots, n-1, n)$, respectively.

- (4) Fusion function is the core of hybrid method. At present, there are many fusion functions, such as sum, weighted sum, weighted product, and so on. The function of these fusion functions is to merge several results into the final result. In this paper, the proposed hybrid stock prediction method selected the weighted sum as the fusion function. The weighted results of all subsequences are accumulated to form the final prediction result for the original stock index data. The weight here can be preset according to the actual application. In this paper, we use the same weight of each subsequence and the weight of each subsequence is 1.
- (5) Finally, we compare the predicted values with the actual value of stock index time-series data sequence and calculate the values of RMSE, MAE, and R^2 . We use three evaluation criteria of the RMSE, MAE, and

R^2 to evaluate the LSTM-EEMD hybrid prediction method. According to these evaluation values, the pros and cons of the method can be judged.

Figure 7 shows the predict progress and data flowchart of the proposed LSTM-EEMD method in this paper. The predict progress in Figure 7 can be introduced in 3 stages. The three stages are input data, model predict, and evaluate model. The stage of input data is divided into 4 steps. The four steps are collect data, preprocess data, decompose data by the EEMD, and generate n sequence. There are n LSTM model in the stage of model predict. The input data of the LSTM model is n sequences generated in the previous stage. These LSTM models separately predict n sequences to obtain n prediction results. The stage of the evaluate model is also divided into 4 steps. The first step is to fuse n predicted values with weights. In this paper, we choose weighted addition as the fusion function. The weighted addition fusion function sums the n prediction value with certain weights. The output result of the weighted addition fusion function is the prediction result of the stock index time-series data. Finally, we need to calculate the value of R^2 , MAE, and RMSE of the prediction results before evaluating the proposed hybrid prediction model. The quality of the proposed hybrid prediction model can directly evaluated by the values of R^2 , MAE, and RMSE.

5. Experiment Data

The experiment data in our research of this paper is introduced in detail in this part. We selected two types of experiment data to test in this paper in order to better evaluate the prediction effect of this method. The first type of experiment data is artificially simulated data generated automatically by computer. The correctness and effectiveness of our method are verified by these artificially simulated data. The second type of experimental data is real-world stock index data. Only the actual data of the society can really test the quality of the method. The model tested through social actual data is the most fundamental requirement for applying our proposed method to some real-world fields.

5.1. Artificially Simulated Experimental Data. We use artificially simulated experiment data to verify the effectiveness and correctness of our method. To get effective and accurate experiment result, the artificially generated simulation experiment data should be long enough. Hence, in the experiment, we choose the length of the artificially simulated experiment data to be 10,000. The artificially simulated experiment data is created by the computer according to the sin function of formula (13).

5.2. Real-World Experimental Data. In order to empirically study the effect of the proposed prediction method, we collected stock indices data in the real-world stock field as experiment data from Yahoo Finance. To obtain more objective experimental results, we choose 4 stock indices from

different countries or regions. These stock indices are all very important stock indices in the world's financial field.

The four stock indices in this experiment are ASX, DAX, HSI, and SP500. The SP500 is used as an abbreviation of the Standard&Poor's 500 in this paper. The SP500 is a US stock market index. This stock index is a synthesis of the stock indexes of 500 companies listed on NASDAQ and NYSE. The HSI is used as an abbreviation of the Hang Seng Index. The HSI is an important stock index for Hong Kong in China. The German DAX is used as an abbreviation of the Deutscher Aktienindex in Germany in Europe. The DAX stock index is compiled by the Frankfurt Stock Exchange. DAX company consists of 30 major German companies, so it is a blue chip market index. The ASX is used as an abbreviation of Australian Securities Exchange and is a stock index compiled by Australia.

The datasets of every stock index include five daily properties. These five properties are transaction volume, highest price, lowest price, closing price, and opening price. The data time interval of each stock index is from the opening date of the stock to May 18, 2020.

5.3. Data Preprocessing. In order to obtain good experiment results, we try our best to deal with all the wrong or incorrect data in the experiment data. Of course, more experimental data is also an important condition for obtaining good experimental results. Incorrect data mainly include records with zero trading volume and records with exactly the same data for two and more consecutive days. After removing wrong or incorrect noise data from original data, we show the trend of the close price for the four major stock indexes in Figure 8.

In the experiment, we usually first standardize the experimental data. Let $X_i = \{x_i(t)\}$ be the i th stock time-series index at time t , where $t = 1, 2, 3, \dots, T$ and $i = 1, 2, 3, \dots, N$. We define the daily return logarithmic as shown in the formula $G_i(t) = \log(x_i(t)) - \log(x_i(t-1))$. We define the daily return standardization as shown in the formula $R_i(t) = (G_i(t) - \langle G_i(t) \rangle) / \delta$, where $\langle G_i(t) \rangle$ is the mean values of the daily return logarithmic $G_i(t)$ and δ is the standard deviation of the daily return logarithmic $G_i(t)$:

$$\delta = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}, \quad (14)$$

where $\bar{x} = 1/n \sum_{i=1}^n x_i$.

6. Experiment Results of Other Methods

In order to compare the performance of other forecasting methods, we comparatively studied and analyzed the results of other three forecasting methods on the same data in the experiment. Table 1 shows the experiment results of five prediction methods. Since the LSTM is introduced above, it will not be repeated here. Firstly, SVR, BARDDR, and KNR are briefly introduced. Then, the experimental results of these three methods are analyzed in detail.

6.1. SVR. SVR is used as an abbreviation of the Support Vector Regression. The SVR is a widely used regression method. Refer to the manual of the *libsvm* toolbox, the loss, and penalty function control the training process of machine learning. The *libsvm* toolbox is support vector machine toolbox software, and this toolbox is mainly used for SVM pattern recognition and regression software package. In the experiment, the SVR used was a linear kernel, which was implemented with *liblinear* instead of *libsvm*. The SVR should be extended to large number of samples. The SVR can choose a variety of penalty functions or loss functions. SVR has 10 parameters, and the settings of these parameters are shown in Table 2.

6.2. BARDDR. BARDDR is short for Bayesian ARD Regression which is used to fit the weight of the regression model. This method assumes that the weight of the regression model conforms to the Gaussian distribution. To better fit the regression model weights, the BARDDR uses the ARD prior technique. We assume the distribution of the regression model weights conforms to the Gaussian distribution. The parameter alpha and parameter lambda of BARDDR are the precision of the noise distribution and the precisions of the weights distributions, respectively. BARDDR has 12 parameters. Table 3 shows the settings of these parameters.

6.3. KNR. KNR is used as an abbreviation of the K -nearest Neighbors Regression. The K -nearest Neighbors Regression model is also a parameterless model. The K -nearest Neighbors Regression just uses the target value of the K -nearest training samples to make a decision on the regression value of the sample to be tested. That is, predict the regression value based on the similarity of the sample. Table 4 shows the KNR 8 parameters and the settings of these parameters.

The experimental results of other four methods and our proposed two methods for five sequences of sin, SP500, HSI, DAX, and ASX are shown in Table 1. In the experiment, we preprocessed the five sequences for those methods. Table 5 shows the length of the experimental data. Table 1 shows the experiment values of R^2 (R Squared), MAE (Mean Absolute Error), and RMSE (Root Mean Square Error). According to the real results and the predicted results, we try to get smaller experimental value of MAE, RMSE, and R^2 . The smaller the result of MAE or RMSE is, the better experimental values of the method are. However, the larger the result of R^2 is, the better the prediction effect of the method is.

For comparison, we show the top two results of each sequences in bold, as shown in Table 1. Among the above four traditional methods (SVR, BARDDR, KNR, and LSTM) for predicting sin artificial sequences data, BARDDR and LSTM are found to be the best methods. When predicting SP500, HSI, and DAX real time-series data, the BARDDR method and LSTM method have the best results. However, the BARDDR method and SVR method have the best results while predicting ASX real time-series data. Therefore, among the above four traditional methods, the BARDDR

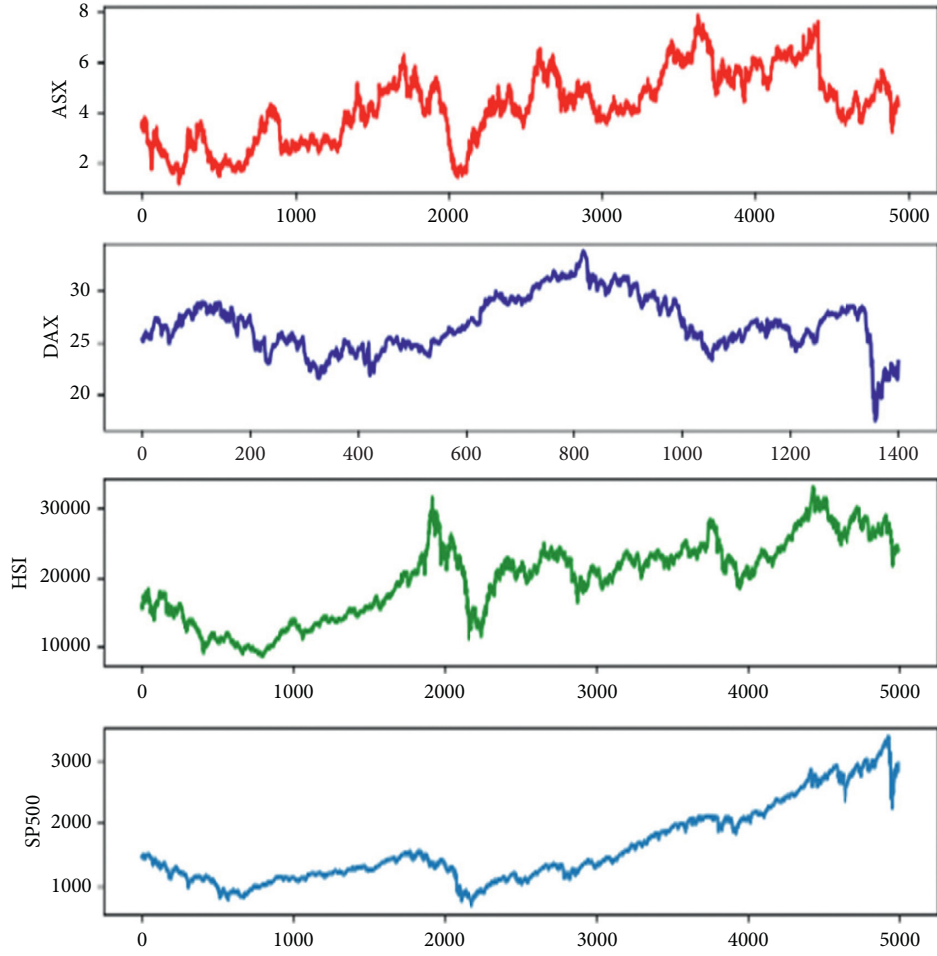


FIGURE 8: The trend of the four major stock indexes.

TABLE 1: Prediction results of time series.

| Method | | Sin | SP500 | HSI | DAX | ASX |
|--------------------|-------|-----------------|------------------|------------------|-----------------|-----------------|
| SVR | RMSE | 0.282409 | 1219.904783 | 1334.991215 | 0.653453 | 0.206921 |
| | MAE | 0.238586 | 990.639020 | 1282.862498 | 0.415796 | 0.146022 |
| | R^2 | 0.968293 | -1.665447 | -1.088731 | 0.938262 | 0.959223 |
| BARDR | RMSE | 0.012349 | 114.140660 | 326.487107 | 0.387025 | 0.110975 |
| | MAE | 0.011054 | 110.146472 | 233.641980 | 0.256125 | 0.075355 |
| | R^2 | 0.999939 | 0.998275 | 0.992213 | 0.978343 | 0.988271 |
| KNR | RMSE | 0.567807 | 1198.484251 | 553.641943 | 0.740763 | 0.303352 |
| | MAE | 0.443162 | 937.309021 | 452.300293 | 0.445212 | 0.186848 |
| | R^2 | 0.871824 | -1.572662 | -1.239516 | 0.920661 | 0.912361 |
| LSTM | RMSE | 0.059597 | 30.261563 | 306.481549 | 0.654662 | 0.135236 |
| | MAE | 0.051376 | 19.164419 | 224.250031 | 0.504645 | 0.106639 |
| | R^2 | 0.998562 | 0.997128 | 0.980270 | 0.962750 | 0.931165 |
| Proposed LSTM-EMD | RMSE | 0.032478 | 76.562082 | 78.487994 | 0.403972 | 0.066237 |
| | MAE | 0.026175 | 47.317009 | 71.062834 | 0.217733 | 0.045643 |
| | R^2 | 0.999569 | 0.985523 | 0.987594 | 0.979444 | 0.996174 |
| Proposed LSTM-EEMD | RMSE | 0.201358 | 48.878331 | 48.783906 | 0.324303 | 0.101511 |
| | MAE | 0.164348 | 38.556500 | 38.564812 | 0.247722 | 0.081644 |
| | R^2 | 0.982980 | 0.994171 | 0.988451 | 0.986832 | 0.991071 |

TABLE 2: The value of eight parameters of SVR.

| Parameter name | Parameter type | Parameter value |
|-------------------|-----------------------------------|-----------------|
| epsilon | Float | 0.0 |
| Tol | Float | $1e-4$ |
| C | Float | 1.0 |
| Loss | “epsilon” or “squared_epsilon” | Epsilon |
| fit_intercept | Bool | True |
| intercept_scaling | Float | 1.0 |
| Dual | Bool | True |
| Verbose | Int | 1 |
| random_state | Int | 0 |
| max_iter | Int | 1000 |

TABLE 3: The value of eight parameters of BARDR.

| Parameter name | Parameter type | Parameter value |
|------------------|----------------|-----------------|
| n_iter | Int | 300 |
| Tol | Float | $1e-3$ |
| alpha_1 | Float | $1e-6$ |
| alpha_2 | Float | $1e-6$ |
| lambda_1 | Float | $1e-6$ |
| lambda_2 | Float | $1e-6$ |
| Compute_score | Bool | False |
| Threshold_lambda | Float | 10000 |
| fit_intercept | Bool | True |
| Normalize | Bool | False |
| copy_X | Bool | True |
| Verbose | Bool | False |

method and LSTM method are the best methods for the five sequences data.

Carefully observed Table 1, we found that the remaining five methods, in addition to the KNR, all have very good prediction effects on sin sequences. The R^2 evaluation indexes of SVR, BARDR, LSTM, LSTM-EMD, and LSTM-EEMD methods are all greater than 0.96. BARDR, LSTM, and LSTM-EMD have the best prediction effect, and their R^2 evaluation indexes are all greater than 0.99. Among them, BARDR has the best prediction effect, and its R^2 evaluation value is greater than 0.9999. These values show that the method has better prediction effect for the time series of change regularity and stability, especially the BARDR method. Here, the result of choosing one of the six methods shows the prediction effect in Figure 9. To make the resulting map clear and legible, the resulting graph of Figure 9 displays only the last 90 prediction results.

Observing the SVR experiment results in Table 1, we found that the prediction values of this method on DAX, ASX, and sin is better than that of SP500 and HSI time-series data. Figure 10 shows the prediction results of the SVR on ASX, which has a better prediction effect on ASX stock data. Because the change of sin sequence has good regularity and stability, the SVR method is more suitable to predict sequence with good regularity and stability. It can be speculated that the DAX and ASX time-series data have good regularity and stability. However, the SVR method predicts SP500 and HSI sequences data. The prediction effect is very

poor. This shows that SP500 and HSI sequences data changes have poor regularity and stability.

Observing the experiment results of the KNR in Table 1, we found this method has similar performance to the SVR method. The prediction results on DAX, ASX, and sin sequence are better than that on SP500 and HSI sequence. Especially for stock SP500 and HSI sequence, the prediction effect is poor. The stock time series is greatly affected by people activities, so the changes in stock sequence are complicated, irregular, and unstable. It can be inferred that the KNR is not suitable to predict the stock sequence, so the KNR is not suitable to predict sequence predictions with unstable and irregular changes.

In Table 1, observing the experiment results of the BARDR and LSTM, we found the performance of the BARDR and LSTM methods is relatively good. These two methods not only have better prediction effects on DAX, ASX, and sin time-series data but also have more significant prediction effects on SP500 and HSI sequence. In particular, the prediction values of stock SP500 and HSI sequences are much better than that of KNR and SVR methods. The changes of stock sequence data are irregular, complex, and unstable, but the prediction effects of the BARDR and LSTM are still very good. It can be concluded that the BARDR and LSTM methods can predict stock time series, so the BARDR and LSTM are more suitable to predict sequence predictions with unstable and irregular changes. Figure 11 shows the prediction results of the BARDR for DAX time-series data, which has a better prediction effect on DAX stock time-series data.

In Table 1, observing the experiment results of the BARDR and LSTM, we found these two methods have good prediction effects on the five different time-series data provided by the experiment. By comparing their experimental results, it is easy to find that the performance of the BARDR is better than the LSTM method. Except that the LSTM is a little better than the BARDR in the SP500 time-series prediction effect, the LSTM is worse than the BARDR in the other four time-series prediction effects. Although the BARDR is better than the LSTM in predicting performance, the BARDR method is several times longer than the LSTM method in experimental time. In addition, although in our experiments, the prediction values of the LSTM is worse than the BARDR method, the experimental time is short and the training parameters used are still very few. In future work, we may be able to improve performance of our proposed methods by increasing the number of neurons and the number of training iterations of the LSTM method. Based on the principle analysis, we think the predictive effect of the LSTM will exceed that of the BARDR method by reasonably increasing the number of neurons and the number of training iterations.

7. Experiments

We selected two types of experiment data to test in this paper in order to better evaluate the prediction effect of this method. The first type of experiment data is artificially simulated data generated automatically by computer. The

TABLE 4: The value of eight parameters of KNR.

| Parameter name | Parameter type | Parameter value |
|----------------|---|-----------------|
| $n_neighbors$ | Int | 5 |
| Weights | “Uniform,” or “distance” | Uniform |
| Algorithm | {“Auto,” “ball_tree,” “kd_tree,” “brute”} | Auto |
| leaf_size | Int | 30 |
| p | Int | 2 |
| Metric | String or callable | Murkowski |
| metric_params | Dict | None |
| n_jobs | Int | 1 |

TABLE 5: The length of the experiment data.

| Name of data | Sin | SP500 | HSI | DAX | ASX |
|-------------------|-------|-------|------|------|------|
| All data length | 10000 | 23204 | 8237 | 1401 | 4937 |
| Train data length | 9000 | 20884 | 7413 | 1261 | 4443 |
| Test data length | 1000 | 2320 | 824 | 140 | 494 |

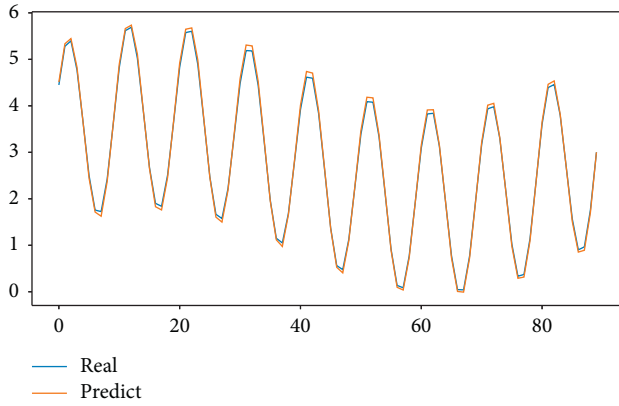


FIGURE 9: Prediction results of the LSTM method for sin sequence.

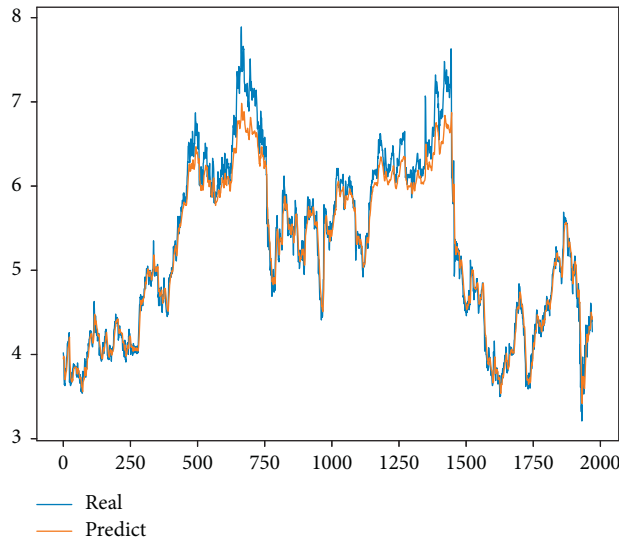


FIGURE 10: Prediction results of SVR for ASX sequence.

second type of experimental data is real-world stock index data. The model tested through social actual data is the most fundamental requirement for applying our proposed

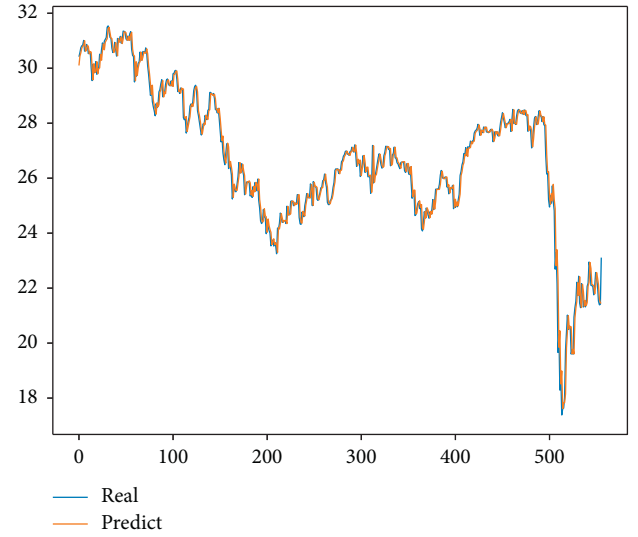


FIGURE 11: Prediction results of BARDR for DAX sequence.

method to some real-world fields. This section conducts detailed research and analysis on the experiment results of two aspects.

7.1. Analysis of Experimental Results Based on Artificial Simulation Data. We use artificially simulated experiment data to verify the effectiveness and correctness of our method. To get effective and accurate experiment result, the artificially generated simulation experiment data should be long enough. Hence, in the experiment, we choose the length of the artificially simulated experiment data to be 10,000, as shown in Table 5.

Before analyzing the experiment results of real-world stock index data, we first use the proposed LSTM-EMD and LSTM-EEMD methods to predict sin simulation sequence. The simulation experiment can verify the effectiveness and correctness of the two proposed methods.

Observing the sin data column of Table 1, we found the three indicators of the LSTM method, LSTM-EMD method, and LSTM-EEMD method are basically equivalent under the same number of neurons and iterations. The three results indicated that these three methods predict effects are similar to the sin simulation time-series data. Among them, the LSTM-EMD method has the best prediction effect, indicating that the method we proposed is effective and can improve the prediction effect of the experiment. Observing

the sin data column in Table 1, we found the experimental results of the LSTM are a little better than the LSTM-EEMD, but the gap is very small and can be almost ignored. After in-depth analysis, this is actually that the sin time-series data itself is very regular. By adding noise data and then using EEMD decomposition, the number of subtime series decomposition becomes more and more complicated, so the prediction effect is slightly worse. In summary, the two proposed prediction methods LSTM-EMD and LSTM-EEMD in this paper have a better comprehensive effect than the original LSTM method, indicating that the two proposed prediction methods are correct and effective.

7.2. Analysis of Experiment Results Based on Real Data. In order to verify the practicability of the proposed LSTM-EMD and LSTM-EEMD prediction methods, we use the two methods proposed in this paper to predict four real-world stock index time series in the financial market. The four time-series are the DAX, ASX, SP500, and HSI. The stock index time series is generated in human society and reflects social phenomena. The time series of stock indexes in the financial market is seriously affected by human factors, so it is complex and unstable. These four time-series are very representative. They come from four different countries or regions and can better reflect the changes in different countries in the world financial market.

By comparing the three evaluation indicators of RMSE, MAE, and R^2 of the LSTM, LSTM-EMD, and LSTM-EEMD prediction methods in Table 1, we found the prediction results of the LSTM-EEMD prediction method in the four sequence data are better than the LSTM-EMD prediction method. The experimental results show that the LSTM-EEMD prediction method is better than the LSTM-EMD prediction method. Figure 12 shows the prediction results of the LSTM-EMD method and the LSTM-EEMD method of HSI time series.

By observing Table 1, it is easy to find that the results of the proposed LSTM-EMD and LSTM-EEMD methods in the three sequences of HSI, DAX, and ASX are much better than the traditional LSTM method. We think that there are two main reasons for obtaining such good experimental results. On the one hand, the method proposed in this paper decomposes HSI, DAX, and ASX time series by EMD or EEMD so that the complex original HSI, DAX, and ASX time series are decomposed into multiple more regular and stable subtime series. The multiple subtime series are easier to predict than the complex original time series. Therefore, the predicting results of multiple subtime series is more accurate than the predicting results of directly predicting complex original time series. On the other hand, although the changes of HSI, DAX, and ASX time series are complicated, they can be decomposed into more regular and stable time series. In order to clearly understand the changes in HSI, DAX, and ASX time series and EMD or EEMD decomposition, Figures 13 and 14 show all changes in EMD or EEMD decomposition of DAX time series, respectively. Figure 8 is the original change of the DAX time series, and all subgraphs in Figures 13 and 14 are the EMD or EEMD decomposition

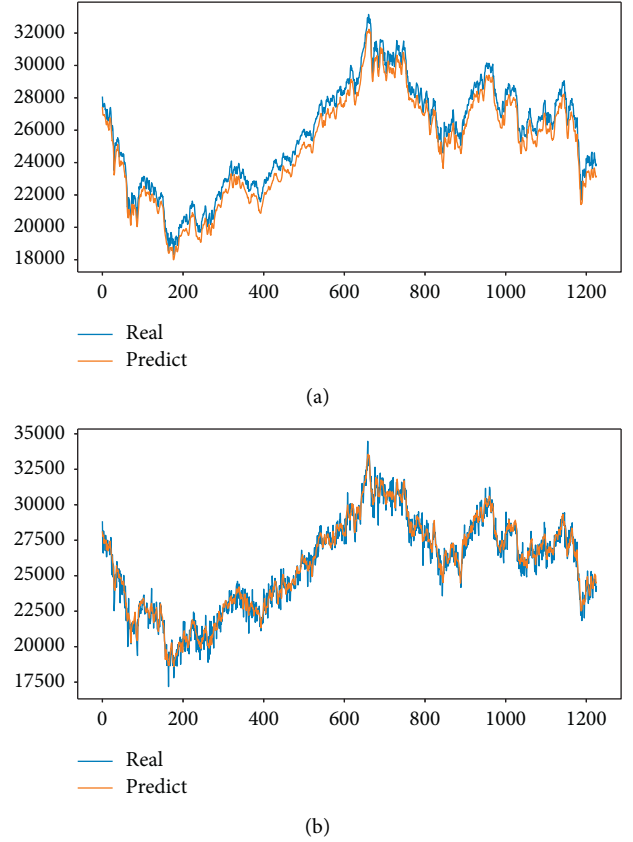


FIGURE 12: Prediction results of (a) LSTM-EMD method and (b) LSTM-EEMD method of HSI time series.

subgraphs of the DAX time series. It is easy to see that the lower EMD or EEMD decomposition subgraphs are gentler, more regular, more stable, more predictable, and have better prediction effects.

Carefully observe the data of the LSTM method, LSTM-EMD method, and LSTM-EEMD method in Table 1 in the three sequences of HSI, DAX, and ASX. It can be found that the proposed LSTM-EEMD method has a better experimental effect than the proposed LSTM-EMD method, and the proposed LSTM-EMD method has a better experimental effect than the traditional LSTM method. The three indicators RMSE, MAE, and R^2 used in the experiment all exemplify the above relationship. The smaller the RMSE or MAE experimental value of a method, the better the prediction effect of the method. However, the larger the R^2 value of a method, the better its prediction effect.

In this paper, we proposed a hybrid prediction method based on EMD or EEMD. The results of experiment show that the fusion prediction results are more superior to the traditional method for direct prediction of complex original time series in most instances. Although the experiments in this paper comparatively studied two hybrid methods based on EMD or EEMD and four other classical methods, such as SVR, BARDR, KNR, and LSTM. The two hybrid methods based on EMD and EEMD have different effects in different time series, and there are different time-series methods that reflect different experimental

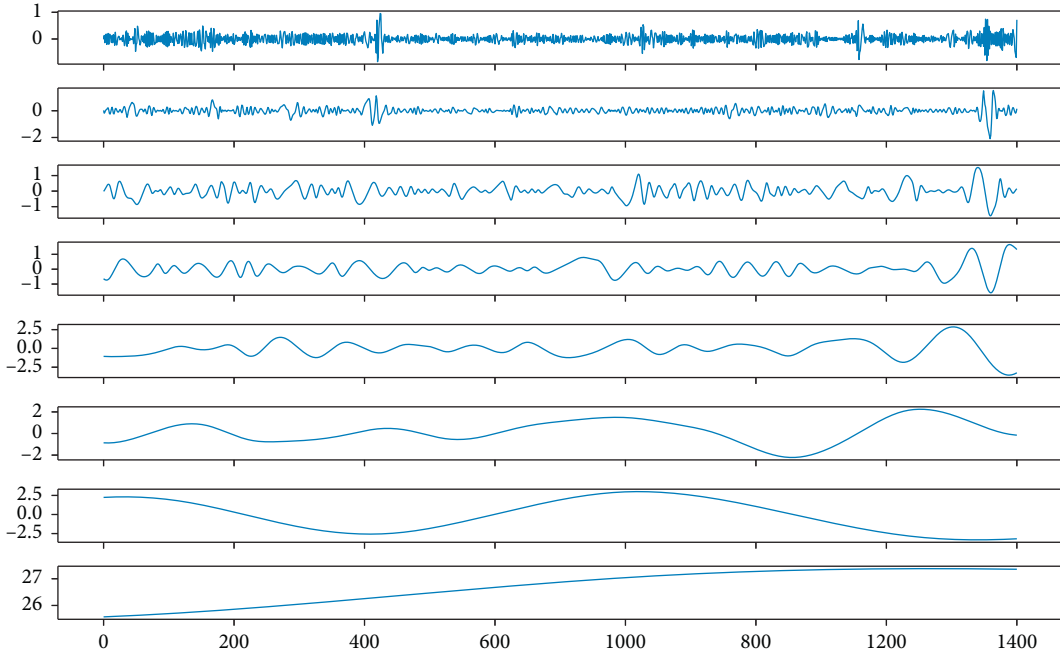


FIGURE 13: The EMD decomposition graphs of DAX time series.

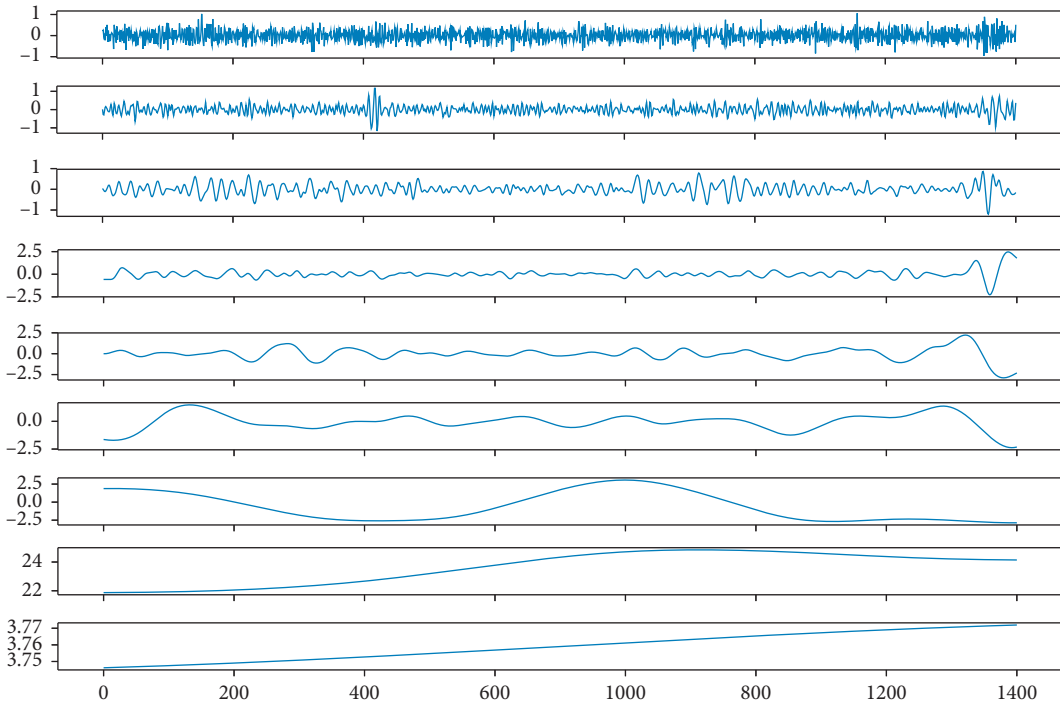


FIGURE 14: The EEMD decomposition graphs of DAX time series.

advantages and disadvantages. In actual application, people can choose different methods according to the actual situation and apply it to specific practical fields. In the actual environment, if the results obtained by the method you choose do not meet your expectations, you can choose another method.

8. Conclusion and Future Work

We proposed a hybrid short-term prediction method LSTM-EMD or LSTM-EEMD based on LSTM and EMD or EEMD decomposition methods. The method is based on a complex problem divided and conquered strategy. Combining the

advantages of EMD, EEMD, and LSTM, we used the EMD or EEMD method to decompose the complex sequence into multiple relatively stable and gentle subsequences. Use the LSTM neural network to train and predict each subtime series. The prediction process is simple and requires only two steps to complete. First, we use the LSTM to predict each subtime series value. Then, the prediction results of multiple subtime series are fused to form a complex original time-series prediction result. In the experiment, we selected five data for testing to fully the performance of the method. The comparison results with the other four prediction methods show that the predicted values show higher accuracy. The hybrid prediction method we proposed is effective and accurate in future stock price prediction. Hence, the hybrid prediction method has practical application and reference value. However, there are some shortcomings. The proposed method has some unexpected effects on the experimental results of time series with very orderly changes.

The research and application of analysis and prediction methods on time series have a long history and rapid development, but the prediction effect of traditional methods fails to meet certain requirement of real application on some aspects in certain fields. Improving the prediction effect is the most direct research goal. Taking the study results of this paper as a starting point, we still have a lot of work in the future that needs further research. In the future, we will combine the EMD method or EEMD method with other methods, or use the LSTM method in combination with wavelet or VMD.

Data Availability

Data are fully available without restriction. The original experimental data can be downloaded from Yahoo Finance for free (<http://finance.yahoo.com>).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Yang Yujun contributed to all aspects of this work. Yang Yimei and Xiao Jianhua conducted the experiment and analyzed the data. All authors reviewed the manuscript.

Acknowledgments

This work was supported in part by the Scientific Research Fund of Hunan Provincial Education under Grants 17C1266 and 19C1472, Key Scientific Research Projects of Huaihua University under Grant HHUY2019-08, Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, and the Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province under Grant ZNKZ2018-5. The fund source has no role in research design, data collection, analysis or interpretation, or the writing of this manuscript.

References

- [1] E. Hadavandi, H. Shavandi, and A. Ghanbari, "A genetic fuzzy expert system for stock price forecasting," in *Proceedings of the 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 41–44, Yantai, China, August 2010.
- [2] C. Zheng and J. Zhu, "Research on stock price forecast based on gray relational analysis and ARMAX model," in *Proceedings of the 2017 International Conference on Grey Systems and Intelligent Services (GSIS)*, pp. 145–148, Stockholm, Sweden, August 2017.
- [3] A. Kulaglic and B. B. Üstündağ, "Stock price forecast using wavelet transformations in multiple time windows and neural networks," in *Proceedings of the 2018 3rd International Conference on Computer Science and Engineering (UBMK)*, pp. 518–521, Sarajevo, Bosnia, September 2018.
- [4] G. Xi, "A novel stock price forecasting method using the dynamic neural network," in *Proceedings of the 2018 International Conference on Robots & Intelligent System (ICRIS)*, pp. 242–245, Changsha, China, February 2018.
- [5] Y. Yu, S. Wang, and L. Zhang, "Stock price forecasting based on BP neural network model of network public opinion," in *Proceedings of the 2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pp. 1058–1062, Chengdu, China, June 2017.
- [6] J.-S. Chou and T.-K. Nguyen, "Forward forecast of stock price using sliding-window metaheuristic-optimized machine-learning regression," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3132–3142, 2018.
- [7] Y. Guo, S. Han, C. Shen, Y. Li, X. Yin, and Y. Bai, "An adaptive SVR for high-frequency stock price forecasting," *IEEE Access*, vol. 6, pp. 11397–11404, 2018.
- [8] J. Lee, R. Kim, Y. Koh, and J. Kang, "Global stock market prediction based on stock chart images using deep Q-network," *IEEE Access*, vol. 7, pp. 167260–167277, 2019.
- [9] J. Zhang, Y.-H. Shao, L.-W. Huang et al., "Can the exchange rate be used to predict the shanghai composite index?" *IEEE Access*, vol. 8, pp. 2188–2199, 2020.
- [10] Y. Guo, S. Han, C. Shen, Y. Li, X. Yin, and Y. Bai, "An adaptive SVR for high-frequency stock price forecasting," *IEEE Access*, vol. 6, pp. 11397–11404, 2018.
- [11] D. Lien Minh, A. Sadeghi-Niaraki, H. D. Huy, K. Min, and H. Moon, "Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network," *IEEE Access*, vol. 6, pp. 55392–55404, 2018.
- [12] I. I. Hassan, "Exploiting noisy data normalization for stock market prediction," *Journal of Engineering and Applied Sciences*, vol. 12, no. 1, pp. 69–77, 2017.
- [13] S. Jeon, B. Hong, and V. Chang, "Pattern graph tracking-based stock price prediction using big data," *Future Generation Computer Systems*, vol. 80, pp. 171–187, 2018.
- [14] E. Chong, C. Han, and F. C. Park, "Deep learning networks for stock market analysis and prediction: methodology, data representations, and case studies," *Expert Systems with Applications*, vol. 83, pp. 187–205, 2017.
- [15] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Using structured events to predict stock price movement: an empirical investigation," in *Proceedings of the EMNLP*, pp. 1–11, Doha, Qatar, October 2014.
- [16] M. Dang and D. Duong, "Improvement methods for stock market prediction using financial news articles," in *Proceedings of 2016 3rd National Foundation for Science and Technology Development Conference on Information and*

- Computer Science*, pp. 125–129, Danang City, Vietnam, September 2016.
- [17] B. Weng, M. A. Ahmed, and F. M. Megahed, “Stock market one-day ahead movement prediction using disparate data sources,” *Expert Systems with Applications*, vol. 79, pp. 153–163, 2017.
 - [18] Y. Yujun, L. Jianping, and Y. Yimei, “An efficient stock recommendation model based on big order net inflow,” *Mathematical Problems in Engineering*, vol. 2016, Article ID 5725143, 15 pages, 2016.
 - [19] S. O. Ojo, P. A. Owolawi, M. Mphahlele, and J. A. Adisa, “Stock market behaviour prediction using stacked LSTM networks,” in *Proceedings of the 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, pp. 1–5, Vanderbijlpark, South Africa, November 2019.
 - [20] S. Liu, G. Liao, and Y. Ding, “Stock transaction prediction modeling and analysis based on LSTM,” in *Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 2787–2790, Wuhan, China, May 2018.
 - [21] D. Wei, “Prediction of stock price based on LSTM neural network,” in *Proceedings of the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, pp. 544–547, Dublin, Ireland, October 2019.
 - [22] K. Chen, Y. Zhou, and F. Dai, “A LSTM-based method for stock returns prediction: a case study of China stock market,” in *Proceedings of the 2015 IEEE International Conference on Big Data (Big Data)*, pp. 2823–2824, Santa Clara, CA, USA, November 2015.
 - [23] Y. Yang and Y. Yang, “Hybrid method for short-term time series forecasting based on EEMD,” *IEEE Access*, vol. 8, pp. 61915–61928, 2020.
 - [24] A. H. Bukhari, M. A. Z. Raja, M. Sulaiman, S. Islam, M. Shoaib, and P. Kumam, “Fractional neuro-sequential ARFIMA-LSTM for financial market forecasting,” *IEEE Access*, vol. 8, p. 71326, 2020.
 - [25] S. Ma, L. Gao, X. Liu, and J. Lin, “Deep learning for track quality evaluation of high-speed railway based on vehicle-body vibration prediction,” *IEEE Access*, vol. 7, pp. 185099–185107, 2019.
 - [26] Y. Hu, X. Sun, X. Nie, Y. Li, and L. Liu, “An enhanced LSTM for trend following of time series,” *IEEE Access*, vol. 7, pp. 34020–34030, 2019.
 - [27] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
 - [28] Y. Yang, J. Li, and Y. Yang, “The cross-correlation analysis of multi property of stock markets based on MM-DFA,” *Physica A: Statistical Mechanics and Its Applications*, vol. 481, pp. 23–33, 2017.
 - [29] Y. Yujun, L. Jianping, and Y. Yimei, “Multiscale multifractal multiproperty analysis of financial time series based on Rényi entropy,” *International Journal of Modern Physics C*, vol. 28, no. 2, 2017.
 - [30] F. A. Gers and J. Schmidhuber, “Recurrent nets that time and count,” in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000*, vol. 3, pp. 189–194, Como, Italy, July 2000.
 - [31] Y. Wei, Z. Wang, M. Xu, and S. Qiao, “An LSTM method for predicting CU splitting in H. 264 to HEVC transcoding,” in *Proceedings of the IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, St. Petersburg, FL, USA, December 2017.
 - [32] C. Zhang, W. Ren, T. Mu, L. Fu, and C. Jia, “Empirical mode decomposition based background removal and de-noising in polarization interference imaging spectrometer,” *Optics Express*, vol. 21, no. 3, pp. 2592–2605, 2013.
 - [33] X. Zhou, H. Zhao, and T. Jiang, “Adaptive analysis of optical fringe patterns using ensemble empirical mode decomposition algorithm,” *Optics Letters*, vol. 34, no. 13, pp. 2033–2035, 2009.
 - [34] N. E. Huang, J. R. Yeh, and J. S. Shieh, “Ensemble empirical mode decomposition: a noise-assisted data analysis method,” *Advances in Adaptive Data Analysis*, vol. 1, no. 1, pp. 1–41, 2009.
 - [35] H. Zhang, F. Wang, D. Jia, T. Liu, and Y. Zhang, “Automatic interference term retrieval from spectral domain low-coherence interferometry using the EEMD-EMD-based method,” *IEEE Photonics Journal*, vol. 8, no. 3, pp. 1–9, Article ID 6900709, 2016.
 - [36] Y. Jiang, C. Tang, X. Zhang, W. Jiao, G. Li, and T. Huang, “A novel rolling bearing defect detection method based on bispectrum analysis and cloud model-improved EEMD,” *IEEE Access*, vol. 8, pp. 24323–24333, 2020.
 - [37] W. Wang, W. Peng, M. Tian, and W. Tao, “Partial discharge of white noise suppression method based on EEMD and higher order statistics,” *The Journal of Engineering*, vol. 2017, no. 13, pp. 2043–2047, 2017.
 - [38] Z. Wei, K. G. Robbersmyr, and H. R. Karimi, “An EEMD aided comparison of time histories and its application in vehicle safety,” *IEEE Access*, vol. 5, pp. 519–528, 2017.
 - [39] Y. Yang and Y. Yang, “Hybrid prediction method for wind speed combining ensemble empirical mode decomposition and Bayesian ridge regression,” *IEEE Access*, vol. 8, pp. 71206–71218, 2020.
 - [40] J. Sun and H. Sheng, “Applications of Ensemble Empirical mode decomposition to stock-futures basis analysis,” in *Proceedings of the 2010 2nd IEEE International Conference on Information and Financial Engineering*, pp. 396–399, Chongqing, China, September 2010.
 - [41] B. Al-Hnaity and M. Abbod, “A novel hybrid ensemble model to predict FTSE100 index by combining neural network and EEMD,” in *Proceedings of the 2015 European Control Conference (ECC)*, pp. 3021–3028, Linz, Austria, July 2015.
 - [42] N. E. Huang, Z. Shen, S. R. Long et al., “The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis,” *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1971, pp. 903–995, 1998.
 - [43] F. Jiang, Z. Zhu, and W. Li, “An improved VMD with empirical mode decomposition and its application in incipient fault detection of rolling bearing,” *IEEE Access*, vol. 6, pp. 44483–44493, 2018.
 - [44] Z. H. Wu and N. E. Huang, “Ensemble empirical mode decomposition: a noise-assisted data analysis method,” *Advances in Adaptive Data Analysis*, vol. 1, no. 1, pp. 1–14, 2009.