Research on stock strategy investment based on LSTM and EEMD models

Jianliang Zhang[1,*], Qirong Zhang[2]

[1]School of Mathematics and Statistics, Chongqing Jiaotong University, Chongqing, China, 402260

[2]School of International Education, Wuhan University of Technology, Wuhan, China, 430070

*Corresponding author: 2770483362@qq.com

Abstract. The high volatility and complexity of the stock market make price prediction a challenging time series problem. Traditional methods such as ARIMA perform well in capturing linear relationships but struggle to effectively address the nonlinear and non-stationary characteristics of stock prices. In recent years, Long Short-Term Memory (LSTM) networks have garnered widespread attention due to their ability to handle sequential data and long-term dependencies. However, a standalone LSTM model is still insufficient when dealing with the high nonlinearity and noise inherent in stock data. To address this, this paper proposes a hybrid model combining Ensemble Empirical Mode Decomposition (EEMD) and LSTM to improve prediction accuracy and stability. By decomposing stock price time series using EEMD, intrinsic mode functions (IMFs) and residual components of different frequencies are extracted. These components are then individually predicted using LSTM models, and the results are weighted and integrated to generate the final prediction. Experimental results demonstrate that the proposed hybrid model significantly outperforms traditional methods in both prediction accuracy and robustness, showcasing its important application value in stock market forecasting.

Keywords: LSTM, EEMD, stock strategy, time series prediction.

1. Introduction

Stock price prediction, as an important research topic in the financial domain, has extensive practical application value. Accurate stock price predictions can help investors formulate scientific investment strategies and provide financial institutions with effective risk management tools. However, stock market data is often highly nonlinear, non-stationary, and noisy, making stock price prediction a complex time series forecasting problem.

Traditional time series forecasting methods, such as Autoregression (AR), Moving Average (MA), and their combination ARIMA models, can handle linear relationships to some extent. However, they exhibit significant limitations when dealing with the complex nonlinear dynamics and volatile market environment of stock prices. In recent years, with the advancement of deep learning technologies, Long Short-Term Memory (LSTM) networks have become mainstream in time series forecasting due to their advantages in capturing long-term dependencies and processing sequential data [1]. LSTM, with its unique gating mechanisms, can effectively mitigate the vanishing gradient problem, thereby performing well in handling data with long time spans. However, standalone LSTM models still face challenges in improving prediction accuracy and stability when dealing with highly nonlinear and noisy stock data [2].

To address the accuracy and stability issues of LSTM models in forecasting complex time series, Yan Peng et al. (2019) further studied the impact of LSTM network structures on stock price prediction [3]. They conducted preprocessing steps such as interpolation, wavelet denoising, and normalization on stock data, then experimented with LSTM models of varying layers and hidden neuron counts. Results showed that an appropriately structured model significantly improved prediction accuracy, while excessive network layers increased computational complexity without notable performance gains. This highlights the critical importance of structural optimization in stock price forecasting.

Sun Ruiqi (2015) conducted trend forecasting for U.S. stock indices and compared the performance of BP Neural Networks, RNN, and LSTM in short-term stock price prediction [4]. The study demonstrated that LSTM outperformed BP and RNN in processing financial time series data, especially in capturing long-term dependencies. Additionally, Sun Ruiqi enhanced the LSTM model by introducing selective memory mechanisms and optimizing the network structure, further improving prediction accuracy and stability.

Research by Qing Yang et al. (2019) expanded the use of LSTM to predict global stock indices and emphasized its ability to handle volatile market data [5]. They integrated LSTM with other hybrid techniques to address noise and improve stability, demonstrating superior results in comparison to traditional statistical models.

Further applications of LSTM and hybrid models have been demonstrated in various fields, such as power generation prediction using EEMD-LSTM [6], highway traffic flow forecasting with EEMD-Att-GCN-LSTM [7], and monthly precipitation prediction [8]. These studies show the versatility and effectiveness of LSTM-based models in dealing with nonlinear and complex datasets.

Bouktif et al. (2018) utilized LSTM for electric load forecasting, integrating feature selection and genetic algorithms to enhance model performance [1]. Li Kejia (2018) also applied LSTM to futures price prediction, revealing the model's adaptability in financial forecasting [2]. In commodity price prediction, Di Hao et al. (2018) employed EEMD-LSTM-Adaboost, combining multiple algorithms to achieve high accuracy [9]. Finally, Ge Na et al. (2019) explored Prophet-LSTM for sales forecasting, highlighting the benefits of hybrid models in capturing seasonal trends [10].

2. The Basic Fundamentals of LSTM and EEMD

Long Short-Term Memory (LSTM) networks are a specialized form of Recurrent Neural Networks (RNNs) developed to mitigate the vanishing and exploding gradient issues that traditional RNNs face when modeling long-term dependencies. By incorporating gating mechanisms, LSTMs regulate information flow, enabling the effective capture of long-range dependencies in sequential data.

2.1. Structure of an LSTM Unit

An LSTM unit consists of the following key components:

Forget Gate: Determines which information from the previous cell state needs to be forgotten.

Input Gate: Controls the amount of current input information to be updated into the cell state.

Candidate State: Generates the candidate values for updating the cell state.

Output Gate: Decides which information to output.

2.2. Mathematical Representation

Symbols and Definitions in an LSTM Unit isshown as table1.

Table 1 Symbols and Definitions in an LSTM Unit

Symbol	Description	Symbol	Description	Symbol	Description
x_t	Current input vector.	$ht-1$	Previous hidden state.	$Ct-1$	Previous cell state.
h_t	Current hidden state.	C_t	Current cell state.	f_t	Forget gate output.
i_t	Input gate output.	$C\tilde{}t$	Candidate cell state.	o_t	Output gate output.
σ	Sigmoid function.	tanh	Hyperbolic tangent function.	W,b	Weight matrix and bias.

The computation process of an LSTM unit is as follows:

The forget gate determines how much information from the previous cell state C_{t-1} should be retained in the current step. The mathematical expression is:

$$f_t = \sigma\left(W_f \cdot \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_f\right) \tag{1}$$

The input gate controls the extent to which the current input x_t influences the cell state, determining how much new information will be written into the cell state. The mathematical expression is:

$$i_t = \sigma\left(W_i \cdot \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_i\right) \tag{2}$$

Meanwhile, the candidate cell state is generated as follows:

$$\tilde{C}_t = \tanh\left(W_C \cdot \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_C\right) \tag{3}$$

The current cell state is updated by combining the outputs of the forget gate and the input gate:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{4}$$

Here, $\odot$ denotes element-wise multiplication.

The output gate determines the current hidden state h_t. First, the output gate filters the cell state, and then the activation function is applied to generate the final hidden state:

$$o_t = \sigma\left(W_o \cdot \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_o\right) \tag{5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

3. Mathematical Principles of Ensemble Empirical Mode Decomposition (EEMD)

3.1. Introduction of EEMD

Ensemble Empirical Mode Decomposition (EEMD) is a signal decomposition method designed to address the mode mixing problem in traditional Empirical Mode Decomposition (EMD). By adding white noise to the original signal and averaging the results of multiple decompositions, EEMD provides more stable signal decomposition, thereby improving accuracy and robustness.

3.2. Working Principles

The basic steps of EEMD are as follows:

Step1:Add white noise of limited amplitude to the original signal to generate multiple noisy signals.

Step2:Apply EMD to each noisy signal to obtain a set of Intrinsic Mode Functions (IMFs).

Step3:Repeat steps 1 and 2 multiple times to generate multiple IMF sets.

Step4:Average the IMFs across all decompositions to obtain the final IMF set and residual.

3.3. Symbols and Definitions

Symbols and Definitions in EEMD isshown as table 2.

Table 2 Symbols and Definitions in EEMD

Symbol	Description	Symbol	Description	Symbol	Description
x_t	Current input vector.	$ht-1$	Previous hidden state.	$Ct-1$	Previous cell state.
h_t	Current hidden state.	C_t	Current cell state.	f_t	Forget gate output.
i_t	Input gate output.	$C\tilde{}t$	Candidate cell state.	o_t	Output gate output.
σ	Sigmoid function.	tanh	Hyperbolic tangent function.	W,b	Weight matrix and bias.

3.4. Mathematical Principles

The EEMD process can be mathematically described as follows:
White noise is added to the original signal to generate the noisy signal in the j-th trial:

$$X_j(t) = X(t) + w_j(t), \quad j = 1,2,\dots,N \tag{7}$$

where $w_j(t)$ is white noise with zero mean and finite energy.
EMD is applied to each noisy signal $X_j(t)$ to obtain the IMFs:

$$X_j(t) = \sum_{k=1}^{K} I\,MF_{j,k}(t) + r_j(t) \tag{8}$$

where K is the number of IMFs, and $r_j(t)$ is the residual.
The IMFs obtained from all decompositions are averaged to yield the final IMF set:

$$\overline{IMF_k}(t) = \frac{1}{N}\sum_{j=1}^{N} I\,MF_{j,k}(t), \quad k = 1,2,\dots,K \tag{9}$$

The residuals from all decompositions are averaged to obtain the final residual:

$$\overline{r}(t) = \frac{1}{N}\sum_{j=1}^{N} r_j\,(t) \tag{10}$$

The original signal is decomposed into multiple IMFs and a residual using EEMD:

$$X(t) = \sum_{k=1}^{K} \overline{IMF_k}\,(t) + \overline{r}(t) \tag{11}$$

4. Model Solution Analysis

4.1. Data Introduction

The experimental data is sourced from the Baostock platform, an open-source financial data service platform focused on providing high-quality data for the Chinese A-share market. The Baostock database contains detailed historical trading information, including stock opening prices, closing prices, highest prices, lowest prices, trading volumes, and more. Additionally, it provides multi-dimensional financial data of listed companies. Due to its timely data updates and comprehensive coverage, Baostock is widely used in academic research and financial modeling practices.

4.2. Introduction to Evaluation Indicators

In statistical analysis, commonly used error evaluation metrics include Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). Their formulas are as follows:
Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{12}$$

MSE quantifies the average squared difference between predicted and actual values, with lower values indicating better model performance.

Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{13}$$

RMSE, being the square root of MSE, retains the same unit as the original data, providing an intuitive measure of error magnitude.

Mean Absolute Error (MAE):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{14}$$

MAE calculates the average absolute difference between predicted and actual values, offering robustness against outliers compared to MSE.

Coefficient of Determination (R^2 Score):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{15}$$

R^2 measures the proportion of variance in the data explained by the model. Its range is [0, 1], with values closer to 1 indicating better model performance.

Where y_i represents the actual values, $\hat{y}_i$ represents the predicted values, $\bar{y}$ is the mean of the actual values, and n denotes the sample size.

4.3. Experimental Results and Analysis

First, the stock price time series is decomposed using EEMD to obtain multiple IMFs and a residual component.

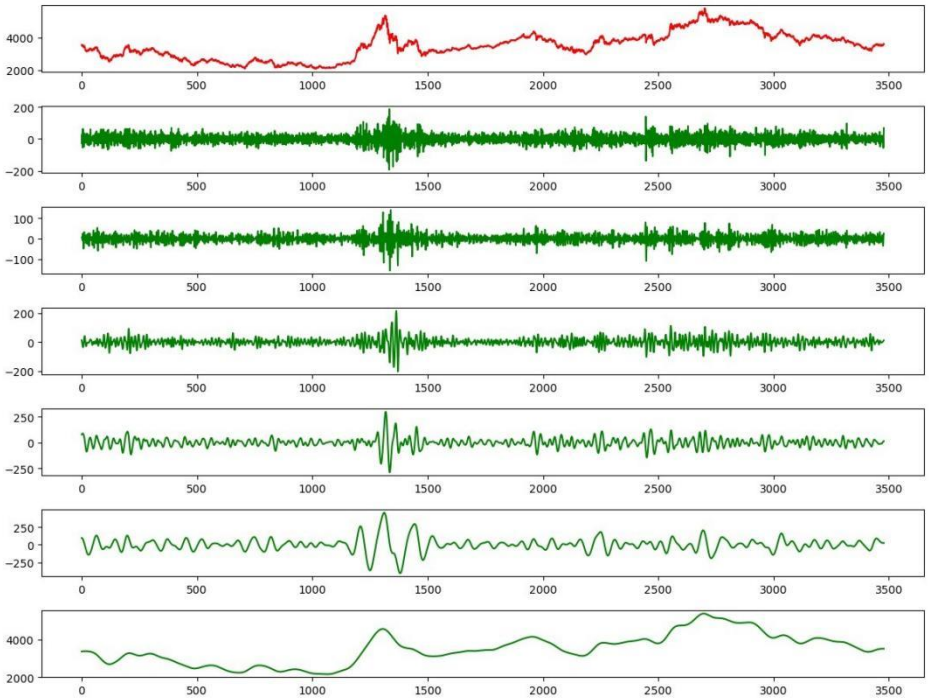

Figure 1 Illustration of the EEMD Decomposition Process

Figure 1 illustrates the decomposition of a stock price time series using the Ensemble Empirical Mode Decomposition (EEMD) method. This process yields multiple Intrinsic Mode Functions (IMFs) and a residual component. From top to bottom, the figure presents the original signal followed by the decomposed frequency components in sequence. It is evident that the higher-frequency IMFs capture

short-term fluctuations, while the lower-frequency IMFs and the residual component represent long-term trends.

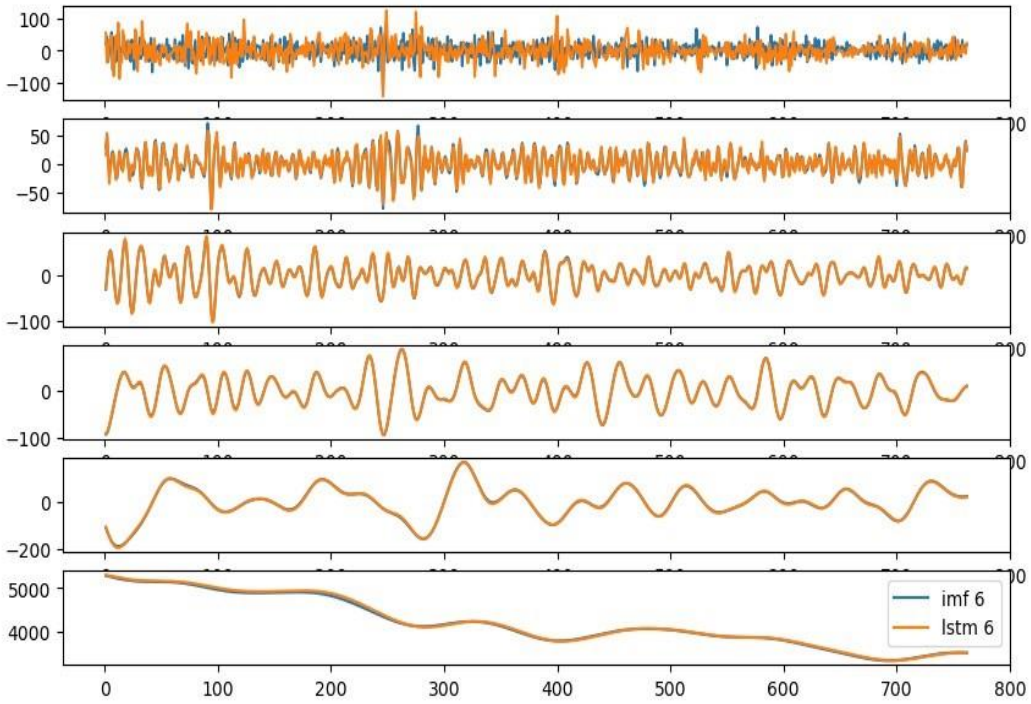

Figure 2 Comparison of LSTM Model Predictions for IMFs

Figure 2 compares the LSTM model predictions for each IMF component. The horizontal axis represents the time series index, while the vertical axis shows the corresponding signal amplitude or predicted value. It can be seen that the LSTM model demonstrates strong fitting capability in capturing signal variations across different frequency components.

• Finally, the prediction results from each LSTM model are integrated through weighted averaging to generate the final prediction.

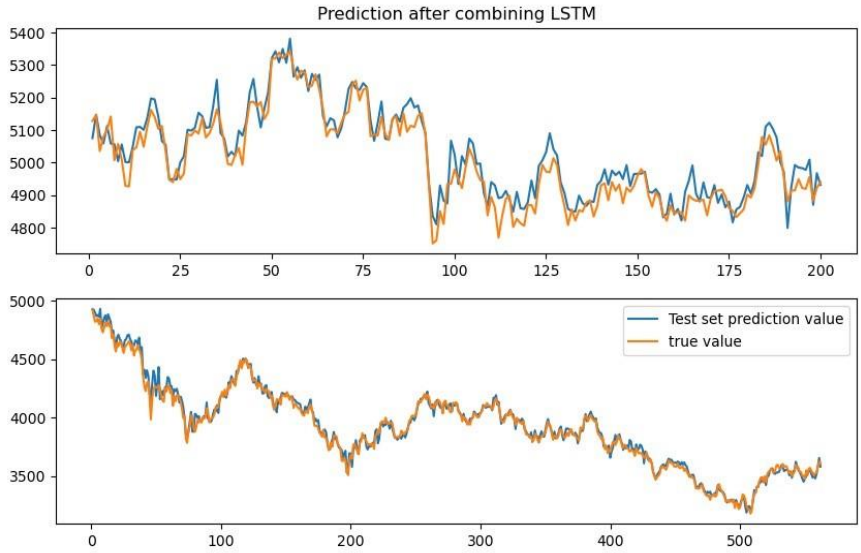

Figure 3 Comparison Between Predicted and Actual Values

As shown in Figure 3, the test set prediction value (blue) are obtained by integrating the predictions of all LSTM sub-models trained on different IMF components generated through EEMD. The true values (orange) serve as a benchmark for comparison. The horizontal axis represents the time series index, while the vertical axis denotes the price amplitude. By leveraging EEMD to decompose the original stock price series into multiple frequency components and utilizing LSTM to capture the nonlinear dependencies within each component, the hybrid model effectively enhances the prediction

accuracy. The strong alignment between the predicted and actual values further demonstrates the effectiveness of this EEMD-LSTM approach in stock price forecasting.

We randomly selected several stocks for experimentation, and the experimental results are as follows:

Table 3 Results Table

Stock Code	MSE	RMSE	MAE	R^2 Score
sh.600071	0.000423	0.020567	0.017225	0.728607
sh.600056	0.000145	0.012036	0.010217	0.441664
sh.600004	0.000326	0.018060	0.014756	0.924368
sh.600015	0.000296	0.017195	0.013756	0.977405
sh.600163	0.000344	0.018537	0.015468	0.862050
sh.600136	0.003178	0.056372	0.054036	-5.745165
sh.600078	0.000471	0.021703	0.018324	0.917271
sh.600089	0.000244	0.015616	0.012585	0.907729
sh.600152	0.000190	0.013792	0.011128	0.816380
sh.600009	0.001277	0.035738	0.028847	0.572543

Through empirical analysis, the effectiveness of the proposed method was validated. A random selection of stocks was used for prediction.

Lowest Error: For stock "sh.600056," the MSE was 0.000145, RMSE was 0.012036,and MAE was0.010217, indicating the lowest prediction error and the best prediction performance for this stock.

Highest Error: For stock "sh.600136," the MSE significantly increased to 0.003178, RMSE was 0.056372, and MAE was 0.054036, demonstrating relatively poor prediction performance. A potential reason for this poor performance is the risk of delisting faced by this stock, with certain intervals showing consistent limit-down trends. Under such circumstances, the LSTM+EEMD approach yielded higher MSE and other error metrics.

Coefficient of Determination (R2 Score):

Highest R2: For "sh.600015", the R2 score reached 0.977405, indicating that the model could explain approximately 97.74.

Lowest R2: For "sh.600136", the R2 score was -5.745165. A negative R2 implies that the model's prediction was even worse than a simple mean prediction, showing extremely poor performance. For other stocks, most R2 scores ranged between 0.4 and 0.9, highlighting significant differences in the model's predictive capability across different stocks.

Table 4 Average Error Metrics of Different Models

Model	MSE	RMSE	MAE
EEMD+LSTM	0.000689	0.022962	0.019634
LSTM	0.880936	0.817799	0.647720
CNN-LSTM	1.203261	0.945474	0.767614
Transformer	51.690321,	5.757597,	5.487630

The combined EEMD+LSTM approach significantly outperforms standalone LSTM, CNN-LSTM, and Transformer models across various evaluation metrics. This demonstrates that the preprocessing step (EEMD) plays a crucial role in denoising and extracting effective signals, thereby enabling LSTM to make more accurate predictions.

Although CNN is theoretically capable of extracting local features, in this experiment, the extracted features did not effectively improve prediction performance. This may be due to the characteristics of the dataset or inappropriate network architecture design, which instead introduced more noise or overfitting issues.

In this experiment, preprocessing the time series data with EEMD before using LSTM for prediction resulted in exceptionally low errors. In contrast, directly applying LSTM, CNN-LSTM, or

Transformer models yielded significantly worse performance. This indicates that data preprocessing techniques such as EEMD are essential for tasks of this nature, as they effectively reduce noise and simplify signals, thereby substantially improving prediction accuracy.

5. Conclusion

This paper proposed a hybrid model combining Ensemble Empirical Mode Decomposition (EEMD) and Long Short-Term Memory (LSTM) networks for stock price prediction. Empirical analysis demonstrated the model's effectiveness in capturing the nonlinear and non-stationary characteristics of stock prices.

One of the key advantages of the EEMD-LSTM approach is the interpretability of the Intrinsic Mode Functions (IMFs). The IMFs obtained through EEMD are typically arranged in descending order of frequency, with each IMF representing oscillatory components within different frequency ranges of the original signal. This decomposition allows researchers to separately analyze high-frequency and low-frequency components, leading to a better understanding of the spectral characteristics of the signal. Although EEMD alleviates the mode mixing issue present in traditional Empirical Mode Decomposition (EMD) to some extent, in certain cases, oscillations of different time scales may still mix within the same IMF, which can affect the interpretability of the components.

Future Research Directions

• Model Parameter Optimization: Further optimize model parameters to improve prediction accuracy. For instance, employing optimization algorithms like Genetic Algorithm (GA) to fine-tune LSTM hyperparameters, which has shown potential for enhancing prediction performance in related studies.

• Incorporating Exogenous Variables: Introduce exogenous variables, such as macroeconomic indicators and industry data, to enhance the model's ability to predict stock price trends. Multi-variable input LSTM models have demonstrated strong performance in time series forecasting.

• Cross-Domain Applications: Apply the EEMD-LSTM hybrid model to time series forecasting in other domains, such as electricity load forecasting. Previous studies combining EEMD and LSTM for electricity load forecasting achieved high predictive accuracy, indicating the broad applicability of this method across various fields.

By applying the model to different fields and further optimizing its structure, the hybrid EEMD-LSTM model is expected to play a more significant role in time series forecasting.

References

[1] Bouktif, S., Fiaz, A., Ouni, A. L., et al. "Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches." Energies, vol. 11, no. 7, 2018, p. 1636, doi:10.3390/en11071636.

[2] Li Kejia. Futures Price Prediction Based on Long Short-Term Memory Model. Doctoral dissertation, University of Science and Technology of China, 2018.

[3] Yan Peng, Yuhong Liu, and Rongfen Zhang. "Modeling and Analysis of Stock Price Forecast Based on LSTM". In: Computer Engineering and Applications 55.11 (2019), pp. 209–212.

[4] Ruiqi Sun. "Research on the Trend Prediction Model of US Stock Index Price Based on LSTM Neural Network". In: Capital University of Economics and Business (2015), pp. 1–57.

[5] Qing Yang et al. "Global Stock Index Prediction Based on LSTM Neural Networks". In: Journal of Financial Studies 30 (2019), pp. 120–130.

[6] Yang, F., He, Q., Zhan, Y., et al. Power generation prediction of small hydropower stations based on EEMD-LSTM [C] // 2024 International Conference on Energy Engineering, Beijing, 2024: 24-30.

[7] Sun Jinxin, et al. "Highway Traffic Flow Prediction Based on EEMD-Att-GCN-LSTM." Industrial Control Computer, vol. 37, no. 12, 2024, pp. 39-41.

[8] Qin Zhuang. “Research on Monthly Precipitation Prediction in Shijiazhuang Based on the EEMD-LSTM Combined Model.” Water Conservancy Science and Technology and Economy, vol. 30, no. 2, 2024, pp. 105-108, doi:10.3969/j.issn.1006-7175.2024.02.021.

[9] Di Hao, Zhao Xuejun, and Zhang Zili. “Commodity Price Forecasting Based on EEMD-LSTM-Adaboost.” Statistics and Decision, vol. 34, no. 13, 2018, pp. 72-76.

[10] Ge Na, Sun Lianying, Shi Xiaoda, et al. “Research on Sales Forecasting Using the Prophet-LSTM Combined Model.” Computer Science, vol. 46, no. S1, 2019, pp. 446-451.