PROJECT:STOCK PRICE PREDICTION

PHASE:2

Abstract:

Stock price data have the characteristics of time series. At the same time, based on machine learning long short-term memory (LSTM) which has the advantages of analyzing relationships among time series data through its memory function, we propose a forecasting method of stock price based on CNN-LSTM. In the meanwhile, we use MLP, CNN, RNN, LSTM, CNN-RNN, and other forecasting models to predict the stock price one by one. Moreover, the forecasting results of these models are analyzed and compared. The data utilized in this research concern the daily stock prices from July 1, 1991, to August 31, 2020, including 7127 trading days. In terms of historical data, we choose eight features, including opening price, highest price, lowest price, closing price, volume, turnover, ups and downs, and change. Firstly, we adopt CNN to efficiently extract features from the data, which are the items of the previous 10 days. And then, we adopt LSTM to predict the stock price with the extracted feature data. According to the experimental results, the CNN-LSTM can provide a reliable stock price forecasting with the highest prediction accuracy. This forecasting method not only provides a new research idea for stock price forecasting but also provides practical experience for scholars to study financial time series data.

Introduction:

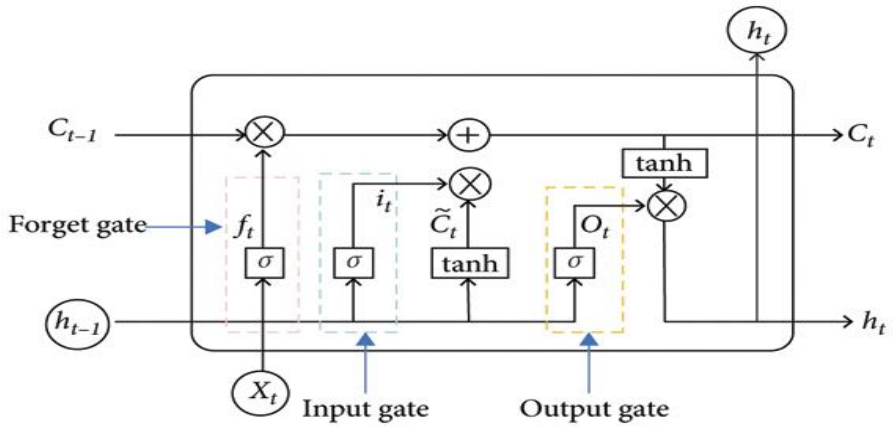
The change trend of the stock price has always been identified as a very important problem in the economic field.. Stock prices are affected by various internal and external factors, such as domestic and foreign economic environment, international situation, industry prospect, financial data of listed companies, and stock market operation. Thus, the forecasting method also has different emphasis.

It has certain limitations to predict stock price trend with single simply using the linear time series forecasting model or neural network model. At present, combining the advantages of various methods and using various best algorithms to improve the hybrid method is the development trend of financial time series deep learning [12]. Therefore, in order to make the best of the time series characteristics of data series, deeply mine the data features, and improve the accuracy of stock price forecasting, this paper proposes a stock price forecasting method based on CNN-LSTM for the stock closing price of the next day forecasting. Combining the advantages of convolutional neural networks (CNN) that can extract effective features from the data, and long short-term memory (LSTM) which can not only find the interdependence of data in time series data, but also automatically detect the best mode suitable for relevant data, this method can effectively improve the accuracy of stock price forecasting. The CNN-LSTM model uses CNN to extract the features of the input time data and uses LSTM to predict the stock closing price on the next day. In order to verify the effectiveness of the model, this paper uses the daily transaction data of 7127 trading days from July 1, 1991, to August 31, 2020, in which the first 6627 trading days data are the training set and the last 500 trading days data are the test set.

CNN-LSTM

CNN-LSTM Model

CNN has the characteristic of paying attention to the most obvious features in the line of sight, so it is widely used in feature engineering. LSTM has the characteristic of expanding according to the sequence of time, and it is widely used in time series. According to the characteristics of CNN and LSTM, a stock forecasting model based on CNN-LSTM is established. The model structure diagram is shown in Figure 1, and the main structure is CNN and LSTM, including input layer, one-dimensional convolution layer, pooling layer, LSTM hidden layer, and full connection layer.



CNN:

CNN is a network model proposed by Lecun et al. in 1998 . CNN is a kind of feedforward neural network, which has good performance in image processing and natural language processing [23]. It can be effectively applied to the forecasting of time series. The local perception and weight sharing of CNN can greatly reduce the number of parameters, thus improving the efficiency of model learning . CNN is mainly composed of two parts: convolution layer and pooling layer. Each convolution layer contains a plurality of convolution kernels, and its calculation formula is shown in formula . After the convolution operation of the convolution layer, the features of the data are extracted, but the extracted feature dimensions are very high, so in order to solve this problem and reduce the cost of training the network, a pooling layer is added after the convolution layer to reduce the feature dimension:

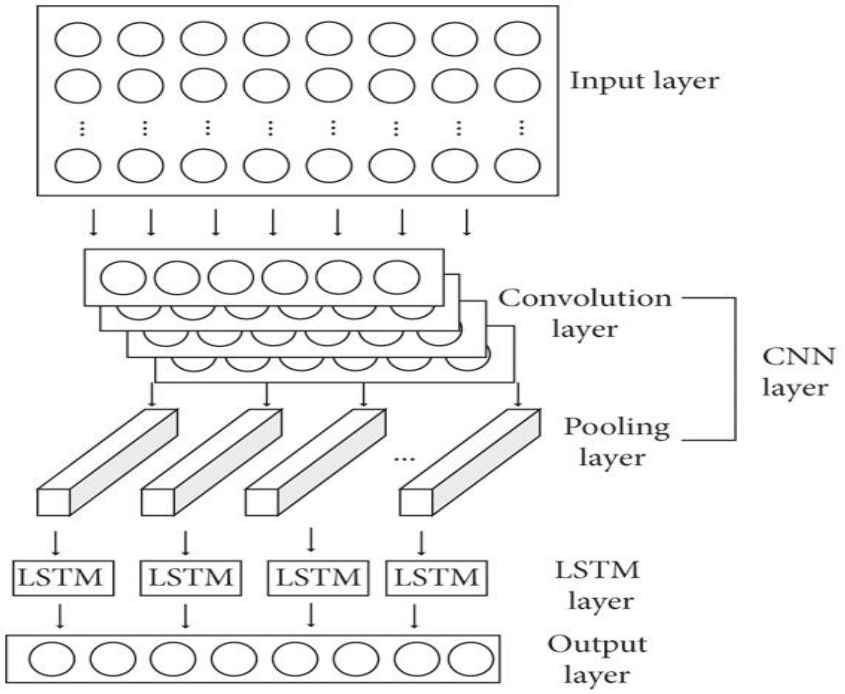
$$lt = \tanh(xt * kt + bt)$$

where lt represents the output value after convolution, tanh is the activation function, xt is the input vector, kt is the weight of the convolution kernel, and bt is the bias of the convolution kernel.

LSTM:

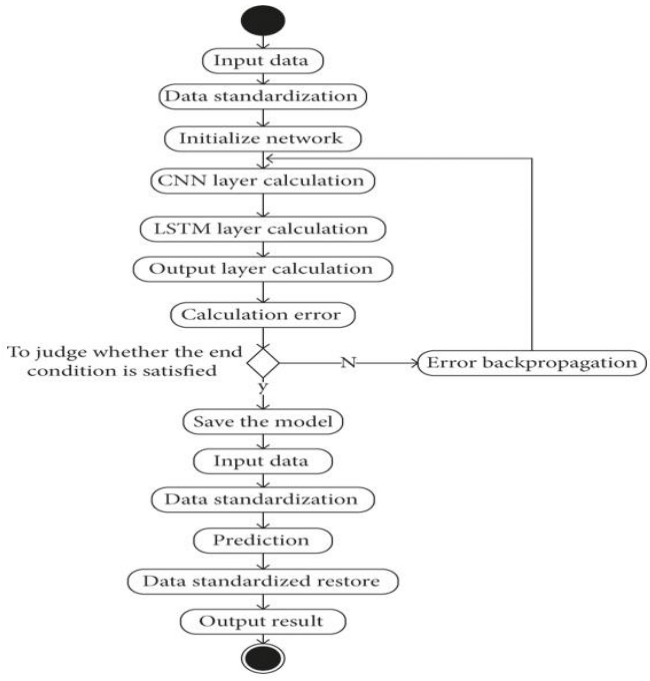
LSTM is a network model proposed by Schmidhuber et al. In 1997 [25]. LSTM is a network model designed to solve the longstanding problems of gradient explosion and gradient disappearance in RNN [26, 27]. It has been widely used in speech recognition, emotional analysis, and text analysis, as it has its own memory and can make relatively accurate forecasting [28, 29]. In recent years, it has also been adopted in the field of stock market

forecasting [30–32]. There is only one repeating module in a standard RNN, and its internal structure is simple. It is usually a tanh layer. However, four of the LSTM modules are similar to the standard RNN modules, and they operate in a special interactive manner [33, 34]. The LSTM memory cell consists of three parts: the forget gate, the input gate, and the output gate.



CNN-LSTM Training and Prediction Process:

The CNN-LSTM process of training and prediction process is



The main steps are as follows :

(1) Input data: input the data required for CNN-LSTM Training.

(2) Data standardization: as there is a large gap in the Input data, in order to train the model better, the z-Score standardization method is adopted to standardize the input data, as shown in the following Formula:

$$Yi = xi - x/S$$

$$Xi = yi * s + x$$

Where yi is the standardized value, xi is the input data, x is the average of the input data, and s is the Standard deviation of the input data.

(3) Initialize network: initialize the weights an biase of each layer of the CNN-LSTM.

(4) CNN layer calculation: the input data are successively passed through the convolution layer and Pooling layer in the CNN layer, the feature extraction of the input data is carried out, and the output value is obtained.

(5) LSTM layer calculation: the output data of the CNN layer are calculated through the LSTM layer, and the output value is obtained.

(6) Output layer calculation: the output value of the LSTM layer is input into the full connection layer to get the output value.

(7) Calculation error: the output value calculated by the output layer is compared with the real value of this group of data, and the corresponding error is obtained.

(8) To judge whether the end condition is satisfied:The conditions for the end are to complete a predetermined number of cycles, the weight is lower than a certain threshold, and the error rate of the forecasting is lower than a certain threshold. If one of the conditions for the end is met, the training will be completed, update the entire CNN-LSTM network, and go to step 10; otherwise, go to step 9.

(9) Error backpropagation: propagate the calculated error in the opposite direction, update the weight and bias of each layer, and go to step 4 to continue to train the network.

(10) Save the model: save the trained model for of forecasting.

(11) Input data: input the input data required for the forecasting.

(12) Data standardization: the input data are standardized according to formula.

(13) Forecasting: input the standardized data into the trained model of CNN-LSTM, and then get the corresponding output value.

(14) Data standardized restore: the output value obtained through the model of CNN-LSTM is the standardized value, and the standardized value is restored to the original value. As shown in the following formula.

(15) Output result: output the restored results to complete the forecasting process.

Experiments:

In order to prove the effectiveness of CNN-LSTM, we compared this method with MLP, CNN, RNN, LSTM, and CNN-RNN using the same training set and test set data under the same operating environment. All the experiments are carried out under the running environment of

Intel i7-4700H 2.6 GHz, 12 GBs of RAM, 500 GBs of hard disk and windows 10. According to the influence factors, including the opening price, highest price, lowest price, closing price,volume, turnover, ups and downs, and change, the next day's closing price is predicted.

Data:

In this experiment, the Shanghai Composite Index (000001) is selected as the experimental data. The daily trading data of 7127 trading days from July 1, 1991, to August 31, 2020, are obtained from the wind database. Each piece of data contains eight items, namely, opening price, highest price, lowest price, closing price, volume, turnover, ups and downs, and change. Take the data of the first 6627 trading days as training set and the data of the last 500 trading days as test set.

Table 1: Partial sample data.

Date.	Opening price	Highest price	Lowest price.	Closing price	Volume (share)	Turnover (RMB).	Ups and downs	Change (%)
1991/7/1	136.64.	138.62	136.56	136.85	2294000	12469884.	−0.71	−0.5161
1991/7/2	135.91	135.96.	135.69.	135.96.	283800	3794100.	−0.89	−0.6503
1991/7/3.	135.28.	135.96	134.98	135.27	271500	1818504.	−0.69	−0.5075
1991/7/4.	136.63.	136.63.	134.19.	136.63	1339400	8095138.	1.36	1.0054
1991/7/5.	136.01	137.68	135.9	135.96.	1454000	9394861	−0.67.	−0.4904

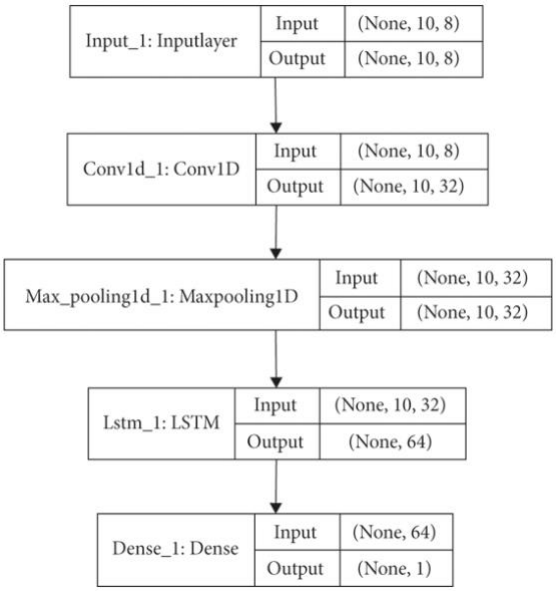
Implementation of CNN-LSTM:

The parameter setting of the CNN-LSTM for this experiment is shown in Table 2.

Table 2: Parameter setting of CNN-LSTM.

Parameters	Value
Convolution layer filters	32
Convolution layer kernel_size.	1
Convolution layer activation function.	tanh
Convolution layer padding.	Same
Pooling layer pool_size.	1
Pooling layer padding	Same
Pooling layer activation function.	Relu
Number of hidden units in LSTM layer	64
LSTM layer activation function.	tanh
Time_step.	10
Batch_size.	64

Learning rate.	0.001
Optimizer.	Adam
Loss function.	mean_absolute_error
Epochs.	100



According to the parameter setting of CNN-LSTM network, we can know that the specific model is constructed as follows: the input training set data is a three-dimensional data vector (None, 10, 8), in which 10 is the size of the time_step and 8 is the 8 features of the input dimension. First, the data enter the one-dimensional convolution layer to further extract features and obtain a three-dimensional output vector (None, 10, 32), in which 32 is the size of the convolution layer filters. Next, the vector enters the pooling layer, and a three-dimensional output vector (None, 10, 32) is also obtained. And then, the output vector enters the LSTM layer for training, and the output data (None, 64) after training enter another layer of full connection layer to get the output value; 64 is the number of hidden units in the LSTM layer. The specific CNN-LSTM model structure is shown in Figure.

Results

After using the processed training set data to train MLP, CNN, RNN, LSTM CNN-RNN, and CNN-LSTM, respectively, the model completed by training is used to predict the test set data, and the real value is compared with the predicted value as shown in Figures 5-10.

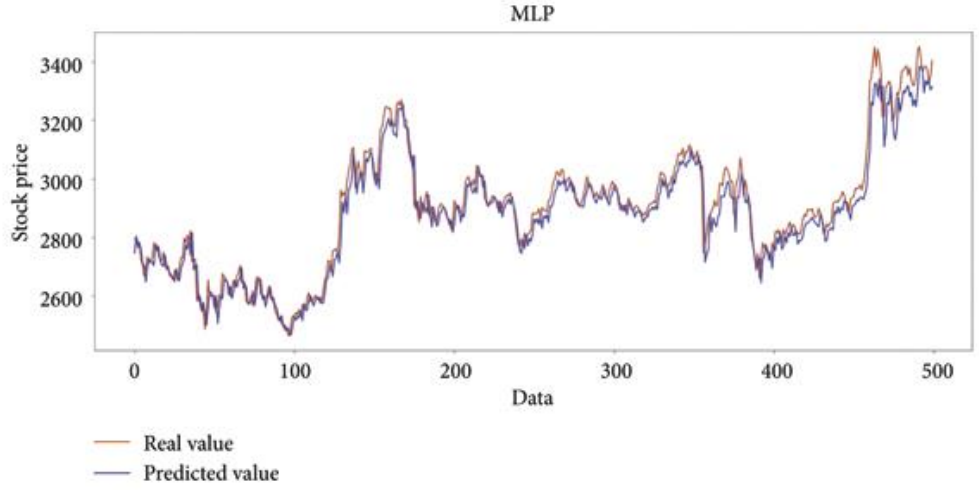


Figure 5:Comparison of the predicted value and the real value for MLP.

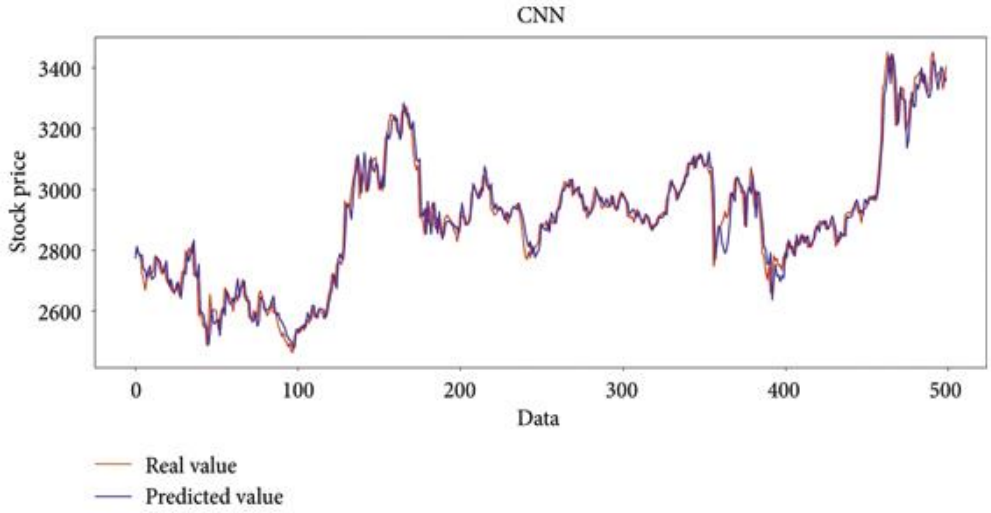


Figure 6:Comparison of the predicted value and the real value for CNN.

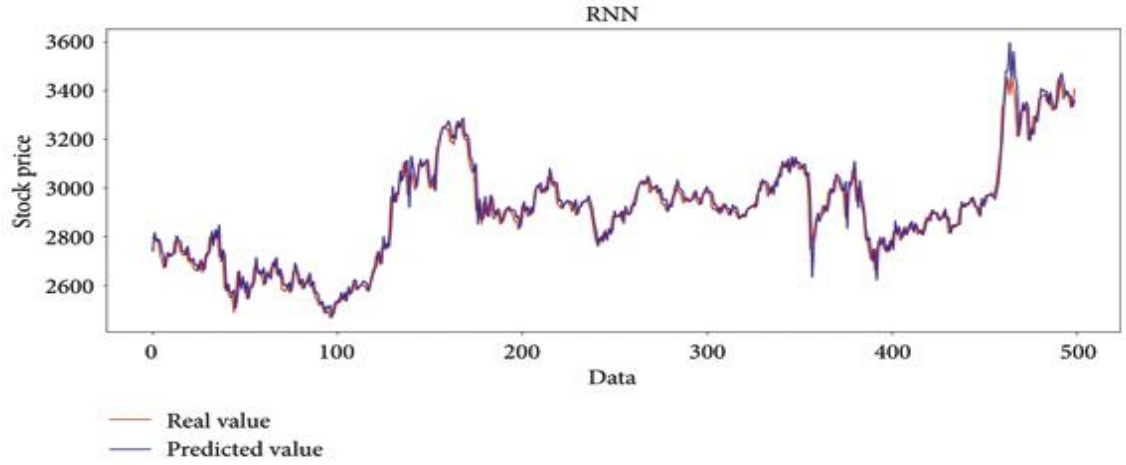


Figure 7:Comparison of the predicted value and the real value for RNN.

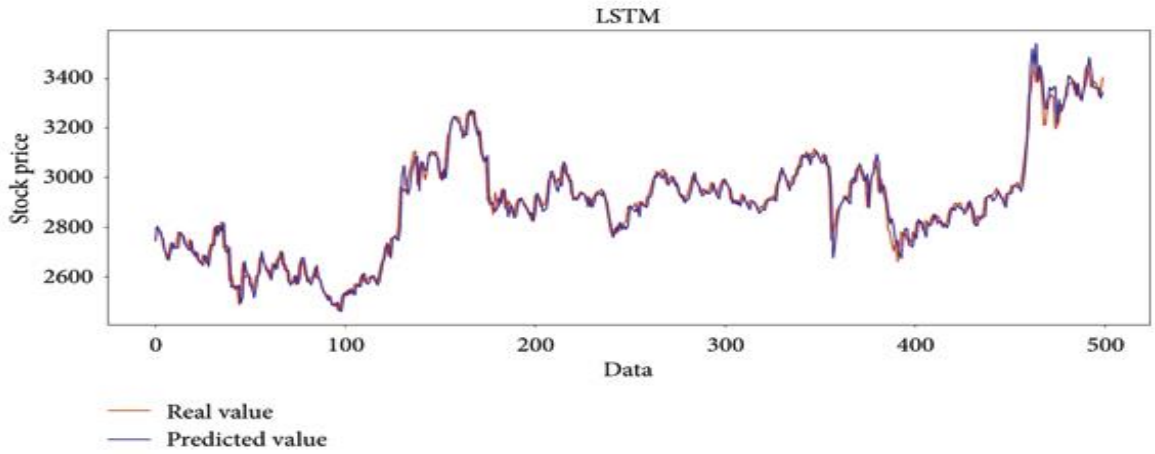


Figure 8:Comparison of the predicted value and the real value for LSTM.

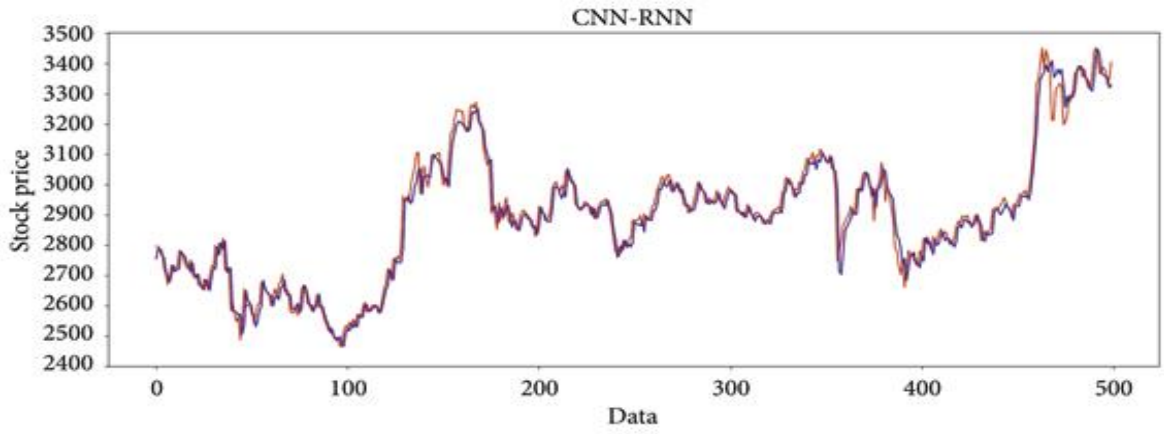


Figure 9:Comparison of the predicted value and the real value for CNN-RNN.

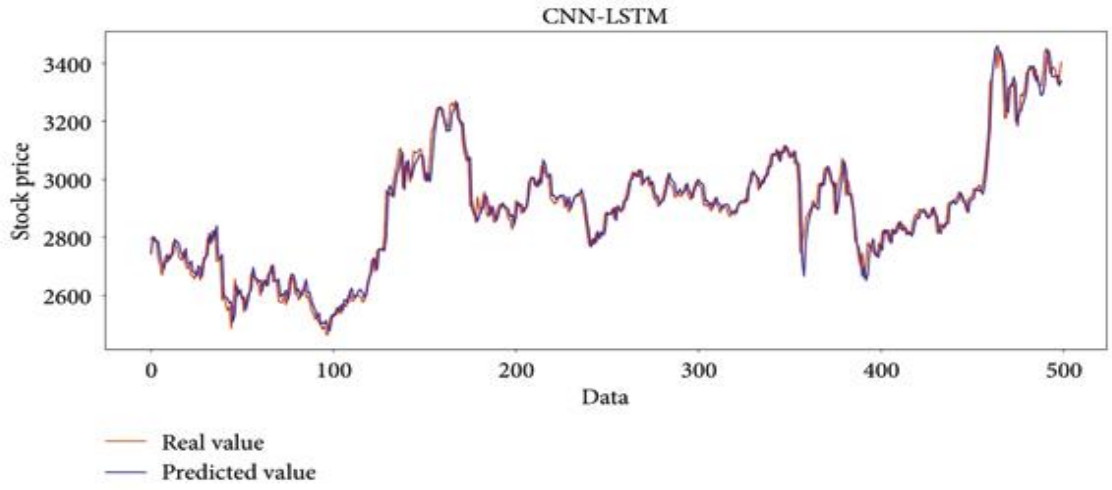


Figure 10:Comparison of the predicted value and the real value for CNN-LSTM.

According to the predicted value and real value of each method, the evaluation index of each method can be calculated, and the comparison results of the six methods are shown in Table 3 and Figures 11–13.

Table 3: Comparison of nine methods evaluation indexes.

Method	MAE	RMSE	R2
MLP	37.584	49.799	0.9442
CNN	30.138	42.967	0.9585
RNN	29.916	42.957	0.9593
LSTM	28.712	41.003	0.9622
CNN-RNN	28.285	40.538	0.9630
CNN-LSTM	27.564	39.688	0.9646

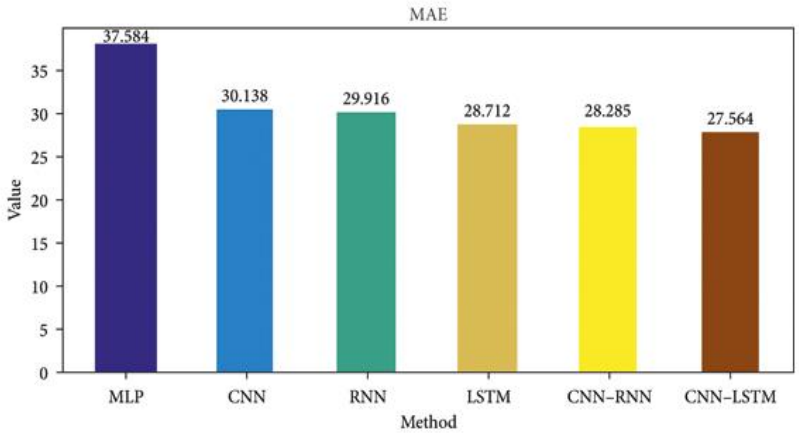


Figure 11:The result of MAE comparison among different methods.

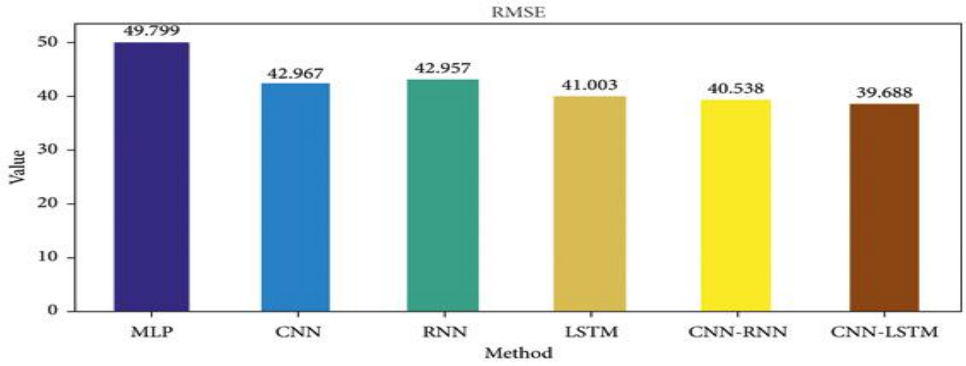


Figure 12:The result of RMSE comparison among different methods.

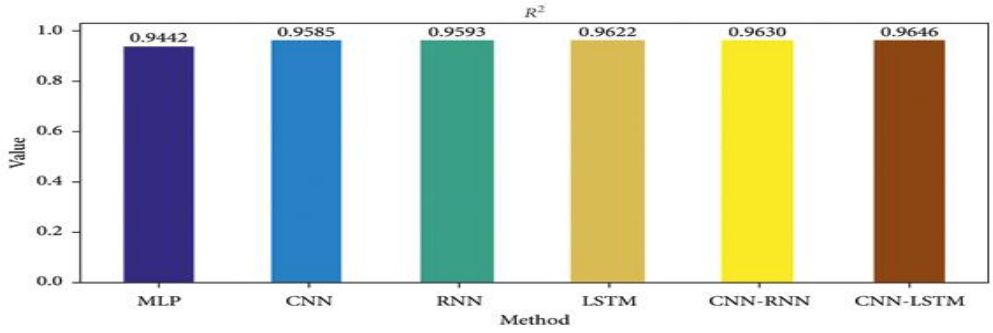


Figure 13:The result of R^2 comparison among different methods.

From Table 3 and Figures 10–12, the MAE and RMSE of MLP are the largest and R^2 is the smallest, while the MAE and RMSE of CNN-LSTM are the smallest, R^2 is the largest, and the closest is 1.

By comparing LSTM with RNN, the MAE and RMSE of LSTM decrease, R2 increases by 0.3%, MAE decreases from 29.916 to 28.712 by 4.0%, and RMSE decreases from 42.957 to 41.003 by 4.5%, so LSTM was better than RNN. However, the error measurement indexes MAE and RMSE of CNN-LSTM are the smallest, and the maximum R^2 is close to 1. Compared with LSTM, after CNN layer, MAE and RMSE of CNN-LSTM proposed in this paper are lower than those without CNN layer; R^2 has a certain improvement; MAE decreases by 4.0%, from 28.712 to 27.564; RMSE decreases by 3.2%, from 41.003 to 39.688; and R^2 increases by 0.2%. It shows that the forecasting performance of LSTM can be effectively improved by extracting data features through CNN.

The results show that the performance of CNN-LSTM is the best among the six methods. In terms of forecasting accuracy, MAE is 27.564 and RMSE is 39.688, which is the smallest among the six forecasting models and has high forecasting accuracy, in terms of forecasting performance, and the R^2 of CNN-LSTM is 0.9646, which is improved by 2.2%, 0.6%, 0.5%, and 0.2%, respectively, compared with the other four methods. Therefore, the CNN-LSTM proposed in this paper is superior to the other four comparative models in terms of fitting degree and error value. It can well predict the closing price of the next day and provide a reference for investors' investment.

Conclusion:

According to the chronological characteristics of stock price data, this paper proposes a CNN-LSTM to predict the stock closing price of the next day. The method uses opening price, highest price, lowest price, closing price, volume, turnover, ups and downs, and change of the stock data as the input, making full use of the time sequence characteristics of the stock data. CNN is used to extract the features of the input data. LSTM is used to learn the extracted feature data and predict the closing price of the stock the next day. This paper takes the relevant data of the Shanghai Composite Index as an example to verify the experimental results. The experimental results show that the CNN-LSTM has the highest forecasting accuracy and the best performance compared with the MLP, CNN, RNN, LSTM, and CNN-RNN. MAE and RMSE are the smallest of all methods, and R^2 is close to 1. CNN-LSTM is suitable for the forecasting of stock prices and can provide a relevant reference for investors to maximize investment returns. CNN-LSTM also provides the proposal of practical experience for people's research on financial time series data. However, the model still has some shortcomings. For example, it only considers the impact of stock price data on closing prices and fails to integrate emotional factors such as news and national policy into the forecast. Our future research work is mainly to increase the sentiment analysis of stock-related news and national policies, so as to ensure the accuracy of stock forecast.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was funded by the Soft Science Project of Hebei Province, Grant 205576142D, and Humanities and Social Science Research Project of Hebei Education Department, Grant SD201010.