REVIEW

A CNN-BiLSTM-AM method for stock price prediction

Wenjie Lu[1,2] · Jiazheng Li[3] · Jingyang Wang[3] · Lele Qin[1]

Abstract

In recent years, with the rapid development of the economy, more and more people begin to invest into the stock market. Accurately predicting the change of stock price can reduce the investment risk of stock investors and effectively improve the investment return. Due to the volatility characteristics of the stock market, stock price prediction is often a nonlinear time series prediction. Stock price is affected by many factors. It is difficult to predict through a simple model. Therefore, this paper proposes a CNN-BiLSTM-AM method to predict the stock closing price of the next day. This method is composed of convolutional neural networks (CNN), bi-directional long short-term Memory (BiLSTM), and attention mechanism (AM). CNN is used to extract the features of the input data. BiLSTM uses the extracted feature data to predict stock closing price of the next day. AM is used to capture the influence of feature states on the stock closing price at different times in the past to improve the prediction accuracy. In order to prove the effectiveness of this method, this method and other seven methods are used to predict the stock closing price of the next day for 1000 trading days of the Shanghai Composite Index. The results show that the performance of this method is the best, MAE and RMSE are the smallest (which are 21.952 and 31.694). R^2 is the largest (its value is 0.9804). Compared with other methods, the CNN-BiLSTM-AM method is more suitable for the prediction of stock price and for providing a reliable way for investors' to make stock investment decisions.

Keywords CNN · BiLSTM · AM · Stock price prediction

✉ Wenjie Lu
lulu@hebust.edu.cn

Jiazheng Li
69880353@qq.com

Jingyang Wang
jingyangw@hebust.edu.cn

Lele Qin
Mr_qin@163.com

1 School of Economics and Management, Hebei University of Science and Technology, Shijiazhuang 050018, China

2 Business School, Jiangsu Second Normal University, Nanjing 210000, China

3 School of Information Science and Engineering, Hebei University of Science and Technology, Shijiazhuang 050018, China

1 Introduction

The stock market is a place where stocks can be transferred, traded, and circulated. It has a history of 400 years and can be used as a channel for companies to raise funds [1]. By issuing stocks, a large amount of capital flows into the stock market. This promotes the concentration of capital, improves the organic composition of enterprise capital and greatly promotes the development of the commodity economy. Therefore, the stock market is regarded as a barometer of economic and financial activities in a country or region [2].

The Chinese stock market started later than the western stock market. The Chinese stock market was established in the early 1990s. Although the Chinese stock market started relatively late, the market scale and organizational structure of the Chinese stock market are comparable to those of western stock markets. With the rapid development of China's economy, the scale of the stock market has expanded rapidly, and more and more people have entered it to participate in stock investments [3, 4].

One of the issues that investors pay most attention to in the stock market is the changing trend of stock price [5]. Stock price is affected by many factors, such as the change of national policies, domestic and foreign economic environments, international situations, etc. [6, 7]. Stock price changes are often nonlinear. Predicting stock price changes in advance has always been an important issue for economists [8, 9]. Making a reasonable and accurate forecast for the change in stock price can greatly reduce the investment risk of investors. Such forecast allows investors to include the predicted stock price into their investment strategy and helps investors maximize their investment income.

In order to predict the stock price more accurately, this paper proposes a method based on CNN-BiLSTM-AM to predict the stock closing price of the next day. The model consists of convolutional neural networks (CNN), bi-directional long short-term memory (BiLSTM), and attention mechanism (AM). CNN can extract features from the input stock data. Long short-term memory (LSTM) is an improvement of recurrent neural network (RNN), which avoids the problem of gradient disappearance and gradient explosion caused by RNN. BiLSTM can fully find the interdependence of stock time series data. AM is a mechanism to obtain better results, which can capture the influence of the past characteristic states of time series data on stock price.

The main contributions of this paper are as follows:

(1) By analyzing the time sequence and correlation of stock price data, a new deep learning method CNN-BiLSTM-AM is proposed to predict the stock closing price of the next day.

(2) According to the influence level of the past characteristic states on the stock closing price of the next day, AM can be weighted to calculate the past characteristic states, so as to improve the accuracy of the prediction.

(3) By comparing with the other seven machine learning methods to predict stock price, it is proved that the CNN-BiLSTM-AM method is the most accurate and effective, which shows that it is more suitable for predicting stock price.

2 Related work

Traditionally, stock price prediction is based on simple mathematical models. Finance scholars initially used simple linear models to process stock data, such as simple autoregression model and simple moving average model. The unit root test is used to verify how much of time series is stationary. The non-stationary time series is transformed into different operations [10]. However, as stock data contains a large number of noise and uncertain factors, with the lengthening of the prediction period, the limitations of the linear model become clear [11]. Scholars at home and abroad have tried many methods to predict stock price, such as Vector Auto-Regressive model, Bayesian Vector Auto-Regressive model, Error Correction Model, and Kalman filter model.

Scientists then attempted to use nonlinear models for prediction, introduced machine learning methods such as neural networks, support vector machines (SVM) and successfully applied them to stock price time series prediction [5, 12–15]. In recent years, the application of machine learning methods in the field of stock prediction has gradually become a hot research direction for scholars [16–18]. In 1988, White used a neural network to predict IBM stock price, but the results were not good [19]. In 2003, Zhang predicted stock price using neural network and autoregressive integrated moving average (ARIMA), respectively. The experimental results showed that the neural network had obvious advantages in nonlinear data prediction, but the accuracy needed to be improved [20]. In 2007, Hammad et al. analyzed the stock price of the Jordan stock market using a multi-layer back propagation (BP) network, but did not mention that the traditional BP network is easy to fall into a local minimum [21]. In 2013, Wang et al. mixed the decision tree (DT) algorithm with SVM model. They first filtered most of the noise data using the DT algorithm, then processed the second stage training data using the SVM to predict the future price trend [22]. In 2015, Nayak et al. used an artificial chemical reaction optimization (ACRO) algorithm to train a multi-layer perception machine (MLP) to predict the stock market index [23]. In 2017, Wang proposed a stock price forecasting method based on a wavelet neural network [24]. In 2018, Hu Yue used CNN to predict stock price. The experimental results showed that CNN could predict time series, and in-depth learning was more suitable for solving time series problems [25]. In 2019, Zeng et al. used BiLSTM to predict the S&P 500 index. The results show that with the application of LSTM instead, the prediction results were more accurate than the existing prediction models [26].

3 CNN-BiLSTM-AM

3.1 CNN-BiLSTM-AM

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. BiLSTM has the characteristic of expanding according to the sequence of time, and it is widely used in time series analysis. AM has the importance

of adding the past characteristic states of time series data to the output results. It is more widely used to adjust the prediction results after BiLSTM. According to the characteristics of CNN, BiLSTM, and AM, a stock forecasting model based on CNN-BiLSTM-AM is established. The model structure diagram is shown in Fig. 1. The main structure is CNN, BiLSTM, and AM, including input layer, CNN layer (one-dimensional convolution layer, pooling layer), BiLSTM layer (forward LSTM layer, reverse LSTM layer), AM layer, and output layer.

3.2 CNN

CNN is a network model proposed by Lecun et al. in 1998 [27]. CNN is a kind of feed forward neural network, which has good performance in image processing and natural
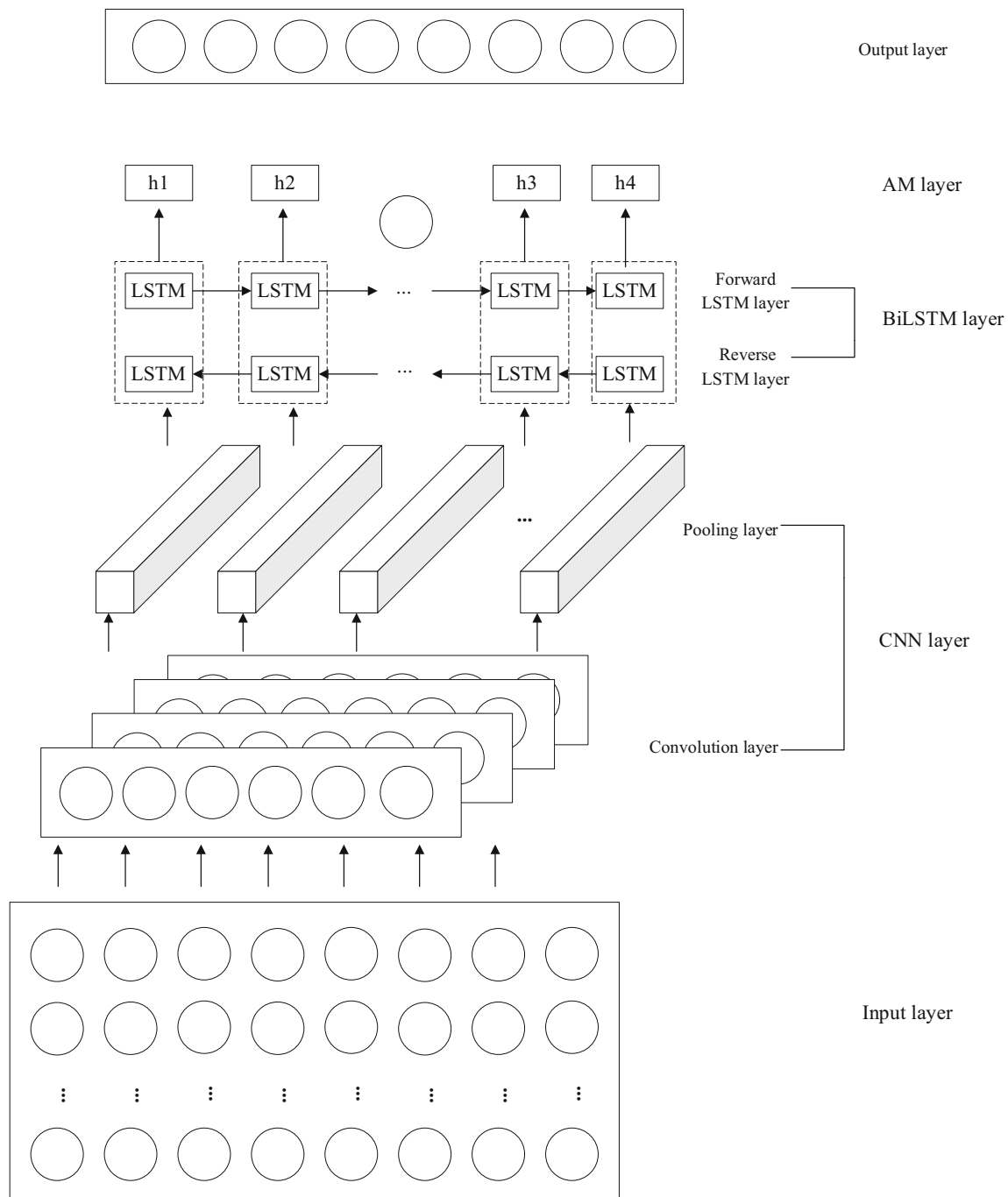

Fig. 1 CNN-BiLSTM-AM model structure diagram

language processing (NLP). It can be effectively applied to the prediction of time series [28]. The local perception and weight sharing of CNN can greatly reduce the number of parameters, thus improving the efficiency of learning models. CNN is mainly composed of three parts: convolution layer, pooling layer, and full connection layer [29]. Each convolution layer contains a plurality of convolution kernel, and its calculation is shown in formula (1). After the convolution operation of the convolution layer, the features of the data are extracted. However, 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 dimensions [30].

$$l_t = \tanh(x_t * k_t + b_t) \tag{1}$$

where l_t is the output value after convolution, tanh is the activation function, x_t is the input vector, k_t is the weight of the convolution kernel, and b_t is the bias of the convolution kernel.

3.3 LSTM

LSTM is a networking model proposed by Schmidhuber et al. in 1997 [31]. LSTM is a network model designed to solve the long-standing problems of gradient explosion and gradient disappearance in RNN [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 [33–35]. The LSTM memory cell consists of three parts: the forget gate, the input gate, and the output gate, as shown in Fig. 2.
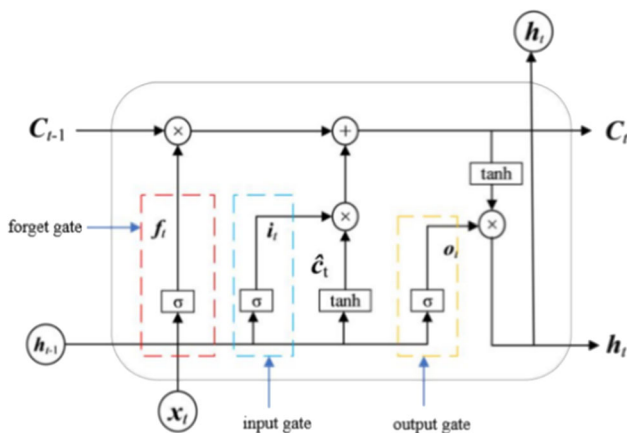

Fig. 2 Architecture of LSTM memory cell

C_{t-1} is the cell state of the previous moment, h_{t-1} is the final output value of the LSTM neuronal unit at the last moment, x_t is the input for the current moment, σ is the activation function, f_t is the output of the forget gate at the current moment, i_t is the input gate output for the current moment, $\widetilde{C_t}$ is the candidate cell status at the current moment, o_t is the output value of the output gate, C_t is the cell state at the current moment, h_t is the output of the current moment. The LSTM calculation process is as follows:

(1) The output value of the last moment and the input value of the current time are inputted into the forget gate. The output value of the forget gate is obtained after calculation as shown in formula (2):

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big) \tag{2}$$

where the value range of f_t is 0 to1, W_f is the weight of the forget gate, b_f is the bias of the forget gate, x_t is the input value of the current time, h_{t-1} is the output value of the last moment.

(2) The output value of the last time and the input value of the current time are inputted into the input gate. The output value and candidate cell state of the input gate are obtained after calculation. This is shown in formula (3) and formula (4):

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

$$\widetilde{C_t} = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

where the value range of i_t is 0 to 1, W_i is the weight of the input gate, b_i is the bias of the input gate, W_c is the weight of the candidate input gate, and b_c is the bias of the candidate input gate.

(3) The current cell state is updated as shown in formula (5):

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C_t} \tag{5}$$

where the value range of C_t is 0 to 1.

(4) The output value of the last moment and the input value of the current time are inputted into the output gate. The output value of the output gate is obtained after calculations shown in formula (6):

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{6}$$

where the value range of o_t is 0 to 1, W_o is the weight of the output gate, and b_o is the bias of the output gate.

(5) The output value of LSTM is obtained by calculating the output of the output gate and the state of the cell, as shown in formula (7):

$$h_t = o_t * \tanh(C_t) \qquad (7)$$

3.4 AM

AM was proposed by Treisman et al. in 1980 [36]. By calculating the probability distribution of attention, the key information is selected from a large number of information, the key input is highlighted, and the traditional model is optimized. The main idea of AM comes from the process of human visual attention. Human vision can quickly find the key areas and add attention focus to the key areas to obtain the required detailed information. Similarly, the AM selectively pays attention to some of the more important information, ignores the un-important information, and allocates the importance of the information.

As shown in Fig. 3, the calculation process of AM is generally divided into three stages:

(1) The similarity or correlation between Query (output feature) and Key (input feature) is calculated as shown in formula (8):

$$s_t = \tanh(W_h h_t + b_h) \qquad (8)$$

where W_h is the weight of AM, b_h is the bias of AM, h_t is the input vector, and W_h, and b_h are shared weights in each layer.

(2) The score of the first stage is normalized, and the softmax function is used to convert the attention score as shown in formula (9):

$$a_t = \frac{\exp(s_t^T v)}{\sum_t \exp(s_t^T v)}, \qquad (9)$$

where v is the attention value.

(3) According to the weight coefficient, the final attention value is obtained by weighted summation of value as shown in formula (10):

$$s = \sum_t a_t h_t \qquad (10)$$

3.5 CNN-BiLSTM-AM Training Process

The CNN-BiLSTM-AM training process is shown in Fig. 4:

The main steps are as follows:

1. Input Data: The data required for CNN-BiLSTM-AM training is inputted
2. Input Data Standardization: As there is a large gap in the input data, in order to better train the model, the z-score standardization method is adopted to standardize the input data as shown in formula (11):
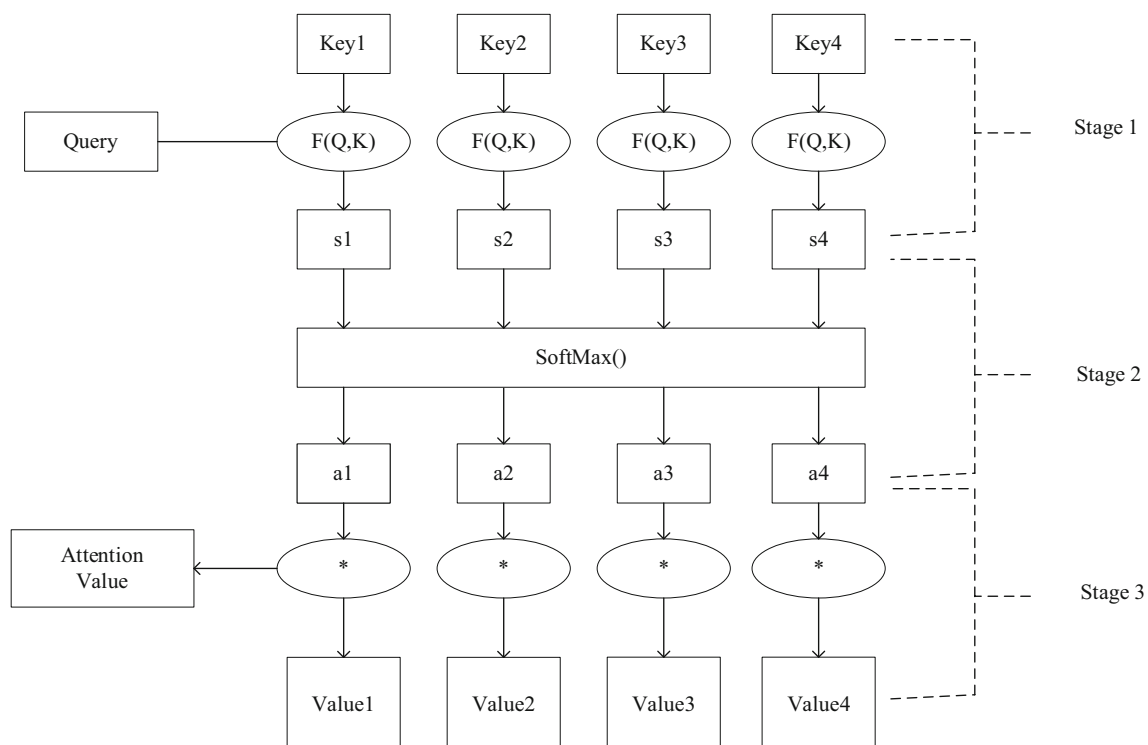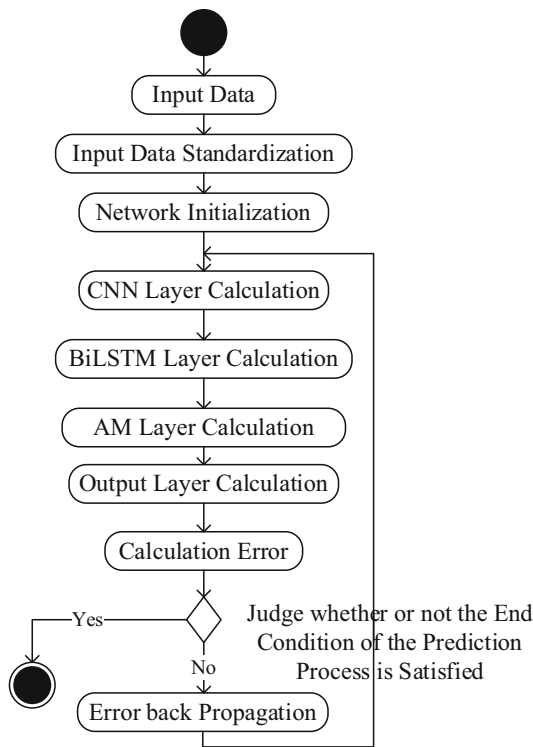

Fig. 3 AM process diagram

Fig. 4 Activity diagram of CNN-BiLSTM-AM training process

$$y_i = \frac{x_i - \bar{x}}{s} \tag{11}$$

where y_i is the standardized value, x_i is the input data, $\bar{x}$ is the average of the input data, and s is the standard deviation of the input data.

(3) Network Initialization: The weights and biases of each layer of the CNN-BiLSTM-AM are initialized

(4) CNN Layer Calculation: The input data is successively passed through the convolution layer and pooling layer within the CNN layer, the feature extraction of the input data is carried out, and the output value is obtained.

(5) BiLSTM Layer Calculation: The output data of the CNN layer is calculated through the hidden layer of the BiLSTM layer, and the output value is obtained.

(6) AM Layer Calculation: The output data of the BiLSTM layer is calculated through the AM layer, and the output value is obtained.

(7) Output Layer Calculation: The output value of the AM layer is calculated to obtain the output value of the model.

(8) 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 calculated.

(9) Judge whether or not the end condition of the prediction process is satisfied: The conditions for a

successful end are to complete a pre-determined number of cycles, the weight is lower than a certain threshold, and the error rate of the prediction is lower than a certain threshold. If at least one of the conditions for the end is met, the training will be completed. Otherwise, the training will continue.

(10) Error back Propagation: The calculated error is propagated in the opposite direction, the weight and bias of each layer is updated, and then the process goes back to step (4) to continue the network training.

3.6 CNN-BiLSTM-AM Prediction Process

The pre-condition for CNN-BiLSTM-AM prediction is that CNN-BiLSTM-AM has completed its training. The CNN-BiLSTM-AM prediction process is shown in Fig. 5.

The main steps are as follows:

(1) Input Data: The input data required for the prediction are inputted.

(2) Input Data Standardization: The input data are standardized according to the formula (11).

(3) Prediction: The standardized data are inputted into the trained CNN-BiLSTM-AM to get the corresponding output value.

(4) Data Standardization Restoration: The output value obtained through the CNN-BiLSTM-AM is the standardized value. The standardized value is restored to the original value using formula (12):

$$x_i = y_i * s + \bar{x} \tag{12}$$

where x_i is the standardized restored value, y_i is the output value of the CNN-BiLSTM-AM, s is the standard deviation of the input data, and $\bar{x}$ is the average value of the input data.

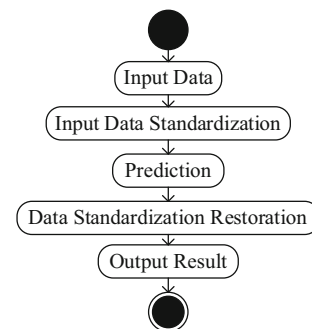(5) Output Result: The restored results are outputted to complete the prediction process.

Fig. 5 Activity diagram of CNN-BiLSTM-AM prediction process

Table 1 Partial sample data

Date	Opening price	Highest price	Lowest price	Closing price	Volume	Turnover	Ups and downs	Change
1991/7/1	136.64	138.62	136.56	136.85	2,294,000	12,469,884	− 0.71	− 0.5161
1991/7/2	135.91	135.96	135.69	135.96	283,800	3,794,100	− 0.89	− 0.6503
1991/7/3	135.28	135.96	134.98	135.27	271,500	1,818,504	− 0.69	− 0.5075
1991/7/4	136.63	136.63	134.19	136.63	1,339,400	8,095,138	1.36	1.0054
1991/7/5	136.01	137.68	135.9	135.96	1,454,000	9,394,861	− 0.67	− 0.4904

Table 2 Parameters' setting of CNN-BiLSTM-AM method

Parameters	Value
Convolution layer filters	64
Convolution layer kernel size	1
Convolution layer activation function	RELU
Convolution layer padding	Same
Pooling layer pool size	1
Pooling layer padding	Same
Pooling layer activation function	RELU
Number of hidden units in BiLSTM layer	64

4 Experiments

In order to prove the effectiveness of CNN-BiLSTM-AM, this method is compared with MLP, CNN, RNN, LSTM, BiLSTM, CNN-LSTM, CNN-BiLSTM, BiLSTM-AM, and CNN-BiLSTM using the same training set and test set data under the same operating environment. All methods are implemented in Python and Keras, an open-source learning library based on TensorFlow. All the experiments are carried out under the running environment of Intel i7-4700H 2.6 GHz, 12GBs of RAM, and Windows 10. In order to evaluate the prediction effect of CNN-BiLSTM-AM, the mean absolute error (MAE), root mean square error (RMSE), and R-square (R^2) are used as the evaluation criteria of the methods.

The MAE calculation formula is as follows:

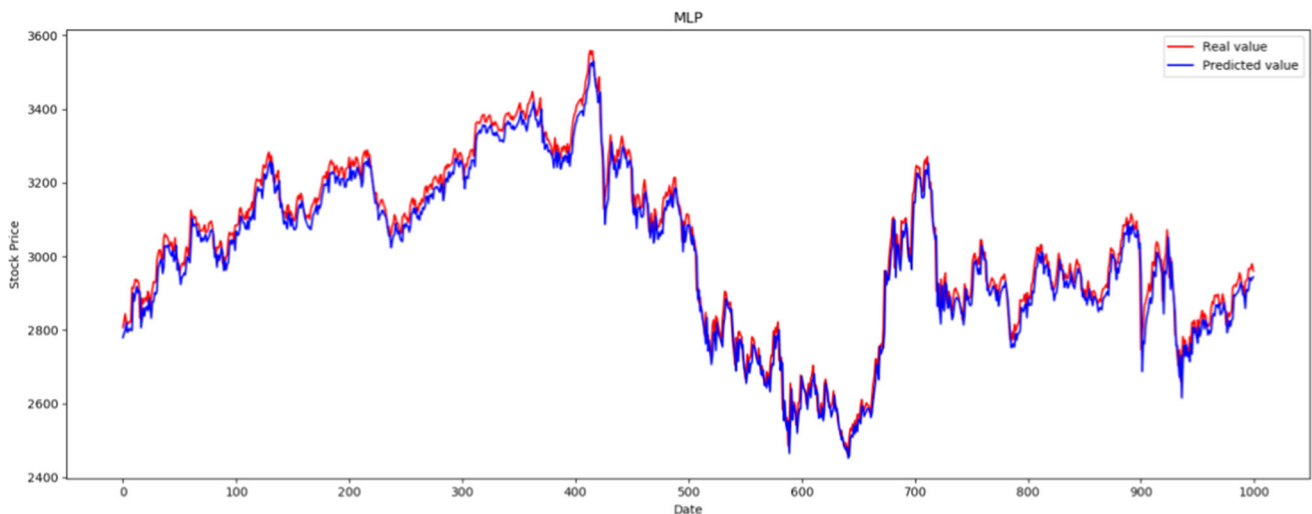$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y_i} - y_i| \tag{13}$$

where $\widehat{y_i}$ is the predicted value and y_i is the real value. The smaller the MAE is, the more accurate the prediction is.

The RMSE calculation formula is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{y_i} - y_i)^2} \tag{14}$$

where $\widehat{y_i}$ is the predicted value and y_i is the real value. The smaller the RMSE is, the more accurate the prediction is.

The R^2 calculation formula is as follows:

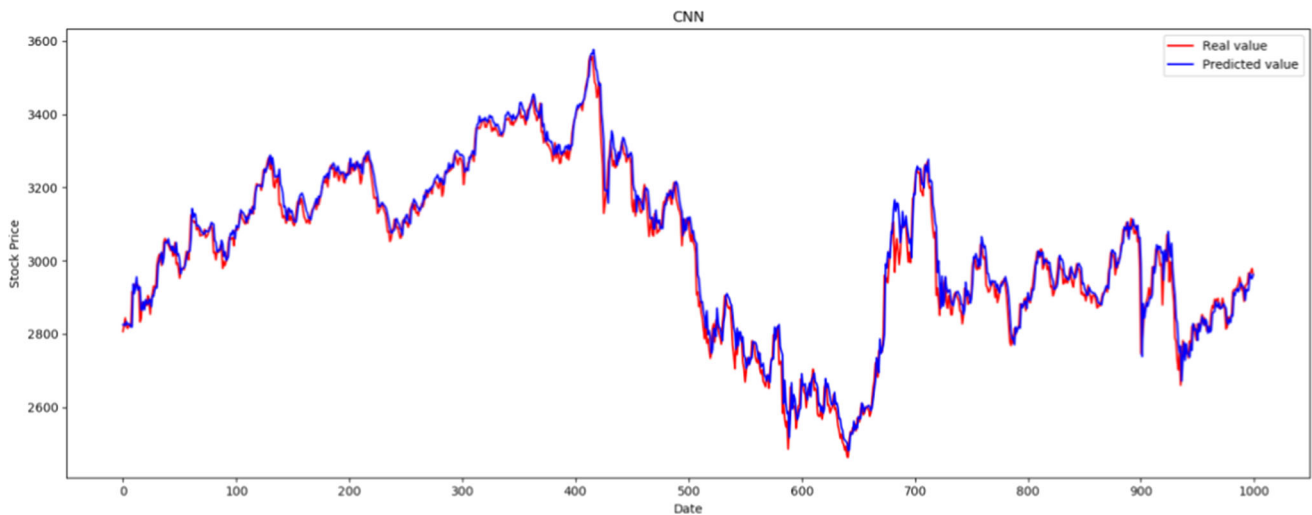**Fig. 6** Comparison of MLP predicted value and real value

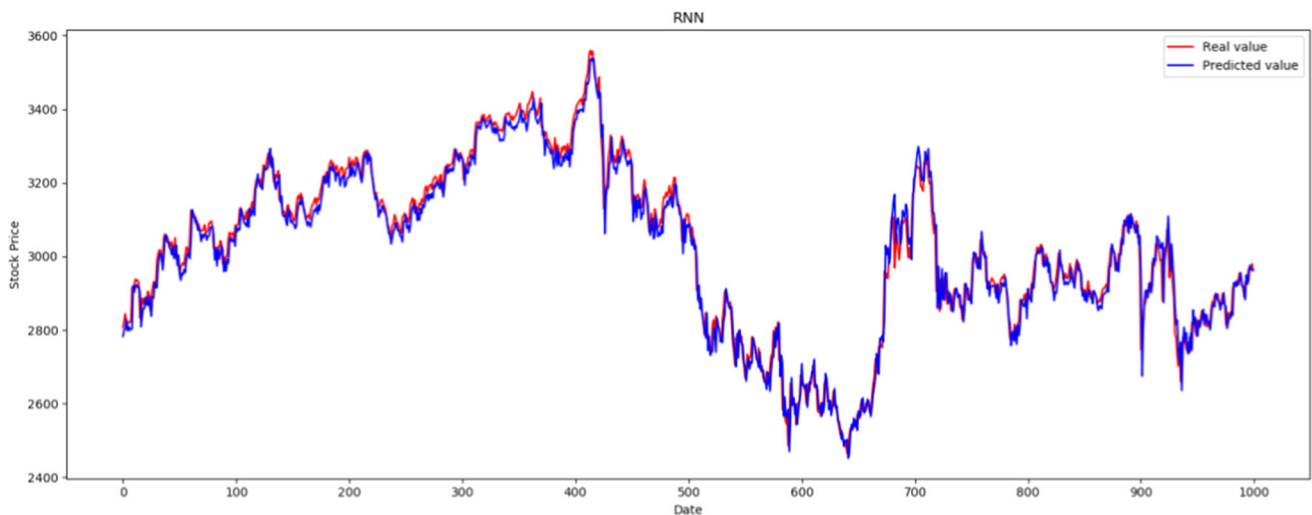**Fig. 7** Comparison of CNN predicted value and real value

Fig. 8 Comparison of RNN predicted value and real value

$$R^2 = 1 - \frac{\left(\sum_{i=1}^{n}(y_i - \widehat{y_i})^2\right)/n}{\left(\sum_{i=1}^{n}(\overline{y_i} - \widehat{y_i})^2\right)/n} \quad (15)$$

where $\widehat{y_i}$ is the predicted value, y_i is the real value, and $\overline{y_i}$ is the average value. The value range of R^2 is 0 to 1. The closer it is to 1, the better the performance.

4.1 Data

In this experiment, the Shanghai Composite Index (000,001) stock is selected as the experimental data. The daily trading data of 7083 trading days from July 1, 1991 to June 30, 2020 are obtained from the wind database. Each piece of data contains eight items: opening price, highest price, lowest price, closing price, volume, turnover,

ups and downs, and change. Some of the data are shown in Table 1. This experiment takes the data of the first 6083 trading days as the training set and the data of the last 1000 trading days as the test set.

Here opening price is the first transaction stock price for a trading day (after the opening of the stock exchange). Highest price refers to the highest price of a stock from the opening to the closing of each trading day. Lowest price refers to the lowest price of a certain stock from the opening to the closing of each trading day. Closing price refers to the weighted average trading volume price of each transaction one minute before the last trading of the stock on that day. Volume refers to the total number of stocks traded on the day. Turnover amount refers to the total amount of shares of all stocks traded that day. Ups and downs refer to the amount of change in price of a stock.
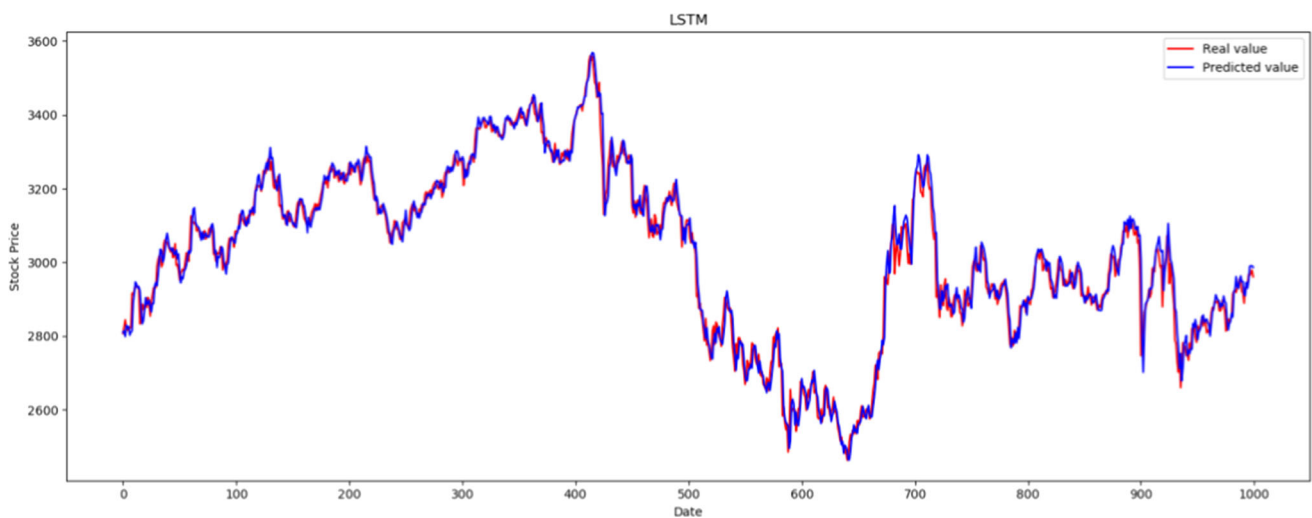
Fig. 9 Comparison of LSTM predicted value and real value
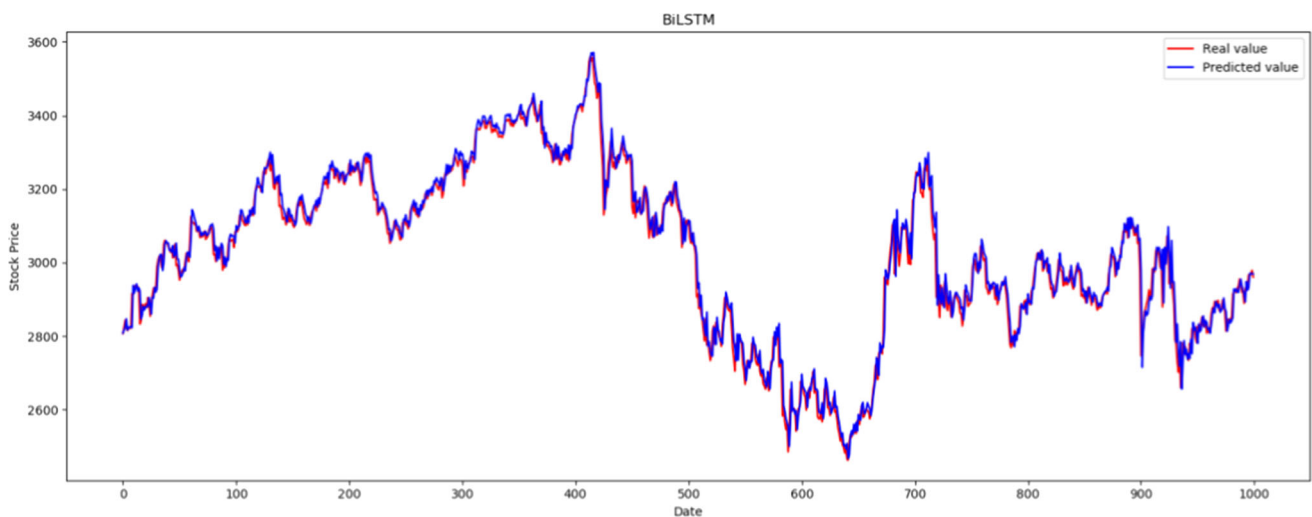
Fig. 10 Comparison of BiLSTM predicted value and real value

Change refers to the current trading day closing price compared with the previous trading day closing price value; this value is generally expressed as a percentage.

4.2 Model Implementation

Parameters' settings of the CNN-BiLSTM-AM model for this experiment are shown in Table 2.

In this experiment, all the method training parameters are the same, the epoch is 100, the loss function is MAE, the optimizer chooses Adam, batch size is 64, time step is 5, and learning rate is 0.001.

4.3 Results

The processed training set data is used to train MLP, CNN, RNN, LSTM, BiLSTM, CNN-LSTM, CNN-BiLSTM,

BiLSTM-AMand, and CNN-BiLSTM-AM, respectively. The model is achieved by training is used to predict the test set data, and the real value is compared with the predicted value as shown in Figs. 6, 7, 8, 9, 10, 11, 12, 13, and 14.

In Figs. 6, 7, 8, 9, 10, 11, 12, 13, and 14, among the eight prediction methods, the ranking of the broken line degree of fitting of real value to predicted value is CNN-BiLSTM-AM, BiLSTM-AM, CNN-BiLSTM, CNN-LSTM, BiLSTM, LSTM, CNN, ML from high to low. The broken line degree of fighting of real value to predicted value of CNN-BiLSTM-AM is the highest, almost completely coincident, while the broken line degree of fitting of MLP is the lowest.

According to the predicted value of each method and the real value, the evaluation error indexes of each method can be calculated, and the comparison results of the eight methods are shown in Table 3.
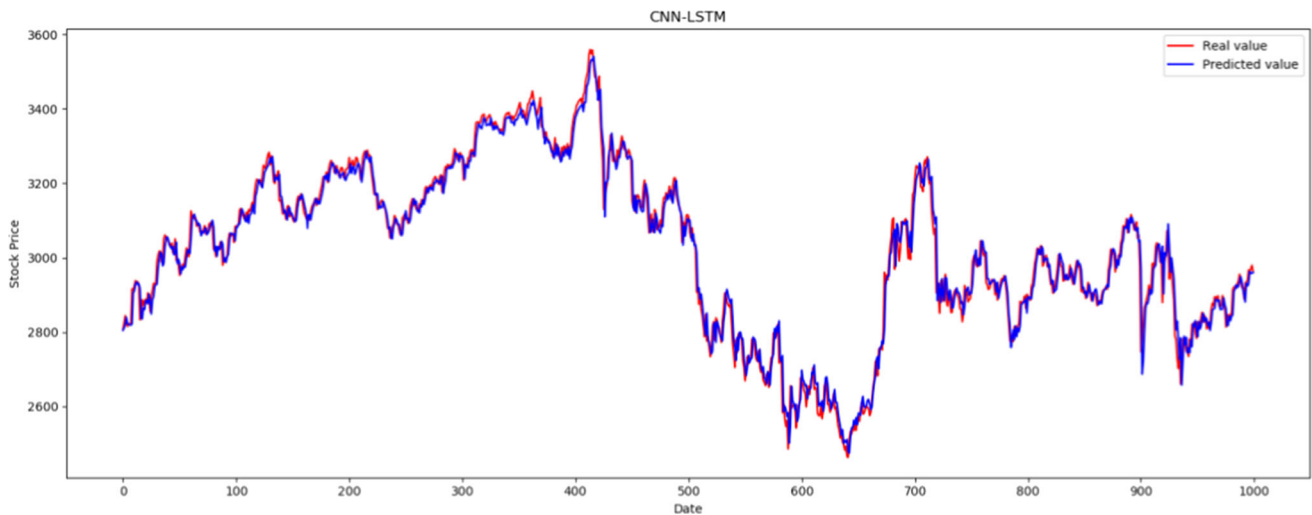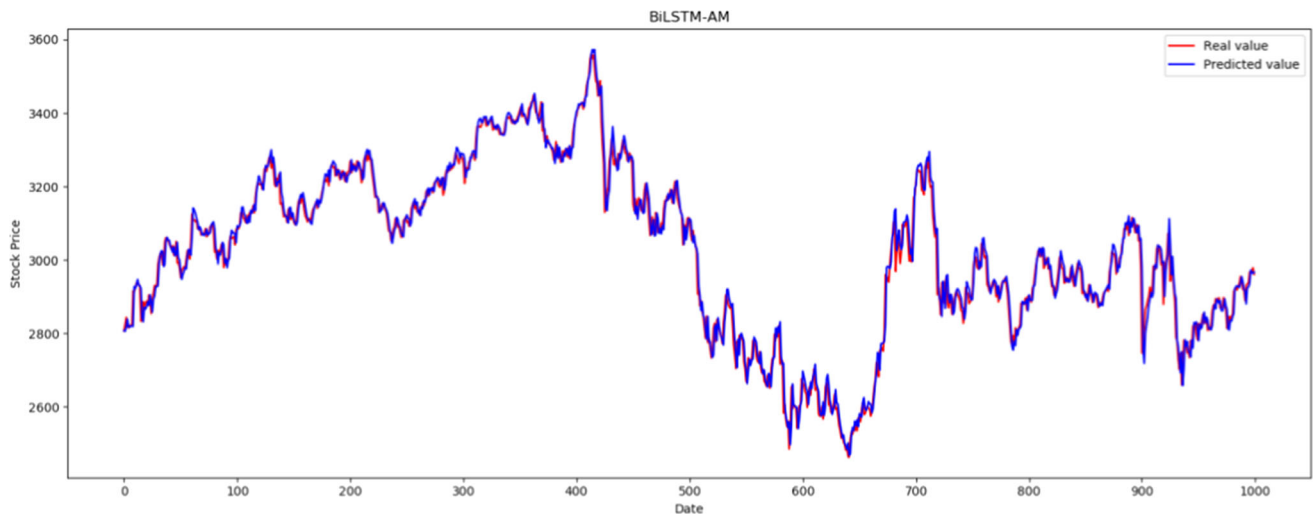
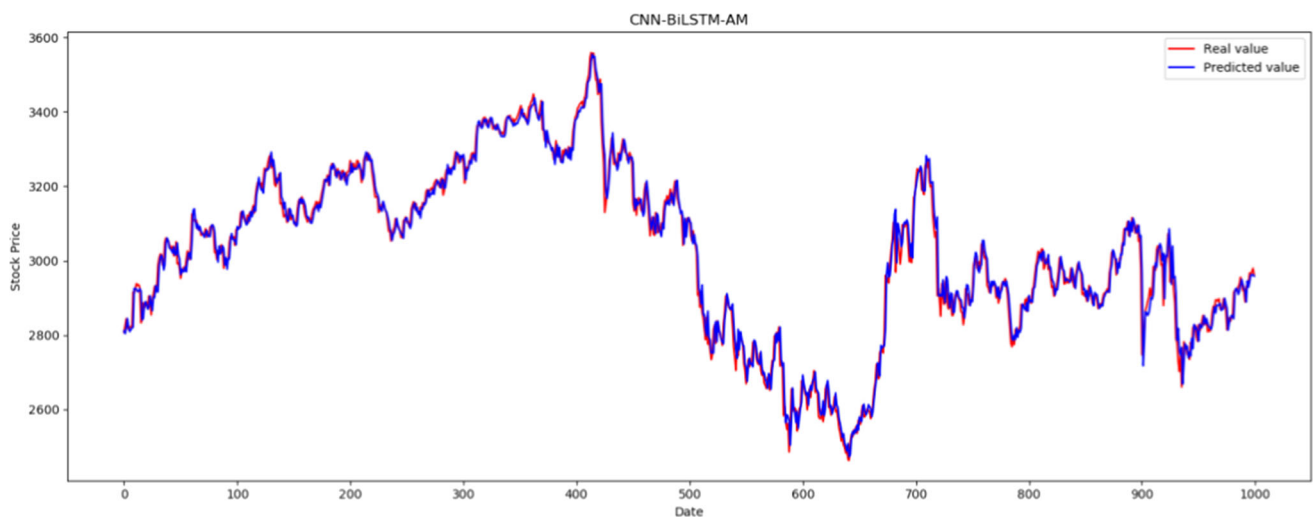**Fig. 11** Comparison of CNN-LSTM predicted value and real value
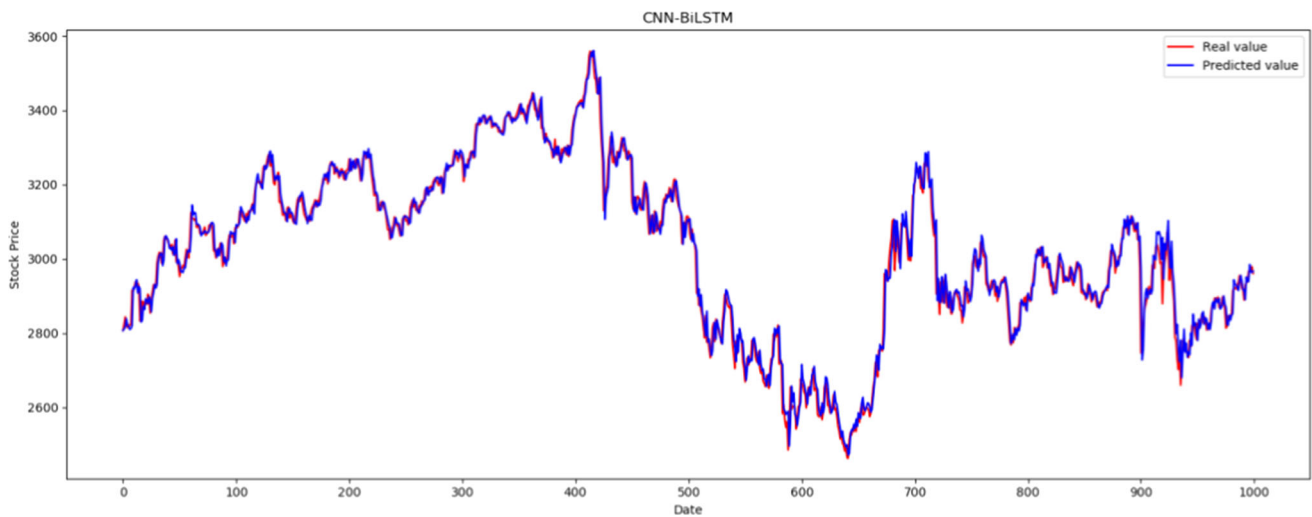
Fig. 12 Comparison of CNN-BiLSTM predicted value and real value

From Table 3, the MAE and RMSE of MLP are the largest and R^2 is the smallest. On the other hand, the MAE and RMSE of CNN-BiLSTM-AM is the smallest, R^2 is the largest, and the closest is 1. The prediction performance of the eight methods from high to low is CNN-BiLSTM-AM, BiLSTM-AM, CNN-BiLSTM, CNN-LSTM, BiLSTM, LSTM, CNN, RNN, and MLP. Comparing LSTM with RNN, its MAE, RMSE is less, while R^2 is more. Its MAE (26.822 compared to 24.361) is less by 9.2%. Its RMSE (35.801 compared to 34.331) is less by 4.1%. Its R^2 is more by 0.2%; thus, LSTM is superior to RNN. Compared with LSTM, BiLSTM reduces MAE from 24.361 to 23.409, RMSE from 34.331 to 33.579, and increases R^2 from 0.9770 to 0.9780, indicating that BiLSTM has a certain improvement in prediction accuracy compared with LSTM.

For BiLSTM, after CNN layer, MAE and RMSE have a certain reduction, R^2 has a certain increment. MAE decreases from 23.409 to 22.715, and RMSE decreases from 33.579 to 32.065. R^2 increases to 0.9800. When CNN-BiLSTM is introduced into AM, its predictive accuracy improves. MAE decreases by 0.763, RMSE decreases by 0.371, and R^2 increases by 0.0004. The results show that among the eight methods, the performance of CNN-BiLSTM-AM is the best. Its MAE is 21.952, the RMSE is 31.694, and R^2 is 0.9804. Therefore, out of the eight methods, the CNN-BiLSTM-AM method proposed in this paper can best predict the stock closing price of the next day and provides a reference for investors to make the right investment decisions.

Fig. 13 Comparison of BiLSTM-AM predicted value and real value

Fig. 14 Comparison of CNN-BiLSTM-AM predicted value and real value

Table 3 Comparison of evaluation error indexes of the five methods

Method	MAE	RMSE	R^2
MLP	31.496	39.260	0.9699
CNN	25.665	36.878	0.9735
RNN	26.822	35.801	0.9751
LSTM	24.361	34.331	0.9770
BiLSTM	23.409	33.579	0.9780
CNN-LSTM	23.195	32.640	0.9792
CNN- BiLSTM	22.715	32.065	0.9800
BiLSTM-AM	22.337	31.955	0.9801
CNN-BiLSTM-AM	**21.952**	**31.694**	**0.9804**

5 Conclusions

According to the chronological characteristics of stock price data, this paper proposes a CNN-BiLSTM-AM method 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. Thus, it makes full use of the time sequence characteristics of the stock data. CNN is used to extract the features of the input data. BiLSTM is used to learn and predict the extracted feature data. AM can be used to capture the influence of the feature states of the time series data at different times on the prediction results. This is done to improve the prediction accuracy of the method. The experimental results show that the CNN-BiLSTM-AM has the highest prediction accuracy

and the best performance compared to MLP, CNN, RNN, LSTM, BiLSTM, CNN-LSTM, CNN-BiLSTM, and BiLSTM-AM. MAE and RMSE of CNN-BiLSTM-AM are the smallest of all methods, and R^2 is the closest to 1. It is difficult to achieve high prediction accuracy by using only a single network, and complicating the network can improve its prediction accuracy. CNN-BiLSTM-AM is suitable for the prediction of stock price and can provide a relevant reference for investors to maximize investment returns. The proposal of CNN-BiLSTM-AM also provides practical experience for peoples' research on financial time series data.

Future research work will mainly adjust the parameters in the model to make the results more accurate. Future research work will also study whether or not the model can be applied to more application fields of time series prediction, such as gold price prediction, oil price prediction, weather prediction, earthquake prediction and so on.

Compliance with ethical standards

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Jin Z, Yang Y, Liu Y (2020) Stock closing price prediction based on sentiment analysis and LSTM. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04504-2
2. Badea L, Ionescu V, Guzun A (2019) What is the causal relationship between stoxx europe 600 sectors? but between large firms and small firms. Econ Comput Econ Cybern Stud Res 53(3):5–20
3. Algheriani NMS, Majstorovic VD, Kirin S, Spasojevic Brkic V (2019) Risk Model for Integrated Management System. Tehnicki vjesnik Tech Gaz 26(6):1833–1840
4. Liu Y, Hu Z, Li H, Zhu H (2019) Does preemption lead to more leveled resource usage in projects? a computational study based on mixed-integer linear programming. Econ Comput Econ Cybern Stud Res 53(4):243–258
5. Li J, Pan S, Huang L, Zhu X (2019) A machine learning based method for customer behavior prediction. Tehnicki vjesnik Tech Gaz 26(6):1670–1676
6. Moon K, Kim H (2019) Performance of deep learning in prediction of stock market volatility. Econ Comput Econ Cybern Stud Res 53(2):77–92
7. Kohara K, Fukuhara Y, Nakamur Y (1996) Selective presentation learning for neural network forecasting of stock markets. Neural Comput Appl. https://doi.org/10.1007/BF01414874
8. Xiao C, Xia W, Jiang J (2020) Stock price forecast based on combined model of ARI-MA-LS-SVM. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04698-5
9. Yu P, Yan X (2020) Stock price prediction based on deep neural networks. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04212-x
10. Dunea D, Iordache S (2015) Time series analysis of air pollutants recorded from romanian emep stations at mountain sites. Environ Eng Manag J 14(11):2725–2735
11. Wu Z, Fan J, Gao Y, Shang H, Song H (2019) Study on prediction model of space-time distribution of air pollutants based on artificial neural network. Environ Eng Manag J 18(7):1575–1590
12. Huang W, Wang H, Zhao H, Wei Y (2019) Temporal-spatial characteristics and key influencing factors of PM2.5 concentrations in China based on Stirpat model and Kuznets curve. Environ Eng Manag J 18(12):2587–2604
13. Kueh S, Kuok K (2018) Forecasting long term precipitation using cuckoo search optimization neural network models. Environ Eng Manag J 17(6):1283–1292
14. Feng Z, Zhang Z, Zhang Q, Gongab D (2018) Evaluation of soil suitability for cultivation based on back-propagation artificial neural network: The case of Jiangxia district. Environ Eng Manag J 17(1):229–236
15. Vrecko I, Kovac J, Rupnik B, Gajsek B (2019) Using queuing simulation model in production process innovations. Int J Simul Model 18(1):47–58
16. Kim K, Lee W (2004) Stock market prediction using artificial neural networks with optimal feature transformation. Neural Comput Appl. https://doi.org/10.1007/s00521-004-0428-x
17. Xu Q, Wang L, Jiang C, Liu Y (2020) A novel (U)MIDAS-SVR model with multi-source market sentiment for forecasting stock returns. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04063-6
18. Chung H, Shin K (2020) Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04236-3
19. White H (1988) Economic perdication using neural networks: the case of ibm daily stock returns. Earth Surf Proc Land 8(5):409–422
20. Zhang G (2003) Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing 50:159–175
21. Hammad AAA, Ali SMA, Hall EL (2007) Forecasting the Jordanian stock price using artificial neural network. In: Intelligent engineering systems through artificial neural networks, vol 17. Digital Collection of The American Society of Mechanical Engineers, pp 1–6. https://doi.org/10.1115/1.802655.paper42
22. Wang D, Liu X, Wang D (2013) A DT-SVM strategy for stock futures prediction with big data. Computational science and engineering. Sydney, NSW, Australia IEEE 2013:1005–1012
23. Nayak S, Misra B, Behera H (2015) Artificial chemical reaction optimization of neural networks for efficient prediction of stock market indices. Ain Shams Eng J. https://doi.org/10.1016/j.asej.2015.07.015
24. Wang P, Lou Y, Lei L (2017) Research on Stock Price Prediction Based on BP Wavelet Neural Network with Mexico Hat Wavelet Basis. In: Proceedings of the 2017 International Conference on Education, Economics and Management Research (ICEEMR 2017). https://doi.org/10.2991/iceemr-17.2017.25
25. Hu Y (2018) Stock market timing model based on convolutional neural network—Taking Shanghai Composite Index as an example. Finance Econ 2018(04):71–74
26. Zeng A, Nie W (2019) Stock Recommendation System Based on Deep Bidirectional LSTM. Comput Sci 46(10):84–89

27. Lecun Y, Botou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
28. Qin L, Yu N, Zhao D (2018) Applying the convolutional neural network deep learning technology to behavioural recognition in intelligent video. Tehnicki vjesnik Tech Gaz 25(2):528–535
29. Hao Y, Gao Q (2020) Predicting the trend of stock market index using the hybrid neural network based on multiple time scale feature learning. Appl Sci 10(11):3961–3974
30. Kamalov F (2020) Forecasting significant stock price changes using neural networks. Neural Comput Appl. https://doi.org/10.1007/s00521-020-04942-3
31. Hochreiter S, Schmidhuber J (1997) Long Short-Term Memory. MIT Press 9(8):1735-1780
32. Ta V, Liu C, Tadesse D (2020) Portfolio optimization-based stock prediction using long-short term memory network in quantitative trading. Appl Sci 10(2):437–456
33. Borovkova S, Tsiamas I (2019) An ensemble of LSTM neural networks for high-frequency stock market classification. J Forecast 38(6):600–619
34. Livieris I, Pintelas E, Pintelas A (2020) A CNN–LSTM model for gold price time-series forecasting. Neural Comput Appl. https://doi.org/10.1007/s00521-020-04867-x
35. Yan X, Weihan W, Chang M (2020) Research on financial assets transaction prediction model based on LSTM neural network. Neural Comput Appl. https://doi.org/10.1007/s00521-020-04992-7
36. Treisman A, Gelade G (1980) A feature-integration theory of attention. Cogn Psychol 12(1):97–146