ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

Analysis of Forecasting Stock Prices Using CNN Model

Linlu Mao

North China University of Water Resources and Electric Power, Zhengzhou, 450046, China

Abstract: *Creating a trading strategy and selecting the ideal time to purchase or sell stocks depends in large part on stock price expectations. This paper provides a CNN-based stock price time series forecasting method, which proves the optimality of the model by comparing the accuracy of different models, which provides a possible direction for the exploration of stock price forecasting. This paper first introduces the working principle of CNN, LSTM, and Conv1D, and then experiments are carried out by establishing a model, and finally the relevant conclusions are obtained. The experimental results show that the Trainscore RMSE, Train MAE, Testscore RMSE, Test MAE, and MAE of CNN has a smaller size. Thus, in comparison to the LSTM and Conv1D-LSTM, CNN is the model with the best efficiency and greatest accuracy in forecasting, which is more suitable for investors to predict future stock prices than LSTM and Conv1D-LSTM.*

Keywords: *CNN, LSTM, Stock price prediction*

1. Introduction

Stock price prediction has become a research hotspot over these days. There are many factors influencing stock prices, macro and microeconomic policies, corporate behavior, investor confidence, market conditions, emergencies and so on, so, accurate prediction of stock prices has become an urgent need for many investors. Many researchers try to take different methods of analysis and forecasting. In the recent past, computational intelligence in finance has been an extremely popular subject in both academia and the financial industry[1]. A deep learning-based forecasting model is essential in light of the amount of uncertainty and unpredictability that exist in the stock market, which incorporates investors' personality traits is put forward to address these difficulties.[2].

Many recent researches have made a push to develop functional computerized trading mechanisms by using machine learning methods to estimate stock prices and portfolio management. The return of an investment can be increased by being able to anticipate future patterns in a stock's progress, which can be maximized. There are many models used to estimate the stock price but all fail[3].

There has been considerable debates over the predictability of the value of stocks. The Efficient Market Hypothesis says that considering historical data on stock prices is unpredictable, neither fundamental nor technical research can be used to identify inexpensive stocks or forecast market patterns, where both institutional and individual investors continued to use these as important tools. More particular, theory and evidence imply that those with an advantage in comprehending data are better at making investments in the stock market. [4].

This paper provides a CNN-based stock price time series forecasting method, which proves the optimality of the model by comparing the accuracy of different models, and we divide this article into four parts, namely literature review, models and methods, experimental analysis and conclusions.

2. Literature Review

In the scholarly landscape of predictive models using advanced machine learning approaches, multiple propositions have emerged. Jing N. and colleagues, in 2021, proffered a hybrid study framework that employs a Long-Short Term Memory (LSTM) neural network methodology. [4]. Furthermore, A. Moghar and his team aimed toward creating a model installing recurrent neural networks (RNN), particularly LSTM, to forecast future valuations of stocks [5].

Equally worth noting is the work of Yi SUN and colleagues, who proposed a freshly developed hybrid

ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

stock model for forecasting, namely ISI-CNN-LSTM. This model considers investor mindsets by combining LSTM and Convolutional Neural Network (CNN). They greatly boosted the model's capacity to predict outcomes through utilizing LSTM to extract geographical qualities and CNN to discover broad elements from the data, given that the network structure incorporates investor emotions [6].

Another crucial contribution to the field has been made by Keren He and her team, who amalgamated the merits of CNN and LSTM to enhance the reasonable projections of stocks. They built a function removal layer using the supervised CNN approach, and then fed the retrieved features into an LSTM to better understand the seasonal properties of the signals. Empirical data demonstrated the superiority of their proposed model over conventional CNN and LSTM, significantly reducing forecasting errors[7].

Looking back to 2017, B. Zhao and colleagues advanced a brand-new Convolutional Neural Network (CNN) structure specifically for recognizing time data. Contrasting with more feature-based categorization methods, CNN was able to discover and extract the appropriate internal structure, generating deep features of the input time series via convolution and pooling operations[8]. Tong Wang and his team implemented two LSTM and CNN models are used to predict MRF stock values, with comparative analysis indicating a superior prediction effect of the CNN model[9].Furthermore, O.B. Sezer and colleagues proposed a 2-D convolutional neuronal network-based CNN-TA trading algorithms model with image processing capabilitiess[10].

In the present study, we compare and contrast the predictive accuracies of the CNN model, LSTM model, and Conv1D-LSTM model in forecasting stock prices. Evaluation indicators such as mean absolute error (MAE) and root mean square error (RMSE) of the CNN model are used to ascertain a model that provides higher prediction accuracy and is most appropriate for predicting stock prices.

3. Models and Methods

3.1 CNN

CNN, as one of the sign algorithms of deep learning, is widely used in the field of image recognition. The CNN composition is as follows:

(1) Input layer: The input characteristics are standardized.

(2) Convolutional layer: Adjust the depth, stride, and zero-padding such that the kernel that performs convolution is able to determine and store the characteristic values for each region in the input image, making it easier to extract features[9].

(4) Pooling layer: Compress input data to lower the computational burden, lower overfitting, which and improve feature extraction.

(5) Fully connected layer: A final CNN hidden layer contains the layer with complete connectivity, which exclusively sends commands to other layers that are completely integrated, and the main function of which is classification.

3.2 LSTM[9]

LSTM is designed to get around the issue with fading gradients in RNN by learning the time series data over an extended period interaction by using memory cells and gates. Some researchers have proven that LSTM is a reliable stock forecasting scheme, to forecast future patterns in the value of shares. LSTM encounters three components: the input gate, output gate, and forget gate, which are as follows:

(1) The forget gate determines which information continues to exist and which has been eliminated based on the results of the previous step and the input from the now-completed step. The calculation formula is:

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

Where f_t denotes that the output is equal to or greater than 1, 0 denotes that this data has been dropped, and 1 denotes that the data continues to exist., h_{t-1} denotes the output of the preceding procedure, x_t stands for the information used in the procedure at hand, W_f stands for the significance of forgetting gate, and b_f for the forgetting gate's departure.

(2) Select the new data you intend to maintain. The Sigmoid layer, also known as an input gate, is the

ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

initial layer and is used to decide what fresh input data can be left behind. The tanh layer is the following and contains an entirely novel candidate value vector generation. The calculation formula is:

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

$$\tilde{C}_t = \sigma(W_C * [h_{t-1}, x_t] + b_C) \tag{3}$$

Where i_t is a feedback signal with a between zero and one alter, W_i is the amount of weight of the input gate and b_i is the difference between the two values of the input gate, W_C is the magnitude of the one being considered input gate and b_C is the departure of the candidate input gate.

(3) Restore. The calculation formula is:

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

Where the $f_t * C_{t-1}$ determines the data that is dismissed, and $i_t * \tilde{C}_t$ is the updated state's newly advised amount, which has changed.

(4) Output value.

Which form of cell is exported is chosen by the sigmoid layer, after which it will be given consideration by the tanh procedure layer and transmitted. The formula is shown below:

$$o_t = \sigma(W_o * [h_{t-1}, x_i] + b_o) \tag{5}$$

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

Where Wo denotes the output gate's weight and bo denotes the output gate's departure [9].

3.3 Conv1D-LSTM

There are many predicting techniques for time series, such as ARIMA, etc., and with the advent of deep learning, many people have started using Conv1D-LSTM for time series forecasting.

The Conv1D layer makes the input time series smoother, so we can reduce the hassle of adding rolling average or rolling standard deviation values to the input features. At the same time, the unique processing mechanism of LSTM for learning the time series data's continuously connection by using memory cells and gates adds great advantages to time series foreseeing, where it may be challenging to apply linear approaches to multimodal or multi-input estimating issues.

4. Experiment

We research with LSTM and Conv1D-LSTM utilizing both the training and test set data in the equal operating setting to demonstrate the efficacy of the CNN, and the relevant variables include open, high, low and close prices.

Table 1: A subset data set

trade_date	open	high	low	close
2012-12-31	15.7100	16.1700	15.6500	16.0200
2013-01-04	16.3200	16.4500	15.9200	15.9900
2013-01-07	15.9800	16.3500	15.8800	16.3000
2013-01-08	16.3000	16.3700	15.8600	16.0000
2013-01-09	15.9600	16.0200	15.8000	15.8600

4.1 Data description

The experimental data used here is Ping An of China (000001. SZ) stock data for 1304 trading days from December 31, 2012 to May 18, 2018, obtained using the Tushare database interface, which contains the open, close, high and low prices for each trading day. Table 1 shows some of the data. As the training set, we use the data from the initial 1038 trading days, and as the set for testing, we employ the data from the most recent 260 trading days.

ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

4.2 The evaluation index system established

Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are two measurement metrics we choose to establish an evaluation index system to evaluate the model, for the purpose to assess the predictive power of CNN.

1) Mean Absolute Error(MAE)

The overall average of every one of the sightings' departures from the mathematical mean is known as the mean absolute error, which accurately reflects the magnitude of the actual prediction error. The decreased the MAE value, the better the prediction is. And the calculation formula is as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}| y_i - f(x_i)| \tag{7}$$

Where $f(x_i)$ is the predicted value， and y_i is the true value.

2) Root Mean Squared Error(RMSE)

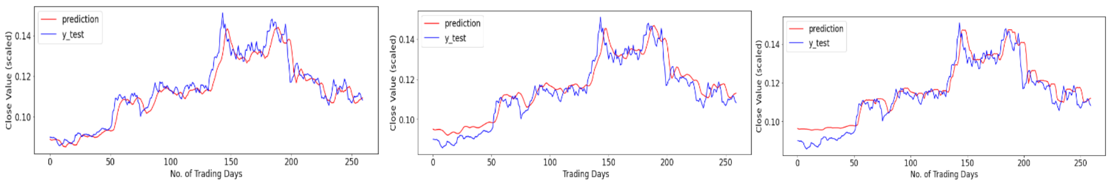
The inverse of the root of the diagonal of the observed value's divergence from its true value to the observational ratio, n, is known as the root mean squared error. The calculation formula is as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2} \tag{8}$$

Where $f(x_i)$ is the predicted value， y_i is the true value. The shorter amount of RMSE, the better the prediction is. The more adjacent the MAE and RMSE figures are to 0, the less substantial the inaccuracy between the predicted value and the actual value is, and the higher the prediction accuracy.

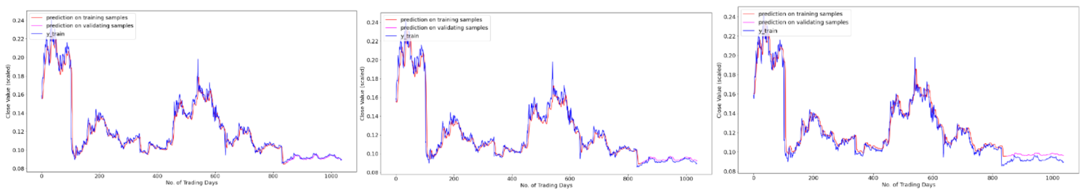
4.3 Model implementation

Train the CNN, LSTM, Conv1D-LSTM, using the practice collection's data, and the correlation function image is shown in Figure 1 and Figure 2.



(a) CNN *(b)LSTM* *(c)Conv1D-LSTM*

Figure 1: Comparison of prediction and y-test of CNN, LSTM and Conv1D-LSTM



(a) CNN *(b)LSTM* *(c)Conv1D-LSTM*

Figure 2: Comparison of prediction on validating samples and y-test of CNN, LSTM and Conv1D-LSTM

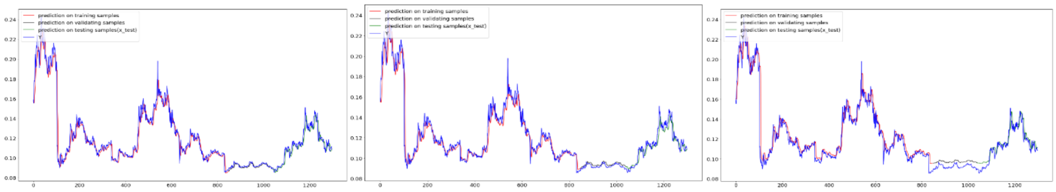
Figure 1 shows the comparison of prediction and y-test of CNN, LSTM, and Conv1D-LSTM, which can be seen that there is a certain gap between the prediction and y-test of three models, and the model performance is not good enough. But in comparison to LSTM and Conv1D-LSTM, the curve of CNN has a smaller visual gap and a closer trend.

Figure 2 shows the comparison of Prediction on validating samples and y-test of CNN, LSTM and Conv1D-LSTM, which can be seen that the fit of Prediction on validating samples and y-test of CNN and LSTM are high, and the Conv1D-LSTM model is relatively poor.

ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

5. Results

Train CNN, LSTM and Conv1D-LSTM with processed training set data, and test set data predictions with trained CNN, Conv1D-LSTM and LSTM, comparing the actual values of each model with the predicted values, are as shown in Figure 3.



(a) CNN *(b)LSTM* *(c)Conv1D-LSTM*

Figure 3: Comparison of the predicted values and actual values of CNN, LSTM and Conv1D-LSTM

Figure 3 shows the comparison of the predicted and actual values of the CNN, LSTM and Conv1D-LSTM. It is found that the predicted values and actual values of the two curves of CNN almost completely coincide, where the estimated cost precedes the price that was paid well, and the dashed line fitting degree of the LSTM and Conv1D-LSTM is lower than that of CNN, indicating the performance is inferior to CNN.

The evaluation indicators of different models are calculated and compared, and the comparison results are shown in Table 2.

Table 2: Evaluation indicators

	Trainscore RMSE	Train MAE	Testscore RMSE	Test MAE	*MAE*
CNN	0.0077	0.0041	0.0049	0.0036	0.3578
LSTM	0.0082	0.0052	0.0053	0.0041	0.4101
Conv1D-LSTM	0.0089	0.0054	0.0058	0.0044	0.4416

As can be seen from Table 2, the Trainscore RMSE, Train MAE, Testscore RMSE, Test MAE, and MAE of CNN are the smallest. These findings demonstrate that the CNN model is the most effective one in three models. CNN provides the most rigidity and is better meant for investors to predict future stock prices than LSTM and Conv1D-LSTM.

6. Conclusions

This article uses Ping An (000001. SZ) stock data for 1304 trading days from December 31, 2012 to May 18, 2018, including the open, close, high and low prices of each trading day, and divided them into training sets and test sets, and the CNN was experimented with the same data from the test set and the instructional set to prove the effectiveness of the CNN, where LSTM and Conv1D-LSTM are in the same operating environment. The experimental results show that the Trainscore RMSE, Train MAE, Testscore RMSE, Test MAE, and MAE of CNN are the lowest. Thus, CNN offers the highest level of forecast accuracy, which is the best performing model in contrast to LSTM and Conv1D-LSTM, and is more suitable for investors to predict future stock prices than LSTM and Conv1D-LSTM.

The model still has significant drawbacks, though. For instance, just the effect of previous stock price data on stock prices is taken into account, and it is not considered in combination with comprehensive factors such as investor sentiment, national policies, and international environment. Therefore, in the future, it is hoped to improve the study of sentiment connected to equity incorporate national policies, international environment and other factors into the model as new variables, and further compare and improve various model methods in order to obtain more accurate stock price prediction results.

Acknowledgements

Thank you to all the people who supported and helped during the study and the pioneers in the field of stock price forecasting with deep learning.

ISSN 2616-5775 Vol. 6, Issue 10: 79-84, DOI: 10.25236/AJCIS.2023.061012

References

[1] Ozbayoglu A. M., Gudelek M. U., Sezer O. B., " Deep learning for financial applications: A survey, " Applied Soft Computing, 93 (2020), Article 106384

[2] Jin Z., Yang Y. & Liu Y. Stock closing price prediction based on sentiment analysis and LSTM. Neural Comput & Applic 32, 9713–9729 (2020). https://doi. org/10. 1007/s00521-019-04504-2

[3] Jaimin Shah, Darsh Vaidya, Manan Shah," A comprehensive review on multiple hybrid deep learning approaches for stock prediction," Intel. Syst. Appl., 16 (2022), pp. 1-14, 10. 1016/j. iswa. 2022. 200111

[4] Jing N., Wu Z., Wang H. "A hybrid model integrating deep learning with investor sentiment analysis for stock price prediction, " Expert Syst. Appl., 178 (2021), Article 115019

[5] A. Moghar, M. Hamiche," Stock market prediction using LSTM recurrent neural network," Procedia Comput. Sci., 170 (2020), pp. 1168-1173, 10. 1016/j. procs. 2020. 03. 049

[6] Yi Sun, Qingsong Sun, Shan Zhu. Prediction of Shanghai Stock Index Based on Investor Sentiment and CNN-LSTM Model. Journal of Systems Science and Information. 2022, 10(6): 620-632 https://doi. org/10. 21078/JSSI-2022-620-13

[7] Keren He, Qian Jiang. Research on stock prediction algorithm based on CNN and LSTM. Academic Journal of Computing & Information Science (2022), Vol. 5, Issue 12: 98-106. https://doi. org/10. 25236/ AJCIS. 2022. 051215.

[8] B. Zhao, H. Lu, S. Chen, J. Liu and D. Wu, "Convolutional neural networks for time series classification, " in Journal of Systems Engineering and Electronics, vol. 28, no. 1, pp. 162-169, Feb. 2017, doi: 10. 21629/JSEE. 2017. 01. 18.

[9] Tong Wang, Panji Wang, Menghan Yu, Compare CNN and LSTM Model to Forecast the Stock Price. Financial Engineering and Risk Management (2022) Vol. 5: 44-52. DOI: http://dx. doi. org/10. 23977/ ferm. 2022. 050607

[10] O. B. Sezer, A. M. Ozbayoglu, " Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach, " Applied Soft Computing, 70 (2018), pp. 525-538