Stock Price Prediction Using Deep-Learning Models: CNN, RNN, and LSTM

Ruixun Cao

International Business School Suzhou, Xi'an Jiaotong-Liverpool University, Suzhou, China, 215123

Abstract: With the rapid development of the economy, stock markets or equity markets have an important role nowadays. More and more people participate in stock investment, the rise or the fall in prices is vital and closely related to investors' earnings. The basic way uses linear or non-linear algorithms, but the stock market has many factors, so it is highly non-linear prediction, so it is helpless to use one simple model, so this paper proposes to figure out a good deep-learning model to capture and analyze the data of six companies from Yahoo Finance by comparing the fitness of three famous neural network: CNN, RNN, and LSTM. The Sliding-Window model was applied to make future predictions in time series. The results of the models were calculated by using MSE, MAE, and MAPE.

1. Introduction

In the past few decades, the prediction of financial assets has been extensively studied. The definition of prediction is to summarize and analyze the current data based on the historical past data to process the data to obtain the results of the future data trend. Forecasting is a very important task. It is of great significance in many fields, including the economy, chemical industry, environment, agriculture, and other fields. We need to understand the future situation in order to make better decisions. Forecasting is to predict possible future results by analyzing a series of data for a period of time. In the stock market, the data we want to predict is the price of the stock. We can get possible future trends by collecting and analyzing previous data. By forecasting the future price of the stock market, we can make better and more informed investment choices to reduce investment risks and obtain higher returns. The current prediction method for stock prices is generally a time-series forecast and fundamental analysis. Fundamental analysis is widely used for long-term stock holding selection. It will analyze the price of the stock company, sales, profits, and economic influence. Another method is time series prediction. There are two types of algorithms, they are nonlinear models and linear models. There are various linear models like AR, ARIMA. Its working principle is to create a mathematical formula to match its image with the given price fluctuation chart. The biggest problem with such a working method is that he only created a fixed mathematical formula for matching, but ignored the underlying logic in it. It did not find out the dynamic changes in the stock market and the mutual influence of stocks, so it is difficult for linear models to adapt to the highly changing stock market and possible changes in the future. Nonlinear models usually use deep learning algorithms, including RNN(Recursive Neural Network), LSTM(Long Short-Term Memory), and CNN(Convolutional Neural Network) etc. As a result of the highly non-linear condition of the stock market, these deep-learning models perform well in predicting future data, people researched to test how it behaves in prediction in multi-dimension space [1] and it showed a good and effective performance. Another research used RNN for crop field prediction [2], which was very challenging because it depended on many factors. And RNN was very fit for predicting, so it may also be suitable for stock prediction. Also, there is another RNN prediction like air pollution which was also accurate [3]. RNN was used in stock prediction compared to other models in past research done by others [4], and there is another group of researchers focused on RNN model for price predictions [5], so RNN was chosen to do the test. LSTM is also a common model for prediction, it was trained to predict highway trajectory to test its accuracy [6]. The outstanding advantage is LSTM has the forget gate which allows it to do continual prediction [7]. LSTM is widely used for prediction and has made many improved models based on it, like AT-LSTM in time series [8], SS-LSTM for pedestrian trajectory prediction [9], and CT-LSTM for air quality prediction [10]. These examples show that LSTM has huge potential in many complex and multi-factor fields, so the LSTM model was selected to make stock forecasts. CNN model is a popular and efficient algorithm in the area of finance, which can be combined with many other models to produce more targeted models. A research composed CNN with bi-directional long short-term Memory (BiLSTM), and attention mechanism (AM), which was a useful application [11]. CNNpred used a diverse set of input data from five major U.S. stock market indices for stock price prediction [12] and achieved good

* Corresponding author: Ruixun.Cao23@student.xjtlu.edu.cn

results. With the massive combinations of CNN in deep learning, the analysis of data and forecasts become more efficient. CNN is also used to be combined with other neural networks like RNN and LSTM, there are different models of CNN-LSTM and CNN-RNN architectures that have good contributions in many fields, like residential energy consumption [13] and anti-cancer peptides [14]. So finally these three deep-learning non-linear models were chosen to research to figure out which one performs better independently.

2. Methodology

The research is done by using the Sliding-Window model which is an approach to make a short-term future prediction. The size of the window was fixed to be xx minutes, including an overlap of the information in the past xx minutes, and according to the given information to predict the xx minutes in the future. The way to identify the best length of the window is by calculating the error for different window sizes.

The three different deep-learning architectures, RNN, LSTM, and CNN were used for the work. RNN(Recursive Neural Network) is a circulating neural network used to process array data. It was first proposed by M. I. Jordan and Jeffrey Elman in 1990 [15], unlike traditional forward-feed neural networks. RNN has a cyclic connection that allows it to maintain a memory state when processing arrays. In RNN, each step has a hidden state, which can receive the input of the current time step and the hidden state of the previous time step as input. The output of the hidden state depends not only on the input of the current time step but also on the input of all previous time steps.

$$h^t = f(h^{t-1}, x^t; \theta) \tag{1}$$

LSTM(Long Short-Term Memory) is a kind of improved RNN architecture. It was created by Sepp Hochreiter in 1997 [16]. LSTM introduces a memory cell, which can store and access information and control the flow of information through a gate mechanism. The key parts of LSTM include the input gate, forget gate, and output gate.

- Input gate: It consists of the input information.
- Cell State: Can add or remove information through the whole network.
- Forget gate: Decide the fraction of the information.
- Output gate: It includes the output of the LSTM.
- Tanh layer makes a new vector added to the state.
- Sigmoid layer outputs a number between 0 and 1. 1 means retaining this value completely, and 0 means completely throwing away this value.

$$f_t = \sigma\left(W_f.[h_{t-1}, x_t] + b_f\right) \tag{2}$$
$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{3}$$
$$c_t = \tan h\,(W_c.[h_{t-1}, x_t] + b_c) \tag{4}$$
$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{5}$$
$$h_t = o_t * \tan h(c_t) \tag{6}$$

Where x_t: input vector, h_t: output vector, c_t: cell state vector, f_t: forget gate vector, i_t: input gate vector, o_t: output gate vector and W,b are the parameter matrix and vector.

CNN(Convolutional Neural Network) is a deep learning model or multi-layer perception similar to artificial neural networks and it was developed by Yann LeCun, Wei Zhang and Alexander Waibel in 1987-1989 [17]. It extracts the local feature layers of the image by sliding various convolutional cores on the image and finally obtains complex graphic features. The aim of using the three models is to explore whether there is any long-term dependency in the given data which can be identified from the performance of the models. The difference is that RNN and LSTM use previous data in the process to identify long-term dependencies, however, CNN only uses the input data. The error percentage was calculated using mean squared error(MSE), mean absolute error(MAE), and mean absolute percentage error(MAPE). MSE is a widely adopted loss function that measures the average of the squared differences between predicted and actual values. Instead of squaring the error term, MAE takes the absolute value of the difference between the predicted and actual values. MAPE is the average of the absolute differences between actual and predicted values, expressed as a percentage of actual.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y_i})^2 \tag{7}$$
$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|Y_i - \widehat{Y_i}\right| \tag{8}$$
$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{actual_i - predicted_i}{actual_i}\right| \tag{9}$$
$$n = number\ of\ considered\ points$$
$$actual_i = actual\ value$$
$$predicted_i = predicted\ value$$

3. Results and discussion

The research was completed by using three different models, the result of their error is given in Table 1, Table 2, and Table 3. From the table, it is clear that the RNN model usually gives a more accurate prediction than the other two models. The three tables all show that conclusion. Of the six companies chosen for the experiment, RNN performed best in five companies, and CNN was the worst. For example, in Fig. 1, the MSE of APPLE, RNN gets 8.68, LSTM gets 38.32, and CNN gets 359.80. And in Fig. 3, the MAPE of TESLA, RNN gets 2.69, LSTM gets 4.69, and CNN gets 6.07. For the other company, CNN was the best, and LSTM was the worst. The reason is that RNN is particularly suitable for processing sequence data, it relates the output of the current moment to the input of the next moment through a loop connection, which can capture the temporal dependence in the sequence. Besides, because the stock market has the characteristics of high change, the stock market is a research object that relies on short-term analysis. LSTM can handle both short-term dependency issues and long-term dependency issues, therefore, it has a more focused analysis and thinking on earlier historical data, to have a lower reference value for the current highly changing stock market. CNN is particularly suitable for

https://doi.org/10.1051/shsconf/202419602004

processing data with spatial structures such as images and videos. Through the combination of convolutional layers and pooling layers, CNN can effectively extract local features of images, so it is less suitable than RNN and LSTM in stock prediction. For companies, among the six companies, SONY, AMD, and BMW have more precise predictions than the other three companies. In Fig. 2, SONY has the minimum error percentage of RNN, which is 1.13, AMD gets 1.32 and BMW gets 1.41, quite smaller than APPLE's 2.17, NVIDIA's 4.30 and TESLA's 5.66. The same model has different prediction accuracy for different companies, because companies belong to different market types, and the changes they encounter in future development and industry trends are also different, so the model will have different prediction accuracy for companies of different market types. The future trend change of some companies is unpredictable because of the development of technology and changes of policies from the government, so the model cannot guess the future trend from the historical data, so the model will have lower forecast accuracy than that of the market that has not changed the factors. That means, if an industry is in a state of long-term and stable development, the sudden change to it is small externally, and the model will often have more accurate predictions of companies in such markets.

Table 1: ERROR PERCENTAGE (MSE)

COMPANY	RNN	LSTM	CNN
APPLE	8.68	38.32	359.80
AMD	2.89	8.85	35.83
BMW	3.48	8.78	89.35
NVIDIA	42.84	118.35	16.69
SONY	2.38	5.59	7.92
TESLA	58.10	160.42	276.10

Table 2: ERROR PERCENTAGE (MAE)

COMPANY	RNN	LSTM	CNN
APPLE	2.17	5.04	18.12
AMD	1.32	2.41	4.86
BMW	1.41	2.40	8.48
NVIDIA	4.30	6.65	3.10
SONY	1.13	1.91	2.31
TESLA	5.66	9.91	13.60

Table 3: ERROR PERCENTAGE (MAPE)

COMPANY	RNN	LSTM	CNN
APPLE	1.17	2.78	9.78
AMD	1.32	2.42	4.77
BMW	1.42	2.42	8.37
NVIDIA	5.21	8.23	4.61
SONY	1.31	2.21	2.63
TESLA	2.69	4.69	6.07

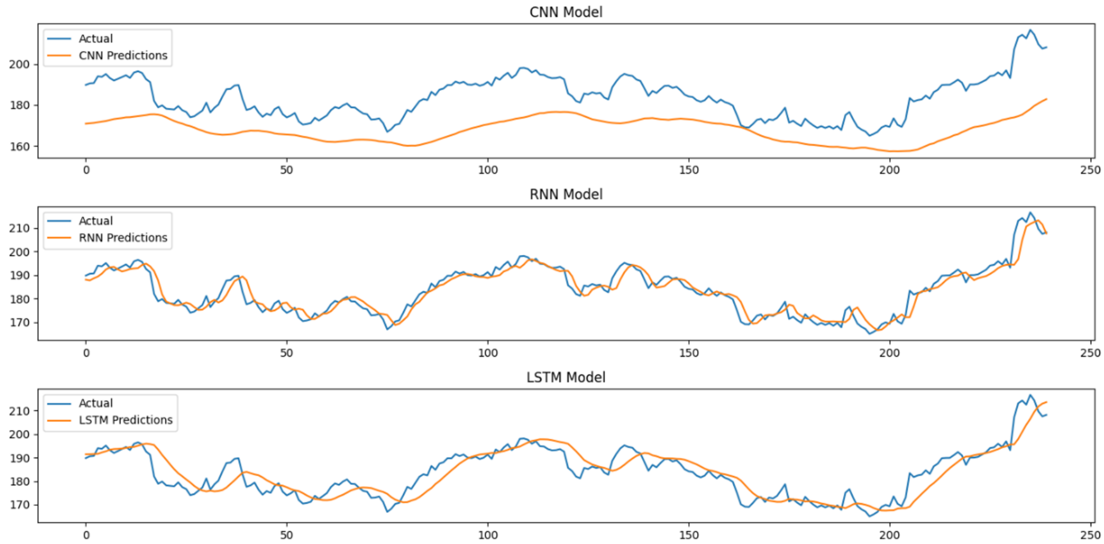

Fig. 1: Plot for Real value vs Predicted value for APPLE

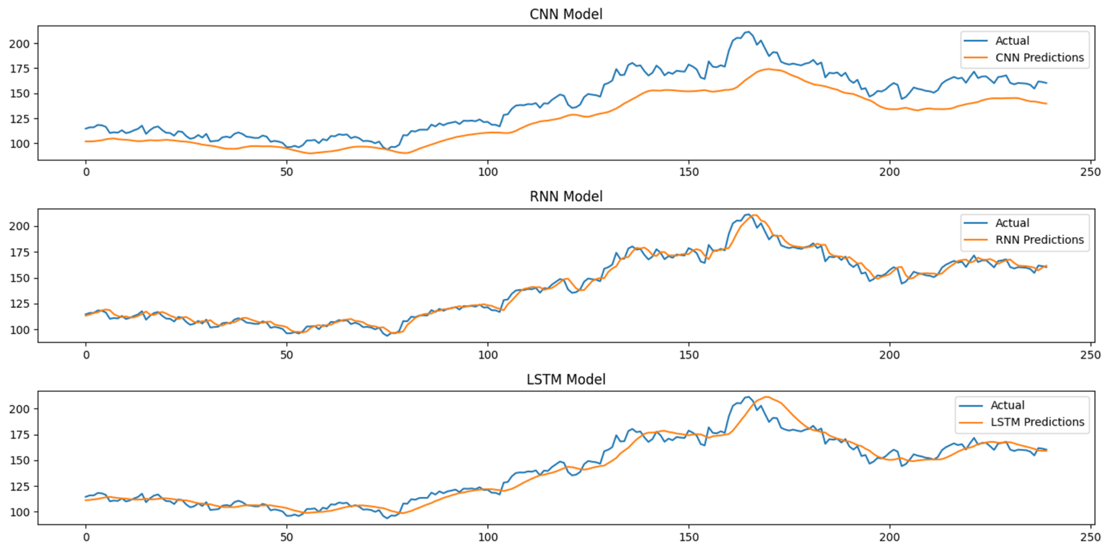

Fig. 2: Plot for Real value vs Predicted value for AMD

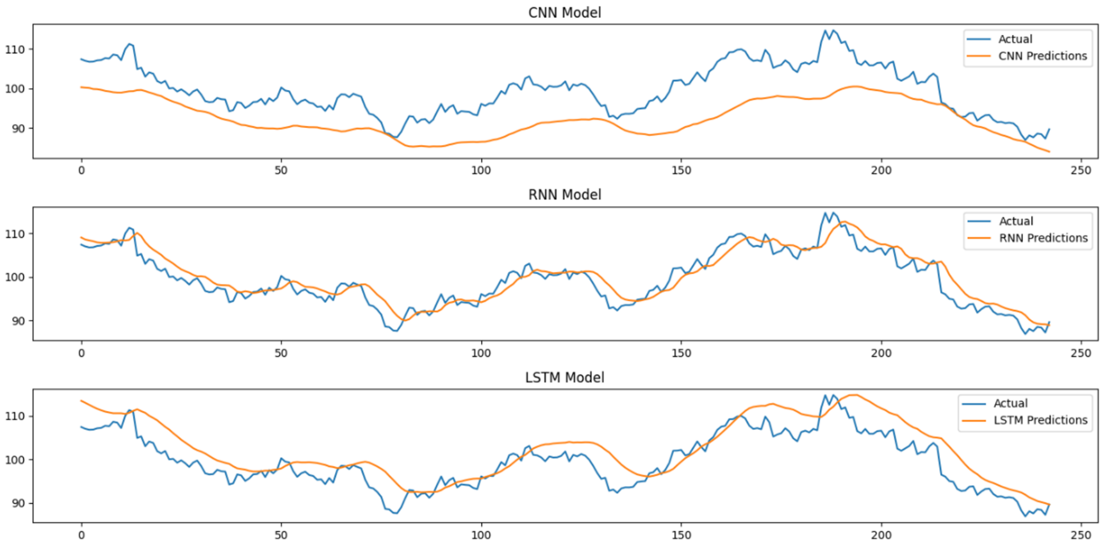

Fig. 3: Plot for Real value vs Predicted value for BMW

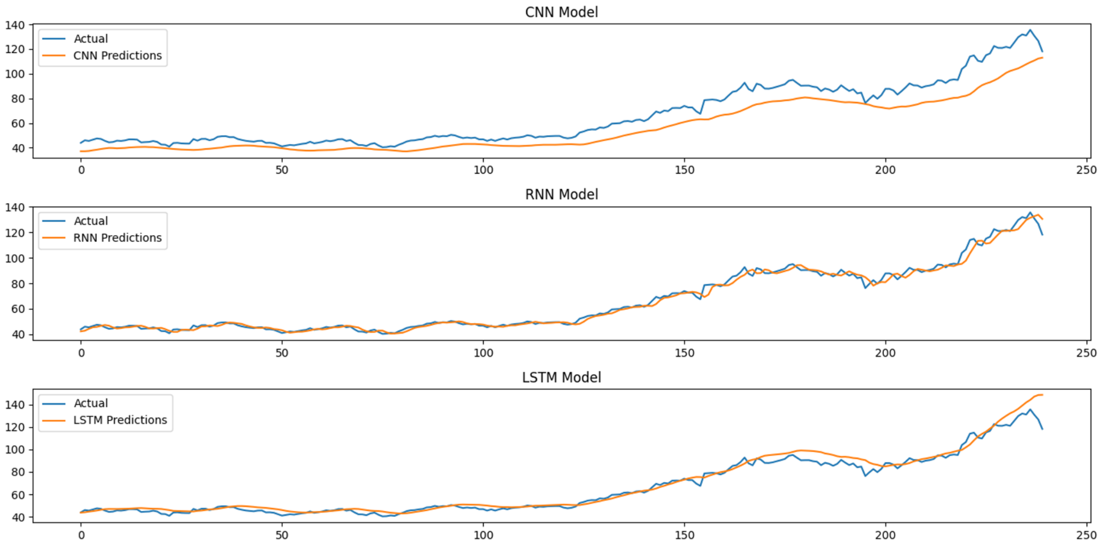

Fig. 4: Plot for Real value vs Predicted value for NVIDIA

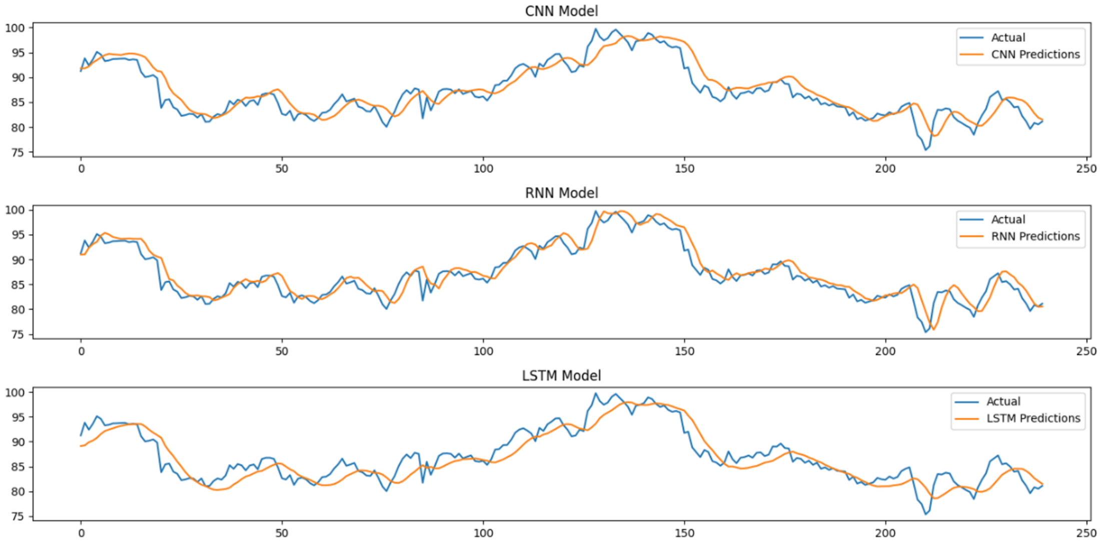

Fig. 5: Plot for Real value vs Predicted value for SONY

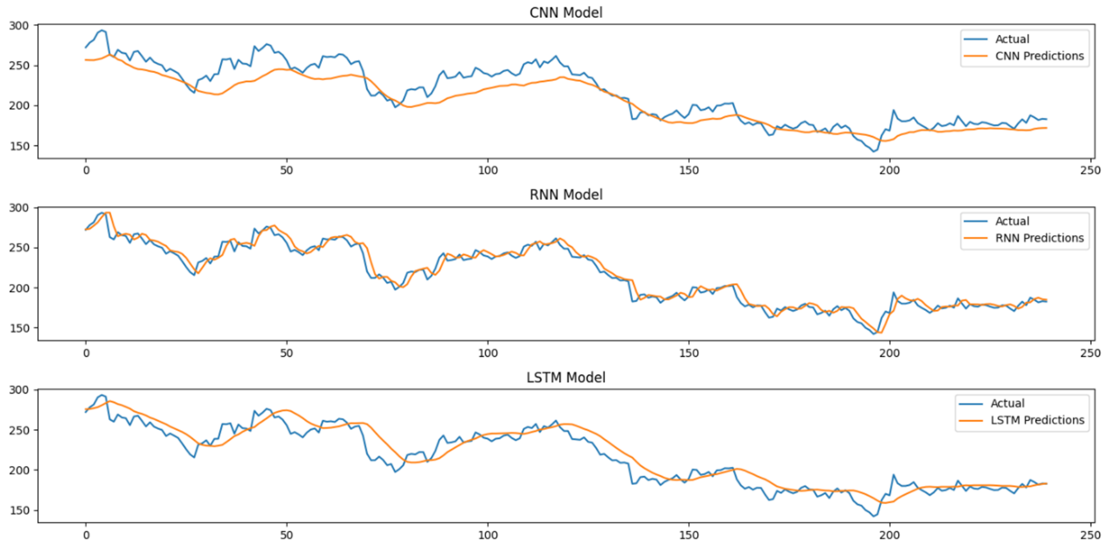

Fig. 6: Plot for Real value vs Predicted value for TESLA

From Figure 1 to Figure 6, the prediction of CNN would decrease compared to the actual value. At the same time, its prediction of fluctuations and trends is not ideal. The images of the other two models, RNN and LSTM, can roughly match the actual picture, but RNN responses are more efficient and accurate, which can almost capture the trend compared to LSTM. The prediction effect of these three models on different periods is similar on average between the earlier time and the future. It can be seen that these models still perform well when they train the predicted results again because of the deep learning architecture.

4. Conclusions

The deep-learning-based formalization was used for stock price prediction. We use three popular and advanced models, CNN, RNN, and LSTM, trained to use the data of six companies to predict their future prices. The data of companies was received from Yahoo Finance in the past five years, and by using the Sliding-Window model to compare the prediction results with the actual results. It turns out that the network architectures can capture the hidden factor and make accurate predictions. Among them, the RNN model performs the best in identifying the changes in trends, because it can learn from short-term historical data to predict the future price. The LSTM model has a lower accuracy because it will consider long-term and short-term historical data together, and the stock market is a highly changeable industry, its future trend mainly depends on recent changes, and the long-term changes have a relatively small impact on it. Therefore, because this model considers less effective historical data, it will have a slightly lower performance. The CNN model is good at processing data in high dimensions and iterating with gradient descent, which makes it easy to converge the results locally rather than globally optimally, therefore, it has good performance in the field of image and video, but its performance in stock price prediction is not as good as the other two models. Through this research, we hope to find better suggestions for investors and researchers who need references for stock market forecasting, refer to the forecast results given by these models to better reduce investment risks and make informed investment choices. At the same time, our research also has some shortcomings. For example, our next future research direction is to explore whether a single company's data can be applied to the whole market or other companies in different industries. We hope to maximize the potential of these deep-learning models so that they can be better and more universally applied in various industries.

References

1. Zhang, J., & Man, K. F. (1998, October). Time series prediction using RNN in multi-dimension embedding phase space. In SMC'98 conference proceedings. 1998 IEEE international conference on systems, man, and cybernetics (cat. no. 98CH36218) (Vol. 2, pp. 1868-1873). IEEE.
2. Khaki, S., Wang, L., & Archontoulis, S. V. (2020). A CNN-RNN framework for crop yield prediction. Frontiers in Plant Science, 10, 1750.
3. Fan, J., Li, Q., Hou, J., Feng, X., Karimian, H., & Lin, S. (2017). A spatiotemporal prediction framework for air pollution based on deep RNN. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4, 15-22.
4. Pawar, K., Jalem, R. S., & Tiwari, V. (2019). Stock market price prediction using LSTM RNN. In Emerging Trends in Expert Applications and Security: Proceedings of ICETEAS 2018 (pp. 493-503). Springer Singapore.
5. Shin, D. H., Choi, K. H., & Kim, C. B. (2017). Deep learning model for prediction rate improvement of stock price using RNN and LSTM. The Journal of Korean Institute of Information Technology, 15(10), 9-16.
6. Altché, F., & de La Fortelle, A. (2017, October). An LSTM network for highway trajectory prediction. In 2017 IEEE 20th international conference on intelligent transportation systems (ITSC) (pp. 353-359). IEEE.

7. Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. Neural computation, 12(10), 2451-2471.
8. Zhang, X., Liang, X., Zhiyuli, A., Zhang, S., Xu, R., & Wu, B. (2019, July). At-lstm: An attention-based lstm model for financial time series prediction. In IOP Conference Series: Materials Science and Engineering (Vol. 569, No. 5, p. 052037). IOP Publishing.
9. Xue, H., Huynh, D. Q., & Reynolds, M. (2018, March). SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction. In 2018 IEEE winter conference on applications of computer vision (WACV) (pp. 1186-1194). IEEE.
10. Wang, J., Li, J., Wang, X., Wang, J., & Huang, M. (2021). Air quality prediction using CT-LSTM. Neural Computing and Applications, 33, 4779-4792.
11. Lu, W., Li, J., Wang, J., & Qin, L. (2021). A CNN-BiLSTM-AM method for stock price prediction. Neural Computing and Applications, 33(10), 4741-4753.
12. Hoseinzade, E., & Haratizadeh, S. (2019). CNNpred: CNN-based stock market prediction using a diverse set of variables. Expert Systems with Applications, 129, 273-285.
13. Kim, T. Y., & Cho, S. B. (2019). Predicting residential energy consumption using CNN-LSTM neural networks. Energy, 182, 72-81.
14. Lane, N., & Kahanda, I. (2021). DeepACPpred: a novel hybrid CNN-RNN architecture for predicting anti-cancer peptides. In Practical Applications of Computational Biology & Bioinformatics, 14th International Conference (PACBB 2020) 14 (pp. 60-69). Springer International Publishing.
15. Elman, J. L. (1990). Finding structure in time. Cognitive science, 14(2), 179-211.
16. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
17. LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. Neural computation, 1(4), 541-551.