SN Computer Science (2022) 3:88
https://doi.org/10.1007/s42979-021-00970-5

ORIGINAL RESEARCH

Design of Machine-Learning Classifier for Stock Market Prediction

Akhilesh Kumar Srivastava[1] · Anand Srivastava[1] · Siddhartha Singh[1] · S. Sugandha[1] · Tripta[1] · Shubham Gupta[1]

Received: 16 July 2021 / Accepted: 12 November 2021 / Published online: 26 November 2021

Abstract

The stock market is complex in nature and it is very difficult to predict. Investors have many factors that affect the stock values. The stock market plays an important role in the financial aspect of the country's growth. The demand to predict stock values is very high, hence is the need for stock market analysis. This article is basically focused on always taking risks to invest his money in the stock market to gain profit. There are various machine-learning techniques available to predict the stock market. There are on predicting the stock market values. In the current scenario, the stock market forecasting is done using machine learning and artificial intelligence which makes the prediction process easier and based on the values of the current stock rate by training on the previous values. Basically, stock price prediction is based on time series data means every new data are dependent or based on previous data value. The dataset used for this is Dell daily stock for the period 17 Aug 2016–21 May 2021 which was used in this article. There are different kinds of models that can help in predicting the stock market. A simple machine-learning model cannot be applied to time series data, that is why we studied many models such as LSTM and ARIMA model, which are best for time series data. In addition, at the end, we saw that ARIMA is one of the best models for predicting the stock market values for short time series. This model is based on previous values. This model gives the more accurate and best results as compared to another one.

Keywords ARIMA model · LSTM · ACF · PACF · AIC

Introduction

The stock market is a financial network that offers a platform for almost every extensive economic transaction in the world at a dynamic rate called the stock market value that is based on market equity. Stock forecasting provides the greatest potential for arbitrage to gain a major source of research in this area. Previous stock price information by half a second can create a great profit.

The stock market offers a platform for economic transactions in the world at a dynamic rate which is called the stock market value. Stock forecasting is a method that helps to gain a major source of research in this field. Accurate predictions can be the reason of a great profit and a minor mistake in prediction can occur the reason for a huge loss.

Today's financial investors are faced with this trading problem as they do not fully understand which stocks should be bought or the shares will be traded for best results. Therefore, the purposed project will do minimize the problem with the appropriate accuracy you encounter in the real-world situation.

Financial time series prediction is extremely difficult, as Fama observed in [1], despite widespread acceptance of a

This article is part of the topical collection "Security for Communication and Computing Application" guest edited by Karan Singh, Ali Ahmadian, Ahmed Mohamed Aziz Ismail, R S Yadav, Md. Akbar Hossain, D. K. Lobiyal, Mohamed Abdel-Basset, Soheil Salahshour, Anura P. Jayasumana, Satya P. Singh, Walid Osamy, Mehdi Salimi and Norazak Senu.

✉ Akhilesh Kumar Srivastava
joinakhilesh@yahoo.com

Anand Srivastava
anand8355@gmail.com

Siddhartha Singh
siddharth.17bcs1035@abes.ac.in

S. Sugandha
sugandha.18bcs3001@abes.ac.in

Tripta
tripta.18bcs3003@abes.ac.in

Shubham Gupta
shubham.17bcs1071@abes.ac.in

1 Deptarment of Computer Science and Engineering, ABES Engineering College, Ghaziabad, India

weak type of market efficacy as well as a high noise level. Wang et al. [2] utilized ANN to prediction of prices of stock in 2003 and concentrated on capacity, which is a unique element of the stock. Among the significant conclusions was that increasing the capacity was unable to improve predicting performance on the datasets they studied, which were the S & amp P 500 and the DJI.

In [3], Ince et al. focused on the short projecting and used a SVM, the support vector machine model for the price prediction of stocks. The authors' key role is a assessment of MLP with that of SVM, which indicated that SVM outperformed MLP in the majority of situations; however, the outcome was also influenced by diverse transaction methods. Meanwhile, financial domain experts were studying stock market data using traditional statistical approaches and signal-processing techniques.

Short-term stock price prediction was also done by means of optimization methods, e.g., PCA, the principle component analysis [4]. Over time, scientists have attempted to evaluate stock market transactions such as volume burst hazards, broadening the stock market analysis study field and indicating that this study area still has a lot of promise [5]. Numerous projected solutions endeavored to merge deep learning (DL) and ML methods created on earlier methods as AI-based methods improved in recent years, such as Liu and Wang [6].

To assess different quantitative methods in stock markets, Liu et al. introduced a CNN as well as LSTM neural-network-based model in [7]. The CNN is used for stock selection, and it inevitably abstracts features based on numerical data, then uses an LSTM to retain time series structures to increase profits. To anticipate the stock market index, a new study offers a hybrid neural network design that syndicates a CNN with bidirectional memory [8].

There are many technologies that are used to resolve this issue related to stock market prediction such as ANN, Fuzzy Logic, and SVM. Recently, The ARIMA method was used for this problem in predicting the pattern. ARIMA has been done a successful job in the field of analyzing and predicting the time series. ARIMA is best known for short-term prediction. In this article, we use a daily fractional change in the stock value. Segment 2 of the paper presents the extensive literature survey and finds similar work in the area. The section next to this discusses the proposed methodology and results are presented in the next section.

Literature Survey

Lee et al. presented the comparison between the forecasting technique and reliability between the BPNN model and a time series (SARIMA) model in Korean Stock Exchange [9]. This article said that the SARIMA model forecasts are more accurate than the BPNN model for the KOSPI model. KOSPI model best predicts the nonlinear, uncertain, and volatile data. Forecasting the accurate value is totally dependent upon the developing process of predicting the model, this study is also helping to generate the procedure of the SARIMA model and BPNN model.

Adebiyi et al. presented the ARIMA model to forecast the future of the securities, stock, financial market moves by examining of regression analysis of their data [10]. Using the ARIMA model, the individual or any entity predicts the value of the stock in the stock market. Its main goal is to predict the differences between values in the series. The ARIMA model helps to predict the short-term price of the stock, which helps the investor to invest in the stock at the right point in time.

Rafiqul et al. presented a relative study of 3 models, Geometric Brownian Motion, ARIMA and ANN, that helps to forecast the future prices of the stock market [11]. The study shows that the ARIMA and Geometric Brownian Motion model is much better than the ANN in predicting the future price of the stock. ARIMA and stochastic model, both can be used for short-term prediction using the time series data.

Devi et al., shown in their paper is that inferences a new investment decision which is based on the less error percentage obtained [12]. This paper also highlighted the point on the next few years' future forecasting of each and every index.

Alma Sarah et al. presented a model that showed short-term predicting of the high technology procedure [13]. ARIMA models are implemented over the previous year dataset to improve short-term prediction. In this paper, they show the use of the method in banking stock market data verified with its accuracy. This study has shown that the method was limited to short-term forecasting.

Proposed Approach

ARIMA Model

ARIMA model stands for (auto-regressive integrated moving average). This model is a generalization of autoregressive and moving average models. It is best for short time series forecasting. Generally, time series analysis needs stationary data but stock market data are non-stationary data. These non-stationary data are handled using the approach of the ARIMA model which is introduced by Box and Jenkins in 1970. This model is best for predicting the stock market value. In this, the future value of a variable is based on a linear combination of past errors and past values:

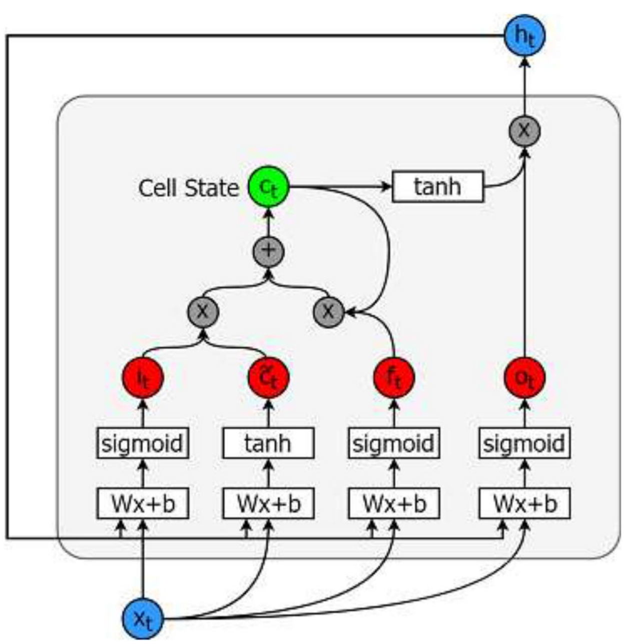
$$Y_t = \varphi_0 + \varphi_1 Y_{t-1} + \varphi_2 Y_{t-2} + \cdots + \varphi_p Y_{t-p} + \epsilon_1 - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q},$$

where Y_t is the actual value, φ_x and θ_y are coefficients, ϵ_t is random error at t, p and q are integers which are denoted as autoregressive and moving average.

LSTM Model

LSTM model is presented by Hochreiter and Schmidhuber [14]. Basically, LSTMs are intended to evade the long-term dependency problem. This model can predict an arbitrary number of steps into the future. It is useful for both long-term and short-term data modeling. It has five components.

Cell state (c_t)	It represents the internal memory of the cell which stores both short term memory and long-term memories
Hidden state (h_t)	This is the output state information which stored the previous calculated hidden state, current input, and current cell input which is eventually used to predict the future stock market values
Input gate (i_t)	It is used to decides how much information flows from current input to the cell state
Forget gate (f_t)	It is used to decide how much information flows into the current cell state from previous cell and the current input cell
Output gate (o_t)	It decides how much information flows from the current cell state into the hidden state, that helps to choose the long-term memories or short term memories and long-term memories



Equations are represented as follows:

- $i_t = \sigma(W_{ix}X_t + W_{ih}h_{t-1} + b_i)$.
- $\tilde{C}_t = \sigma(W_{cx}X_t + W_{ch}h_{t-1} + b_c)$.
- $f_t = \sigma(W_{fx}X_t + W_{fh}h_{t-1} + b_f)$.
- $c_t = f_t c_{t-1} + i_t \tilde{C}_t$.
- $o_t = \sigma(W_{ox}X_t + W_{oh}h_{t-1} + b_o)$.
- $h_t = o_t \tanh(c_t)$.

Methodology

This section is divided into three subsections. One deals with the data that are primarily used for building the model. Another two sections deal with describing various theories and processes for building the models. The performance of the model is tested by the following:

- Root mean square error
- Mean absolute error
- Mean square error and
- Regression score

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

$$\mathrm{MSE} = \frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{n}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{n}}$$

$$R^2 = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}.$$

Dataset

The dataset we used is the time series dataset. Time series data are also denoted as time-stamped data. This represents the data point sequence mentioned in order of time. Dell daily stock for the period 17 Aug 2016 to 21 May 2021 was used in this article. We used the *Pandas-Data reader* package in python software to collect the data directly from Yahoo Finance.

The dataset contains six variables:

- volume,
- close,
- low,
- high,
- daily open,
- adjusted close price.

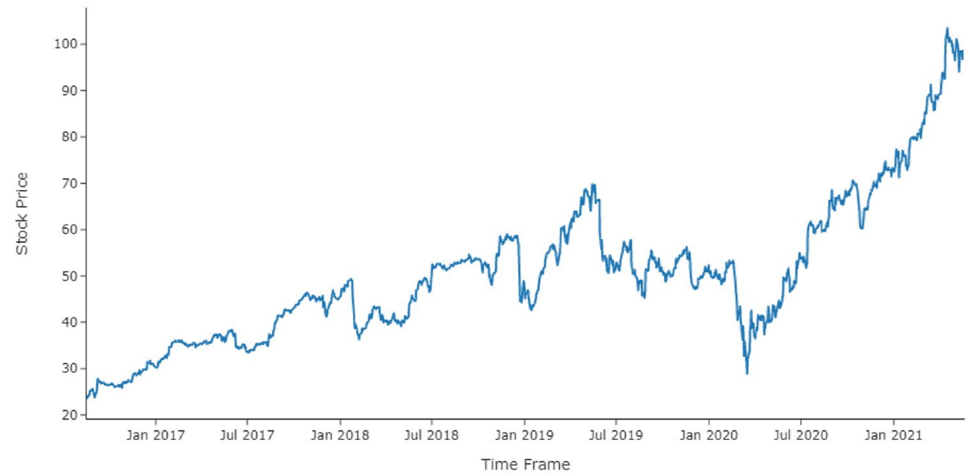


Fig. 1 Raw data time plot

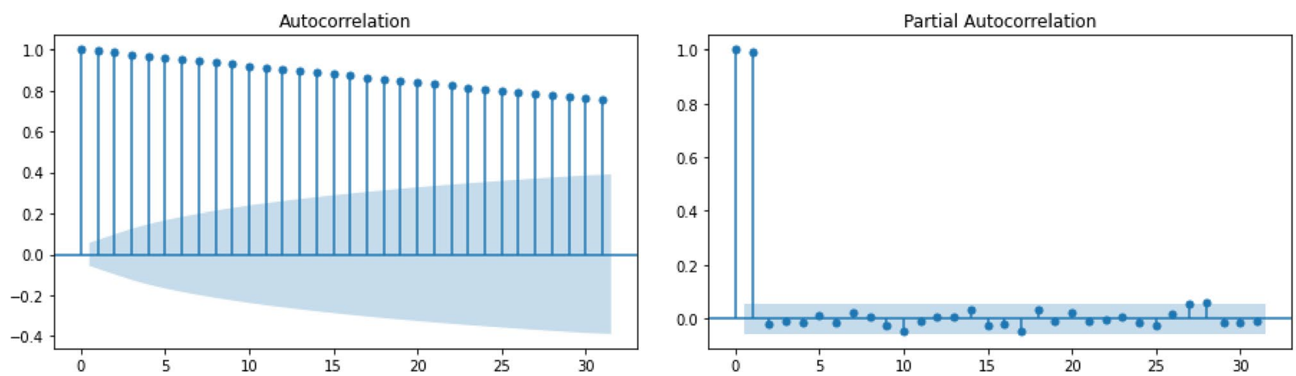


Fig. 2 Autocorrelation sample plot

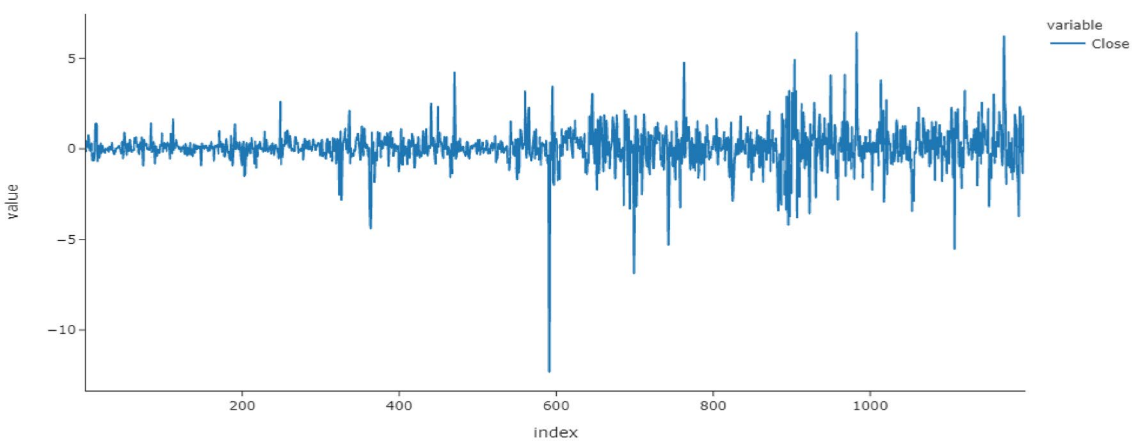


Fig. 3 First differenced log-transformed stock prices

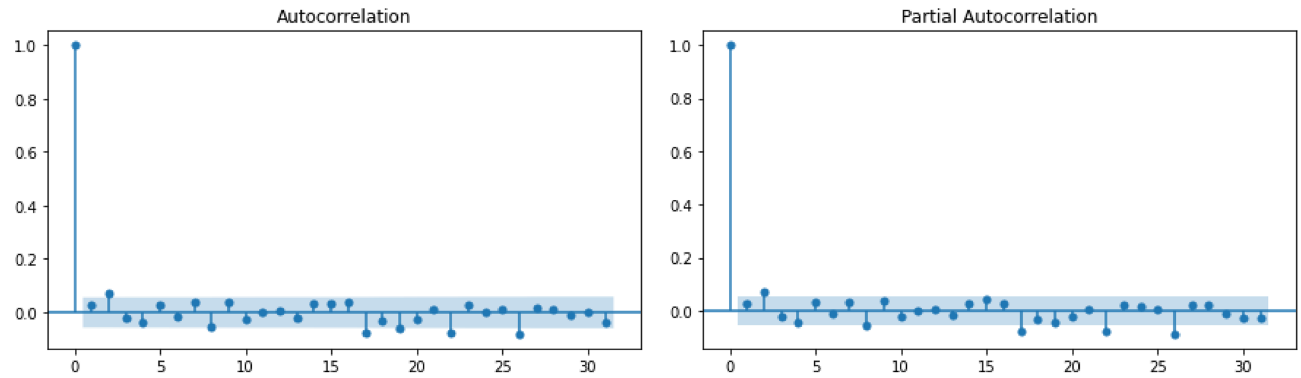


Fig. 4 ACF and PACF of the first differenced log-transformed stock price

Table 1 ADF test results

ADF statistic	− 0.18494419386651875
n_lags	0.9402925963938609
p value	0.9402925963938609
Critical values: 1%	− 3.435829423619109
Critical values: 5%	− 2.863959622178626
Critical values: 10%	− 2.5680582513898056

Table 4 LSTM model summary

Layer (type)	Output shape	Param #
lstm_1 (LSTM)	(None, 64)	16,896
dense_1 (Dense)	(None, 1)	65
Total params: 16,961		
Trainable params: 16,961		
Non-trainable params: 0		

Table 2 ARIMA (0,1,1) model comparison

Model	AIC	Time (s)
ARIMA (0,1,0)	2330.323	0.03
ARIMA (0,1,1)	2322.389	0.06
ARIMA (1,1,1)	2322.646	0.16
ARIMA (0,1,2)	2324.123	0.10
ARIMA (1,1,0)	2322.715	0.05
ARIMA (1,1,2)	2322.959	0.46

The model hence built do the prediction of stock prices for the current day based on the data of previous days.

ARIMA

The DELL stock price close is a time series that was analyzed to create the model. The time series is non-stationary, as shown in the graph below. The graph in Fig. 1 shows an upward trend. The auto correlation function (ACF) goes down slowly, and the partial autocorrelation function (PACF) shuts off at lag 1 with correlation one, as shown in Fig. 2.

The fact that the ACF is steadily falling indicates that the process is ongoing. In other words, the current value is linked to all previous values. The process will be non-stationary as a result of this. We turn the series into differenced log-transformed to make the process stationary. The graph of the first differenced log-transformed stock price is shown in Fig. 3. The data appear to be stationary and scrambled in this plot. The modified Dickey–Fuller test has proven stationarity (ADF). The ADF test's null hypothesis is that the time series is non-stationary. We can reject the null hypothesis and deduce that the time series is actually stationary if the *p* value of the test is less than the significance level (0.05). If the *p* value is more than 0.05, we must determine the order of differencing.

As we see the *p* value is greater than 0.05 which is 0.940 for the data, so we have to find the order of differencing. To find the order of differencing value, we have used *ndiffs*

Table 3 Error measures of ARIMA (0,1,1)

Error measures	MAE	MSE	RMSE	R2 score
	1.0585526723799226	2.14871772691582	1.4658505131546735	0.9932782407357383

function from *pmdarima* python package and in result found the value 1.

From the PACF and ACF plots of data in Fig. 4, the autoregressive and moving average orders p and q were found.

In both, PACF and ACF plots, there is no significant lag which indicates that the AR process of order $p=0$ and MA process of order $q=0$, i.e., ARIMA (0,1,0). Other ARIMA (p,d,q) models were also considered in this article. The most accurate model was picked using the Akaike information criteria (AIC) criteria; the lower the value, the more accurate the model is.

From Table 1, the ARIMA (0,1,1) model has minimum AIC value. In Table 2, there are 3 error factors MAE, MSE, RMSE. R2 score shows the Goodness of fit of the model.

Table 5 Error measures of LSTM

Error measures	MAE	MSE	RMSE	R2 score
	1.5855925149757775	4.514654849757082	2.1247717170927047	0.9857818665164186

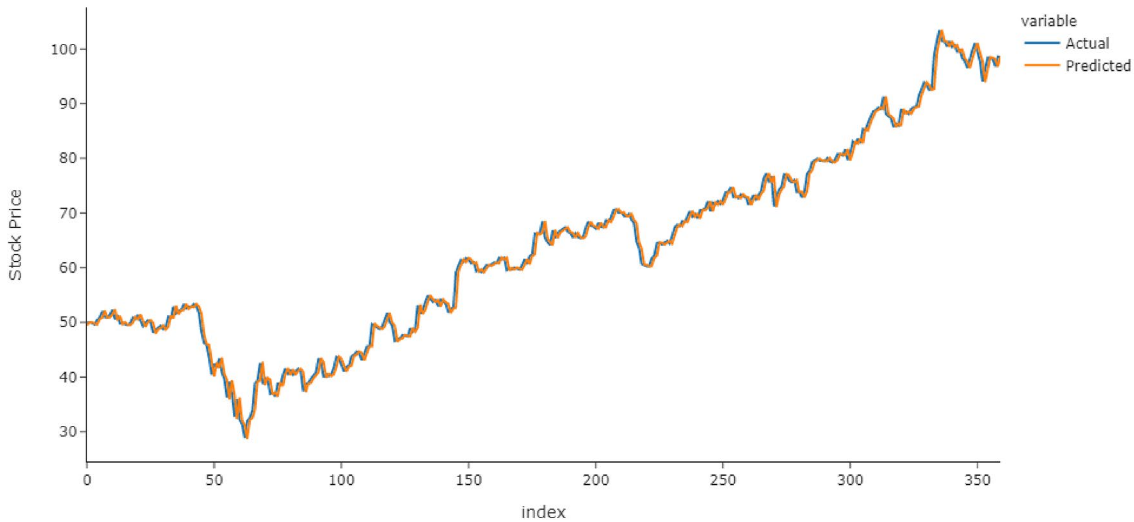


Fig. 5 ARIMA (0,1,1) model prediction

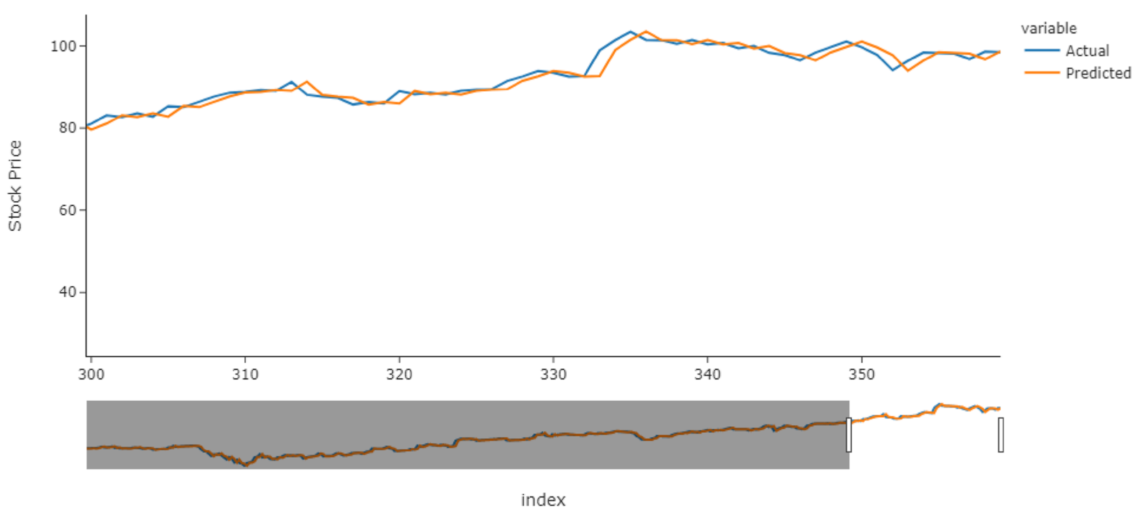


Fig. 6 Zoom view of ARIMA (0,1,1) model prediction

Table 6 Prediction by ARIMA (0,1,1) model

	Actual	Predicted	Error
1	50	49.97671	0.046588
2	49.89	50.00259	− 0.22568
3	49.62	49.87748	− 0.51891
4	50.46	49.59137	1.721431
5	50.83	50.55635	0.538376
6	52.01	50.86041	2.210314
7	50.98	52.13819	− 2.27185
8	50.91	50.85306	0.111842
9	51.39	50.91623	0.921906
10	52.29	51.44202	1.621683
11	50.57	52.38352	− 3.58617
12	51.08	50.37357	1.382995
13	49.8	51.15498	− 2.72085
14	50	49.65782	0.684369
15	49.62	50.03563	− 0.83762
16	49.51	49.57673	− 0.13479
17	49.83	49.50305	0.656138
18	50.86	49.86406	1.958199
19	50.6	50.96416	− 0.71968

LSTM

The first step after the data collection is data pre-processing, which is used for data transformation, data cleaning, and data integration. Data transformation involved data normalization, *MinMaxScaler* scales all the data to be in the region of 0 and 1. After the dataset is normalized and cleaned, the dataset is divided into training and testing sets. The testing data are kept as 30% of the total dataset.

We will utilize the LSTM RNN to create our model, which will use 70% of the data for training and 30% for testing. We optimize our model using mean squared error for training. We also utilized 300 epochs for training data, thus our model will look like this.

Table 3 shows a summary of the LSTM model and in Table 4, error factors that help to decide how the model is performing.

Result

In this result section, we discussed the above two models and an overview of the actual and predicted prices which is shown through a graphical representation.

ARIMA

To calculate error, the predicted values have been related through the real values. This comparison is shown in Table 5.

$$\text{Error} = \frac{\text{actual} - \text{predicted}}{\text{actual}} \times 100.$$

From Table 5, we can observe that the errors are less than ₹4 for the daily forecast. Relative errors lie in the range of negative 3.58617 to positive 2.210314. In Fig. 5, the graph is drawn for actual prices against the predicted figures. Some observations are presented in Fig. 5 as follows: predicted prices are very close to actual prices in ARIMA (0,1,1). The ARIMA (0,1,1) model's performance was assessed using Table 2 error measure, and Table 5 illustrates the comparison between test and projected results. Figure 6 is a zoomed-in version of Fig. 5.

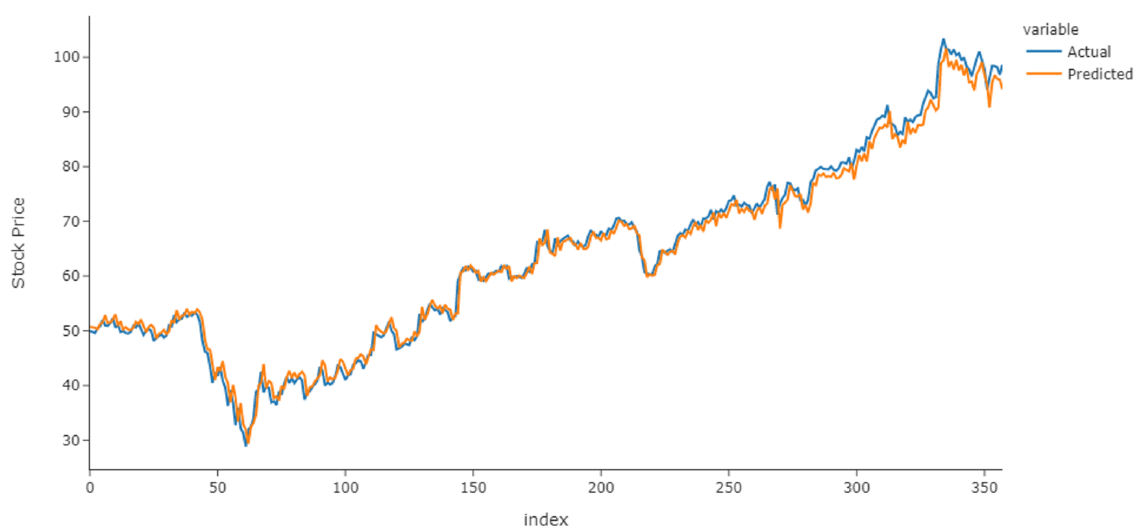


Fig. 7 LSTM model prediction

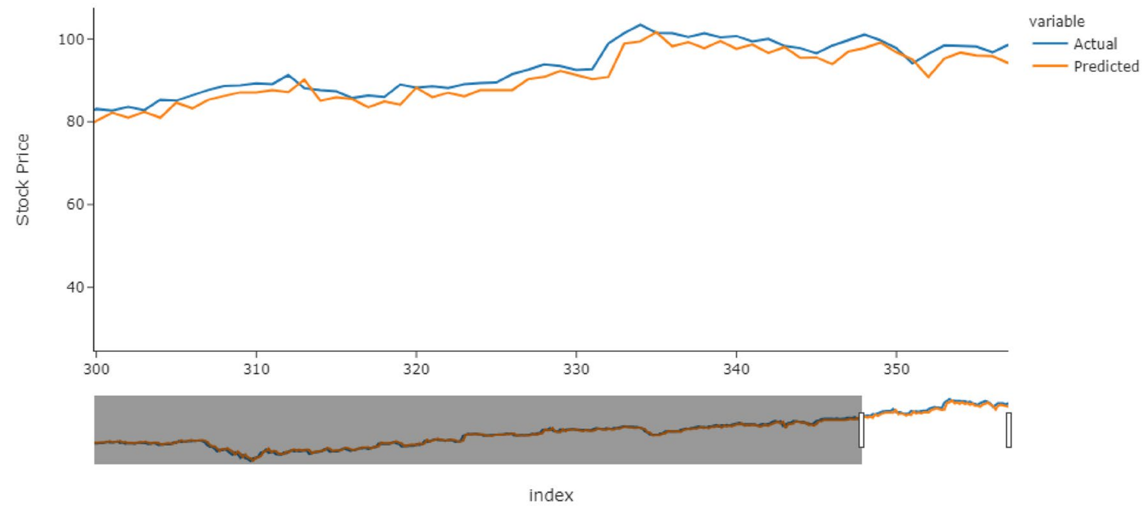


Fig. 8 Zoom view of LSTM model prediction

Table 7 Prediction by LSTM model

	Actual	Predicted	Error
1	50	50.56885	− 1.1377
2	49.89	50.58073	− 1.38451
3	49.62	50.43788	− 1.64828
4	50.46	50.14669	0.620905
5	50.83	51.09684	− 0.52497
6	52.01	51.39624	1.18007
7	50.98	52.64707	− 3.27004
8	50.91	51.35383	− 0.87179
9	51.39	51.40783	− 0.0347
10	52.29	51.93532	0.678292
11	50.57	52.85579	− 4.52004
12	51.08	50.8346	0.480429
13	49.8	51.62289	− 3.66042
14	50	50.15513	− 0.31026
15	49.62	50.53177	− 1.83751
16	49.51	50.09539	− 1.18236
17	49.83	50.03061	− 0.40258
18	50.86	50.40498	0.894646
19	50.6	51.50288	− 1.78434

LSTM

To calculate the inaccuracy, the anticipated prices were compared to the actual prices, as given in Table 6. Using the formula's unique error calculation,

$$\text{Error} = \frac{\text{actual} - \text{predicted}}{\text{actual}} \times 100.$$

Table 6 shows that the relative errors for the daily forecast are less than 5, with relative errors ranging from − 4.52004 to 1.18007. Figure 7 depicts a graph of the actual data and the LSTM model's predicted stock price value. The blue line in this graph reflects Dell's actual stock price, while the orange line represents its forecasted stock price. The evaluation of the performance of this model is evaluated by the evaluation from Table 6 error measure and from Table 4. It represents the comparison between Actual and predicted. Figure 8 shows the zoom view of Fig. 7.

Conclusion

We now discuss the collective results obtained from two the models discussed above. Table 7 presents the experimental output generated from the supplied models, whereas Fig. 9 graphically depicts the result.

The output of the ARIMA (0,1,1) model and the output of the LSTM model are quite close, and they sometimes match, as seen in Fig. 10.

When comparing the error measures in Table 8, it is evident that the ARIMA model outperforms the LSTM model when it comes to predicting the next-day stock price.

The goal of this research is to compare the performance of an LSTM model and a time series ARIMA model in forecasting. We discover the following using DELL data: first, the ARIMA model consistently outperforms the LSTM

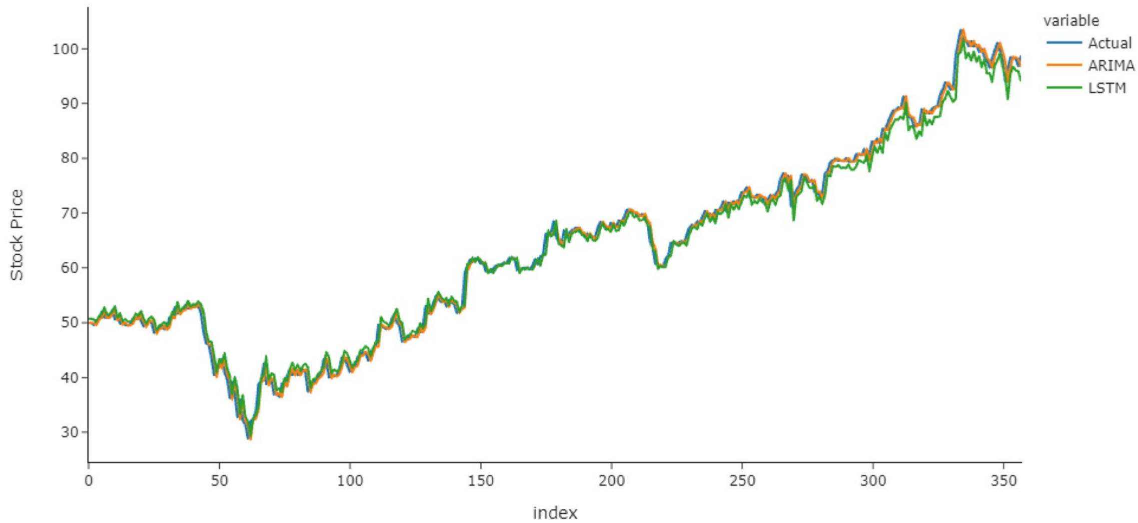


Fig. 9 Prediction by ARIMA (0,1,1) and LSTM against actual price

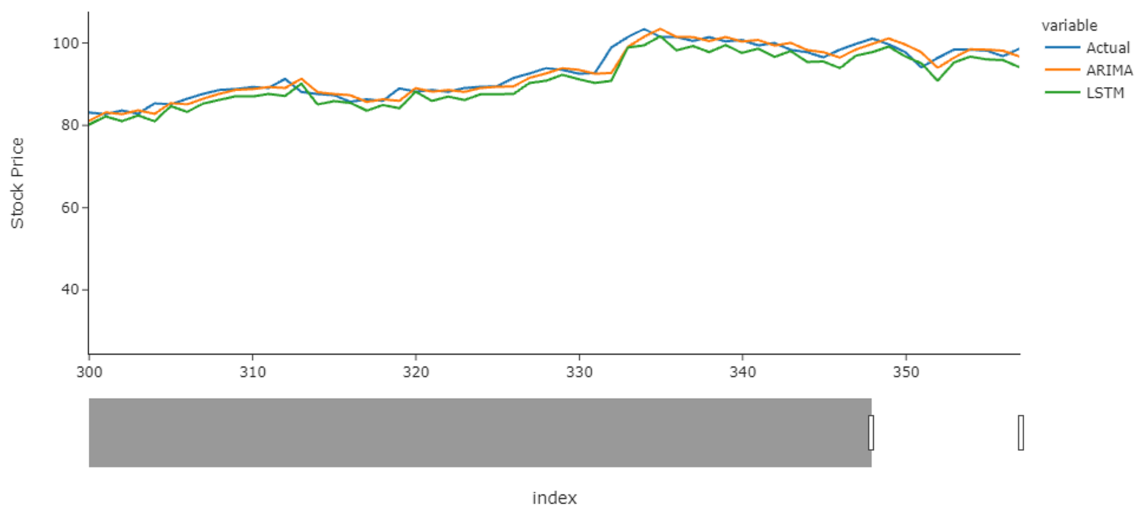


Fig. 10 Zoom view of prediction by ARIMA (0,1,1) and LSTM against actual price

model in terms of DELL results. Second, the error rate of the ARIMA model is less than the LSTM model (Table 9).

ARIMA performs better in short-term forecasting, while LSTM performs better in long-term forecasting.

ARIMA and other traditional methods focus on fixed temporal dependence: the relationship between data at various periods, which involves the study and definition of the amount of lag observations provided as input [15–23].

Declarations

Conflict of interest The authors state that no conflict of interest exists.

References

1. Malkiel BG, Fama EF. Efficient capital markets: a review of theory and empirical work. J Finance. 1970;25(2):383–417.
2. Wang X, Lin W. Stock market prediction using neural networks: does trading volume help in short-term prediction?. n.d.

Table 8 ARIMA (0,1,1) and LSTM, Sample results from the models

	Actual	ARIMA	LSTM
1	50	49.97671	50.70328
2	49.89	50.00259	50.70702
3	49.62	49.87748	50.55855
4	50.46	49.59136	50.26105
5	50.83	50.55635	51.24133
6	52.01	50.86041	51.53958
7	50.98	52.13819	52.81548
8	50.91	50.85306	51.45399
9	51.39	50.91623	51.51614
10	52.29	51.44202	52.05922
11	50.57	52.38352	53.0074
12	51.08	50.37357	50.91423
13	49.8	51.15498	51.7465
14	50	49.65782	50.23284
15	49.62	50.03563	50.65029
16	49.51	49.57673	50.20906
17	49.83	49.50305	50.1547
18	50.86	49.86406	50.54063
19	50.6	50.96416	51.66629

Table 9 Error measures comparison between ARIMA (0,1,1) and LSTM

Error measures	MAE	MSE	RMSE	R2 score
ARIMA	1.0585526723799226	2.14871772691582	1.4658505131546735	0.9932782407357383
LSTM	1.5855925149757775	4.514654849757082	2.1247717170927047	0.9857818665164186

3. Jeon S, Hong B, Chang V. Pattern graph tracking-based stock price prediction using big data. Future Gener Comput Syst. 2018;80:171–87. https://doi.org/10.1016/j.future.2017.02.010.
4. Lin X, Yang Z, Song Y. Expert systems with applications short-term stock price prediction based on echo state networks. Expert Syst Appl. 2009;36(3):7313–7. https://doi.org/10.1016/j.eswa.2008.09.049.
5. Shih D, Hsu H, Shih P. A study of early warning system in volume burst risk assessment of stock with big data platform. In: 2019 IEEE 4th international conference on cloud computing and big data analysis (ICCCBDA). 2019. p. 244–248. https://doi.org/10.1109/ICCCBDA.2019.8725738.
6. Liu G, Wang X. A new metric for individual stock trend prediction. Eng Appl Artif Intell. 2019;82:1–12. https://doi.org/10.1016/j.engappai.2019.03.019.
7. Liu S, Zhang C, Ma J. CNN-LSTM neural network model for quantitative strategy analysis in stock markets. 2017;1:198–206. https://doi.org/10.1007/978-3-319-70096-0.
8. Eapen J, Bein D, Verma A. Novel deep learning model with CNN and bi-directional LSTM for improved stock market index prediction. In: 2019 IEEE 9th annual computing and communication workshop and conference (CCWC). 2019. pp. 264–70. https://doi.org/10.1109/CCWC.2019.8666592.
9. Lee K, Yoo S, Jongdae JJ. Neural network model vs SARIMA model in forecasting Korean Stock Price Index (KOSPI). Issues Inf Syst. 2007;8(2):372–8.
10. Ariyo AA, Adewumi AO, Ayo CK (2014) Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th international conference on computer modelling and simulation. 2014. p. 106–112.
11. Islam MR, Nguyen N. Comparison of financial models for stock price prediction. J Risk Financ Manag. 2020;13(8):181.
12. Uma Devi B, Sundar D, Alli P. An effective time series analysis for stock trend prediction using ARIMA model for nifty Midcap-50. Int J Data Min Knowl Manag Process. 2013;3(1).
13. Almasarweh M, Al Wadi S. ARIMA model in predicting banking stock market data. Mod Appl Sci. 2018;12(11)
14. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
15. Edward A..J., & Manoj Jyothi. (2016).Forecast Model Using ARIMA for stock prices of Automobile sector. International Journal of Research in Finance and Marketing(Impact Factor 5.861), 6(4), 174-178.
16. Srivastava AK, Kumar Y, Singh PK. A rule-based monitoring system for accurate prediction of diabetes: monitoring system for diabetes. Int J E-Health Med Commun. 2020;11(3):32–53. https://doi.org/10.4018/IJEHMC.2020070103.
17. Moghar A, Hamiche M. Stock market prediction using LSTM recurrent neural network. Procedia Comput Sci. 2020;170:1168–1173. https://doi.org/10.1016/j.procs.2020.03.049.
18. Nochai R, Nochai T. ARIMA model for forecasting oilpalm price. In: Proceedings of 2nd IMT-GT reginal conference on mathematics, statistics and applications Universiti Sains Malaysia, Penang, June 13–15, 2006, 2006.

19. Roondiwala R, Patel H, Varma S. Predicting stock prices using LSTM. Int J Sci Res. 2015.
20. Srivastava AK, Singh PK, Kumar Y. A taxonomy on machine learning based techniques to identify the heart disease. In: Prateek M, Sharma D, Tiwari R, Sharma R, Kumar K, Kumar N, editors. Next generation computing technologies on computational intelligence. NGCT 2018. Communications in computer and information science, vol 922. Springer, Singapore; 2019. https://doi.org/10.1007/978-981-15-1718-1_2.
21. https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/. Accessed 5 Aug 2021 (**for complete understanding of LSTM model**).
22. https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/. Accsessed 5 Aug 2021 (**for complete understanding of ARIMA model**).
23. https://machinelearningmastery.com/how-to-load-visualize-and-explore-a-complex-multivariate-multistep-time-series-forecasting-dataset/ (**how to load, visualize and explore a time series dataset**).