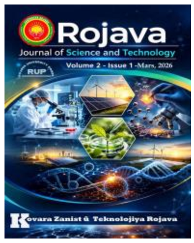
Rojava Journal of Science and Technology

2026, VOL. 2, No 1, [113-126]



LSTMNetworks versus ARIMA Models for Stock Price Prediction: A Case Study

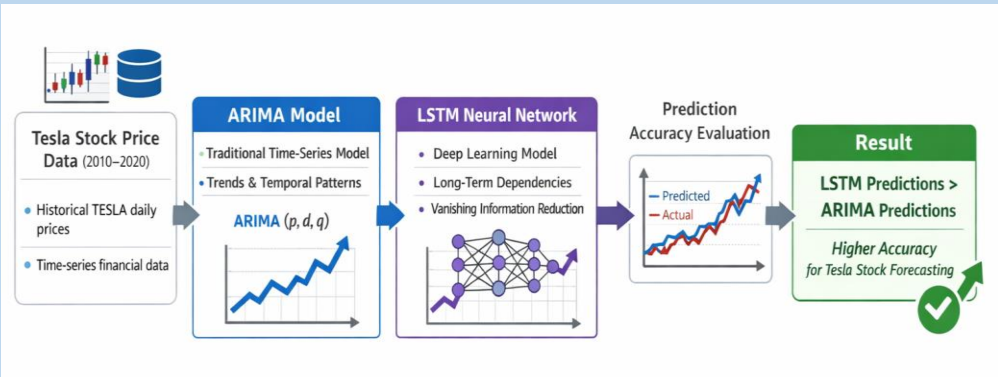
Asma Suleiman[1], Yasmine Abouelseoud*[2], Ahmed Tayel[2]

[1] Institute of Science and Modern Technology, Rojava university, Syria

[2] Faculty of Engineering, Alexandria University, Egypt

Abstract

There is no doubt that economy is of central importance at the present time. A strong economy is considered a fundamental asset of the strength of countries. Therefore, analyzing and predicting economy-related time series is necessary. In this paper, we use artificial neural networks, which are considered one of the most successful machine learning techniques, in the field of stock market price prediction. Long short-term memory (LSTM) network is chosen for this task as it is characterized by the ability to retain information for a long period of time and eliminate unimportant information, which reduces the effect of fading that traditional neural networks suffer from. Autoregressive integrated moving average (ARIMA) model has the ability to deal with temporal data that contains upward or downward trends, which makes it useful in analyzing financial data that are affected by temporal factors. However, as indicated in our case study of TESLA daily stock prices in the period from 2010 to 2020, LSTMnetwork-based predictions are more accurate than those obtained using the fitted ARIMA model.



ARTICLE HISTORY

Received 12 Jan. 2025

Revised 3 sept. 2025

Accepted 4 Dec. 2025

Keywords

Autoregressive integrated moving Average (ARIMA), forecasting, long short-term memory (LSTM), stock prices, time series.

* Corresponding author: yasmine.abouelseoud@alexu.edu.eg

1. Introduction

A time series is a set of data that depends on time. These data can be collected at regular or irregular time intervals. If the series has a specific pattern, then any value in the series must be a function of previous values. Time series models differ in the way they predict future values based on the current and previous values. There are now many ways to collect and analyze time series data [1]. The issue of forecasting is considered one of the most difficult and important problems that time series analysts must address. The performance and accuracy of the predicted data vary according to the type of data. The analysis is also affected
by unexpected events and changes in the structures from which information is collected. There is a large amount of research on forecasting models, and studies indicate that there are two categories of methods, including statistical methods and machine learning methods [2].

The ARIMA model is one of the most important and most widely used time series models, as the model involves a variety of exponential smoothing techniques. ARIMA models are versatile as they combine a variety of time series models, including an autoregressive (AR) model, and moving average (MA) model together with a differencing mechanism. The main drawback of such traditional models collectively is the
assumed linear form of the model [3]. The study in [4] indicated that ARIMA is a useful technique for evaluating broad changes in market prices and the government can use ARIMA to predict inflation.
Another type of model is based on deep learning. A commonly used deep learning model for time series prediction is the long short-term memory (LSTM) model. It is suitable for modelling time series data such as stock prices. It is a type of recurrent neural network (RNN). Although traditional RNNs can learn the sequential relationships in data, they suffer from the problem of inability to learn long-term dependencies.
Consequently, scientists developed the LSTMmodel to improve it. The gating mechanism in the cache unit of an LSTMmodel can adjust the long-term memory in a timely manner according to the current input and historical information [5].

With the focus on ARIMA and LSTMmodels for time series forecasting, many works in literature have utilized these powerful models for price forecasting [5, 6, 7, 8]. Comparative studies including the ARIMA and/or the LSTMmodels are presented in [5, 6, 7]. In [5] both the ARIMA and LSTMmodels are used to forecast the Google stock price. In this work the models' parameters are adjusted for the best accuracy performance measured in terms of mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE). The study results in the superiority of the LSM model over the ARIMA model. In [6], the Bursa Malaysia's closing prices are predicted during the COVID-19 crisis. The ARIMA, LSTM, and
a neural network with lag values fed as input are used in prediction. The compared models are evaluated in terms of RMSE and mean absolute percentage error (MAPE). The study concludes that the LSTMmodel has higher accuracy than the other related models. In [7], the LSTMmodel is also used to predict US stocks in (2016) and compared it to the BP neural network and the traditional RNN. The study confirms that the LSTMmodel has a higher accuracy in prediction. In [8], the superiority of the LSTMmodel is emphasized as it overcomes the limitations of the recurrence neural network models when dealing with large datasets. In this study, the HIK Vision stock price is taken as a case study. Moreover, wavelet denoising is used in data preparation step.

In general, the use of machine learning approaches in predicting stock prices plays an important role for their accuracy as indicated in the review study in [9]. This study performs a comparison between the different prediction techniques of traditional machine learning techniques, e.g., support vector machine (SVM), Deep-learning techniques, e.g., LSTM, time series techniques, e.g., ARIMA, and graph-based techniques. From this study the LSTMmodel showed remarkable accuracy compared to other models including the ARIMA model. The LSTMmodel works well with the non-linear time series and requires high training time and memory. However, the ARIMA model provide good results for linear data and provide good predictions for the short term.

In this paper, we focus on adjusting the data representation and pre-processing, as well as optimizing the parameters involved in both the ARIMA model and LSTMmodel. We will proceed in the paper as follows, examining the structure of the two models, the mechanism of their work, how to adjust the parameters for each model, then a practical application to predict TESLA stock prices using the two models, and finally evaluating the accuracy to choose the best model. The objectives of this study are summarized as follows:

1. Utilizing the LSTMand ARIMA models in the case study of forecasting TESLA daily stock prices.
2. Optimizing the models' parameters for the best accuracy.

3. Applying the wavelet transform as a pre-processing step to enhance the prediction accuracy.

The rest of this paper is organized as follows. The second section discusses our methodology used in forecasting. Section 3 includes the description and analysis of the data set used in our experiments. Additionally, the pre-processing of the data required to obtain more accurate results is described. The pre-processing processes considered include normalization and wavelet transform to get rid of the noise in the data. The fourth section presents the parameters of the models that are used in our experiments, in addition to evaluating the results obtained based on the root mean square error measure.

2. Materials and methods

In this section, we present the general framework for time series prediction. Next, we review the necessary background of the two models used in our study of forecasting stock prices.

2.1. General methodology for time series prediction

Various time series forecasting models share a common methodology. The first step depends on gathering previous historical data from different available resources. In the second step, the collected data is analyzed. This analysis proceeds to determine the trends and the stationarity properties of the time series under study. In the third step, an appropriate model is selected and then its parameters are estimated to provide the best fit to the available data. Finally, the accuracy of the model is evaluated. Various measures of accuracy can be employed such as the mean squared error or any another statistical test. The resulting model can then be used for actual prediction. A block diagram for this general methodology is shown in Figure 1.

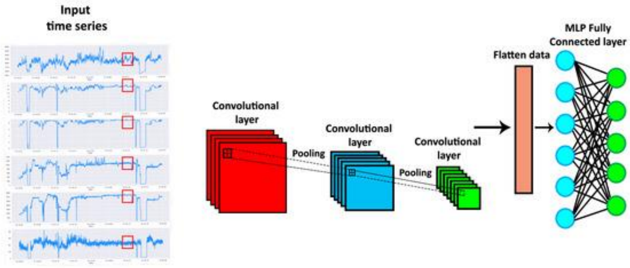


Figure 1: General methodology block diagram

2.2. Long short-term memory network (LSTM)

Traditional RNNs suffer from the problem of vanishing/exploding gradient. Learning long-term dependencies is also problematic to classical RNNs. LSTM networks have emerged to solve both problems [10]. The LSTM model main objective is to impose control over the process of learning long-term dependencies and prohibiting any negative effect on the quality of the learnt model. LSTM employs gates to aid it to remember the most important information. The cell structure lies at the heart of the LSTM network operation. It can be used to keep information without changing it. The status cell can also carry changes to the information through a group gates. The gates are in the form of a set of neurons that end with a sigmoid function and a set of positive multiplication operations. The LSTM network contains three kinds of gates to control the cell state. The operation of these gates is summarized in the following steps [11-12]. The components of an LSTM network memory cell are depicted in Figure 2.

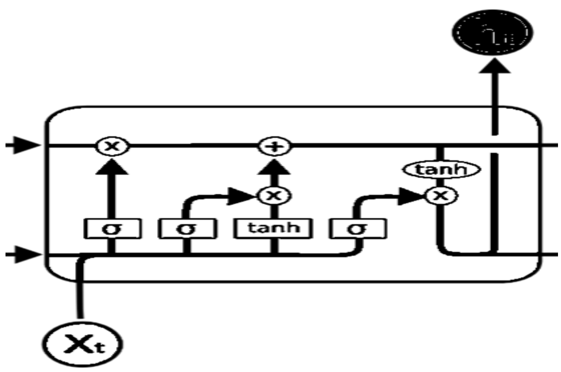


Figure 2: LSTMmemory cell architecture [13]

The first step is to decide on which information to be forgotten and that to be kept. This process takes place within the layer of the sigmoid activation function. This layer is called the forgetting layer.

$$f_t = \sigma\left(W_f.[h_{t-1}, x_t] + b_f\right) \quad (1)$$

where $W_f and\ b_f$ respectively denote weights and biases of the forgetting gate and σ denotes the sigmoid function.

The second step proceeds to identify the information that should be stored in the status cell and involves two sub-tasks: determining the value to be changed (i_t) and forming a vector of new values $\widetilde{C}_t$ as depicted in the following equations.

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

$$\widetilde{C}_t = \text{Tanh}\ (W_c.[h_{t-1}, x_t] + b_c \quad (3)$$

where $W_i$and $W_c$represent the associated weights, while b_i and b_c represent the corresponding biases.

The third step modifies the value of the previous state cell C_{t-1} to its new value C_t, where we multiply the value of the old state by f_t then add $i_t * \widetilde{C}_t$ to the new value as given in the following equation.

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \quad (4)$$

The fourth step determines the final output, which is based on the output of the state cell C_t as in equations (5) and (6).

$$o_t = \sigma(W_0.[h_{t-1}, x_t] + b_0) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

where W_0 and b_0 represent weight and biases of the out put gates and h_t acts as the output value of the present cell.

Deep LSTMs can be constructed by stacking multiple LSTMlayers vertically, with the output sequence of one layer forming the input sequence of the next (in addition to recurrent connections within the same layer), as shown in Figure 3.

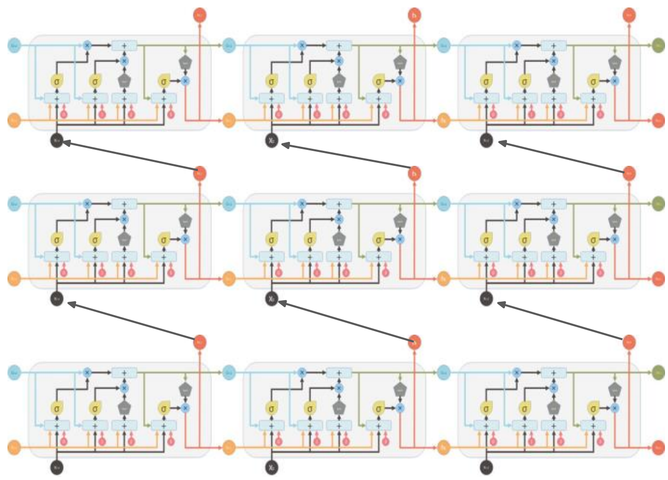


Figure 3: Deep LSTMnetwork [13]

The operation of LSTM model for forecasting time series data is depicted in Figure 4 [14-15].

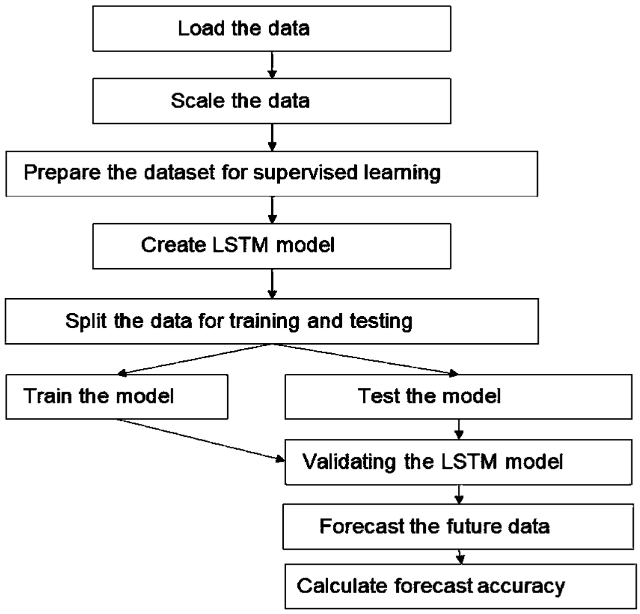


Figure 4: Block diagram of LSTM operation for predicting time series data [14]

2.3. Autoregressive Integrated Moving Average Model

An autoregressive integrated moving average (or ARIMA) model is an advanced statistical model that is commonly employed to either provide a better understanding of time series data or to anticipate future trends[16]. It is a kind of regression model that describes the impact of one dependent variable in terms of other variables representing previous values within the same time series. The model target is to forecast future changes by examining the differences between the values in the series instead of through actual values. ARIMA is comprised of three constituents [17]:

- Autoregression (AR): Autoregressive models analyze the history of the values in the dataset to make predictions about future values using assumptions about the relation between these past values and the current value.
- Integrated (I): The integrated component of the ARIMA model applies differencing steps to the data to transform it into a stable or stationary series.
- Moving Average (MA): The moving average ingredient of the ARIMA model considers the previous and current values of lagged variables to settle on the value of the output variable.

A standard notation for an ARIMA model is ARIMA(p, d, q) involving three key parameters, which are p, d, q that assume positive integer values. These parameters are described as follows [18]:

p: It is the number of lag observations in the model, referred to as the lag order, whose value is identified by analyzing the partial correlation function (PACF). This parameter represents the autoregressive terms number.

d: It is the number of times the actual observations are differenced; that is, the number of times the differencing operation is applied to accomplish temporal stability).

q: It represents the order of the moving average part of the model, whose value is identified by analyzing the partial correlation function.

The first step in constructing an ARIMA model is achieving stability of the time series; that is, converting the series into a stationary one. A time series yt is said to be stationary if it fulfills all of the following characteristics[19]:

E $(y_t) = \mu_y$ for all t (mean is the same at every period)

var $(y_t) = \sigma_y^2$ for all t (variance is same at every period).

Cov$(y_t, y_{t-k}) = \gamma_k$ for all t (autocovariance with respect to a particular lag is the same all the time)

It is common that a non-stationary series can be transformed into a stationary one if differencing is applied. For example, $y_t = y_{t-1} + \varepsilon_t$ can be non-stationary, but $w_t = y_t - y_{t-1}$ is stationary. Let Δ be the difference operator and B the backward shift (or lag) operator such that:

$\Delta y_t = y_t - y_{t-1}$ and $By_t = y_{t-1}$. Thus, $\Delta y_t = y_t - \ By_t = (1-B)y_t$ and $\Delta^2 y_t = (1-B)^2 y_t$. In general, the d^{th} order difference can be written as:

$$\Delta^d y_t = (1-B)^d y_t \qquad (7)$$

The number of times the raw series must be subjected to the differencing operation in order to accomplish stationarity is called the order of integration denoted by (d), as discussed above. The differenced series has $n-1$ observations after taking the first difference, and $n-d$ observations after taking (d) differences, where n is the number of observations in the original time series. In practical scenarios, it is common that the first or second difference are sufficient to achieve stability. This can be attributed to the fact that residuals data generally are close to being stable.

An ARIMA model with parameters (p, d, q) has predictive equations [20] that can be expressed as in equations (8)-(10):

Autoregressive model of order (AR(p)):

$$y_t = \delta + \emptyset_1 y_{t-1} + \emptyset_2 y_{t-2} + \cdots + \emptyset_p y_{t-p} + \varepsilon_t \quad (8)$$

y_t depends on its (p) previous values.

Moving average model of order (MA(q)):

$$y_t = \delta - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - .. - \theta_q \varepsilon_{t-q} + \varepsilon_t \qquad (9)$$

y_t depends on its (q) previous random error terms.

If y_t is of integral order (d), we write $y_t \sim I(d)$. Thus, y_t becomes a stationary series after a d^{th} order differencing, $\Delta^d y_t$ is represented by an ARMA(p,q) model.

Then, we say that y_t is an ARIMA (p, d, q) process that is:

$$y_t = \delta + \emptyset_1 w_{t-1} + \emptyset_2 w_{t-2} + \cdots . + \emptyset_p w_{t-p} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (10)$$

where $(1 - B)^d y_t = w_t$

As stated above, the basic step in fitting a suitable ARIMA model is to make sure that the series is stable and this is achieved by analyzing the autocorrelation function (ACF) and the partial autocorrelation function (PACF), where the autocorrelation function measures the relationship between y_t and y_{t-k} for different values of k (lags). If y_t is related to y_{t-1}, then y_{t-1} is related to y_{t-2}. ACF mathematical formula [21] is given by equation (11):

$$r_k = \frac{\sum_{t=k+1}^{n}(y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum(y_t - \bar{y})^2}, \quad k = 1,2,3 \ldots \quad (11)$$

where $\bar{y} = \frac{\sum_{t=1}^{n} y_t}{n}$.

The partial autocorrelation function measures the relationship between y_t and y_{t-k} after removing the effects of lags it is given mathematically as:

$$r_{kk} = \frac{r_k - \sum_{i=1}^{k-1} r_{k-1,i} r_{k-i}}{1 - \sum_{i=1}^{k-1} r_{k-1,i} r_i}$$

For instance, $r_{11} = r_1$ and $r_{22} = \frac{r_2 - \sum r_{2-1,j} r_{2-1}}{1 - \sum r_{2-1,j} r_j} = \frac{r_2 - r_{11*} r_1}{1 - r_{11*} r_1}$

The process for choosing the best ARIMA model to be fitted to a given time series is summarized in Figure 5.

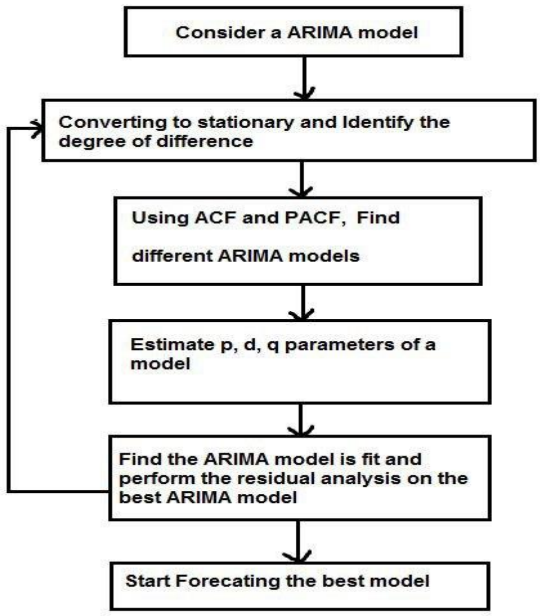


Figure 5: ARIMA model fitting process [22]

3. Time Series Dataset: Description, Analysis and Pre-processing

In our experiments, we use TESLA daily stock prices data from 2010 to 2020 downloaded from (https://www.kaggle.com/datasets/timoboz/tesla-stock-data-from-2010-to-2020?resource=download). It has four columns, each of which represents a time series starting with the opening price, which is the price of the asset in the trading market at the beginning of the day, as well as the highest price reached by trading, as well as the lowest trading price, in addition to the closing price, which represents the final price for intraday trading.

3.1. Correlation between the four time-series

It is possible to find a mathematical representation of the degree of similarity between a certain time series and another series using the Pearson correlation coefficient. The Pearson correlation coefficient provides a measure of the strength of linear association between two variables [23]. It shows the effect of changes in one variable when the other changes. It is calculated mathematically by equation (12):

$$r = \frac{n \sum \sum xy - \sum x \sum y}{\sqrt{(n \sum x^2 - (\sum x)^2)(n \sum y^2 - (\sum y)^2)}} \quad (12)$$

The correlation coefficient value is between - 1 and +1, and each value corresponds to a type of correlation as shown in Table 1.

Table 1 : *The extent and types of correlation*

Type of correlation	**Correlation coefficient**
Perfect direct correlation	r = +1
Strong direct correlation	0.7 ≤ r < +1
Moderate direct correlation	0.4 ≤ r < 0.7
Weak direct correlation	0 < r < 0.4
There is no correlation	r = 0
Strong inverse correlation	-0.7 ≤ r < -1
Moderate inverse correlation	-0.4 ≤ r < -0.7
Weak inverse correlation	0 < r ≤ -0.4

We calculated the Pearson correlation coefficient between the series that represents the opening price and the series that represents the highest trading price during the day, we found that r = 0.999425, which indicates that the correlation between these two series is a strong direct correlation, according to Table 1. Likewise, we calculated the correlation coefficient between the closing price and the lowest trading price during the day, we found that r =0.999447, which indicates that the correlation is direct and strong as well.

3.2. Wavelet transform analysis

The financial time series data is vulnerable to several factors, such as political changes and variation in investors interests, and usually contains a lot of noise. To increase the ability to expand a fitted model, the data must be filtered from noise. When using a neural network, wavelet analysis can perform an accurate multi-scale analysis of signals through dilation and can effectively eliminate the existing noise in the data and retain the characteristics of the original signals. The wavelet transform decomposes each signal into a low-frequency signal and a high-frequency signal and only further analyzes the low-frequency components, as shown in Figure 6.

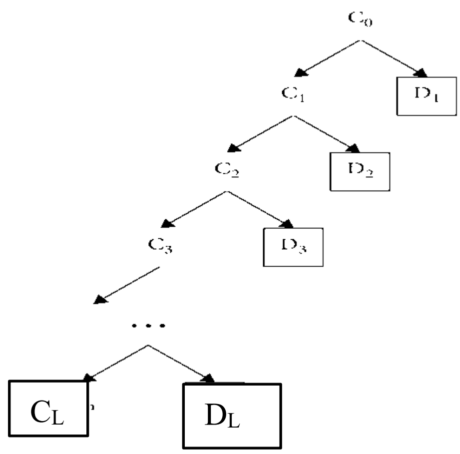


Figure 6: *Wavelet decomposition process [22]*

Assume that C0 is the original financial time series signal; C1, C2 , ..., CL and D1, D2..., DL are the first, second, and L-layer low-frequency and high-frequency signals, respectively. Then, the original time series can be mathematically expressed as follows:

$$C0 = Cl + DL + DL-1 + ... + D2 + D1 \qquad (13)$$

The low-frequency part presents the general trend of the time series, while the high-frequency part presents short-term random disturbance. Setting the high-frequency part to zero can eliminate the noise and can make the signal smoother [24].

3.3. Normalization and Averaging

Simple tools for smoothing a time series to reduce inherent fluctuations magnitude include normalization, where the original time series (y_t) is replaced by z_t, where

$$z_t = \frac{y_t - \bar{y}}{\sigma_y} \qquad (14)$$

with $\bar{y}$ being the mean value of the time series data and σ_y is the associated standard deviation. It is often beneficial in stock market analysis, to make predictions on a weekly or monthly basis. In our experiments, we found that more accurate predictions are obtained when the data is aggregated over a week or a month. The aggregation method used is a simple averaging process.

3.4. TESLA stock prices time series analysis

When analyzing and forecasting a time series, it is good to plot the data and pay attention to the distinctive characteristics of the time series, which guide the researcher in choosing the appropriate modeling strategy to be employed. Figure 7 shows the data for the opening price time series.

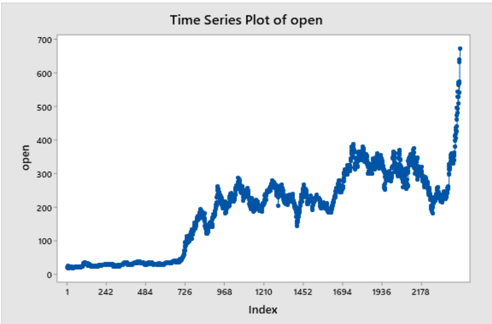


Figure 7: *Opening prices time series for TESLA*

First, the time series must be stationary before starting the procedures of fitting. In the case of a non-stationary series as in Figure 8, we try taking the first difference d = 1, which is ideal for making it a reasonably stationary series as shown in Figure 9.

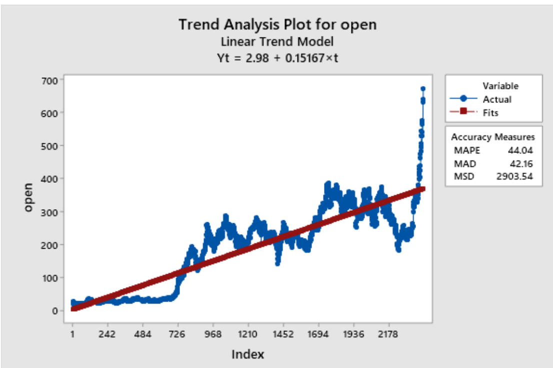


Figure 8: *Unstable non-stationary time series*

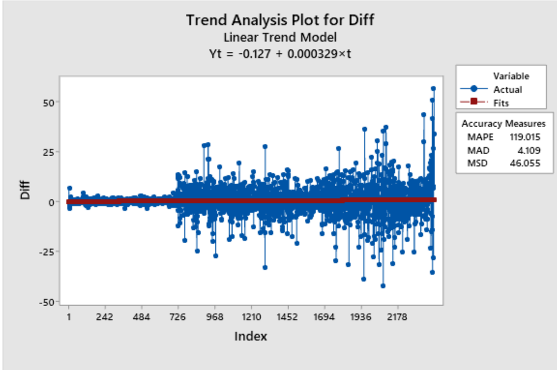


Figure 9: *Stable time series after taking the first difference*

We use partial autocorrelation (PAC) function and autocorrelation (AC) function to determine p and q of the ARIMA model, where the integer value of p can be obtained from the cut-off edges points of PAC graphs. Similarly, the value of q can be obtained using an AC graph. From Figures 10-13, we arrived to the conclusion that the suitable ARIMA model to be fitted has parameters $p = q = d = 1$, which is acceptable at a significance level of 0.05.

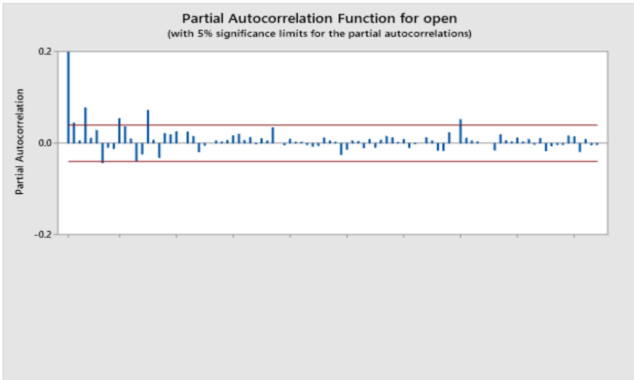


Figure 10: *Partial autocorrelation function of opening prices time series*

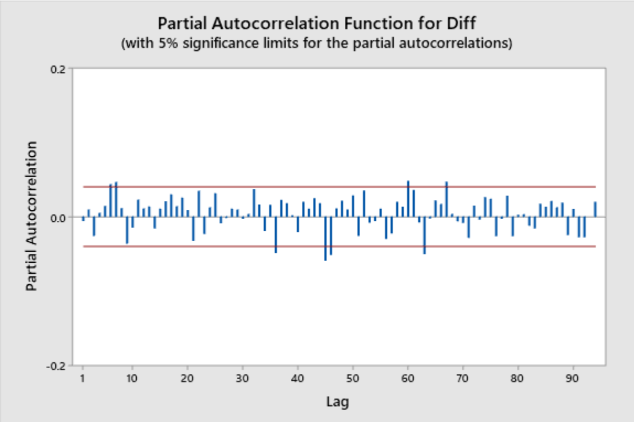


Figure 11: *Partial autocorrelation function of the first difference*

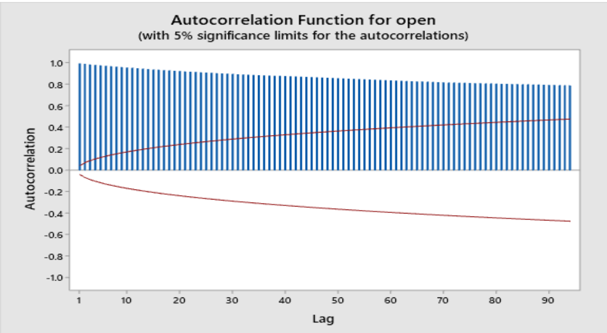


Figure 12: *Autocorrelation function of time series*

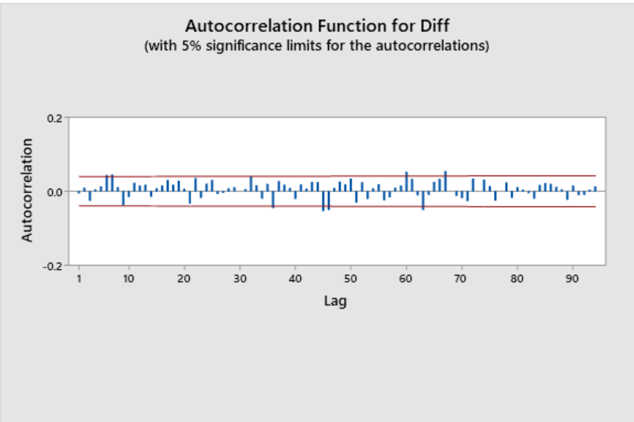


Figure 13: *Autocorrelation function of the first difference time series*

4. Experiments Setup and Results

In this section, we present the description of the employed LSTM and ARIMA models and the obtained results when using them to predict TESLA stock prices on a daily/weekly/monthly basis. We used MATLAB R2024b in the construction of our models.

4.1. Prediction error measure

The accuracy of the predictions made by a fitted model can be measured using the root-mean-square-error (RMSE). In our experiments, we will rely on this metric, which measures the error between the expected values of the fitted model and the actual values mathematically. The RMSE formula looks like a standard deviation formula, which is written as follows:

$$RMSE = \sqrt{\frac{\Sigma(y_i - \hat{y}_i)^2}{n}} \qquad (15)$$

where y_i are the actual values of the observations, $\hat{y}_i$ are the expected values or predicted values of the observations and n is the number of observations.

4.2. Forecasting Numerical Results

The available data records are divided into 50% training data and 50% validation data. First, we fed the LSTM and ARIMA models with normalized opening prices data and predicted the corresponding values. However, trying numerous combinations values for the ARIMA model parameters, we came to the conclusion of failing to construct an appropriate ARIMA model to fit the normalized data. On the other hand, the LSTM models with the parameters listed in Table 2 succeeded in predicting the normalized data series, providing an acceptable error level. To further enhance the model, we used the Mallat wavelet transform to eliminate the noise in the data. Five levels of detail or high frequency coefficients were used. The results for the testing data are shown in Figures 14-16. We tried predictions at different lags k, with k=1 referring to daily predictions, k=7 to weekly predictions and k =30 to monthly predictions. The RMSE values for the different lags are 0.119, 0.164, 0.3444, respectively.

Table 2: *First set of LSTMmodels parameters settings*

Parameter	Setting		
Lag	k= 1	k= 7	k= 30
Training algorithm	Adam's algorithm	Adam's algorithm	Adam's algorithm
Input layer neurons	1	1	1
Output layer neurons	1	1	1
Bi-LSTMlayer neurons	200	200	700
Max number of epochs	50	50	100
Mini batch size	4	4	240
Initial learning rate	0.001	0.001	0.001
Learning rate drop factor	0.1	0.1	0.1
Learning rate drop period	50	50	50

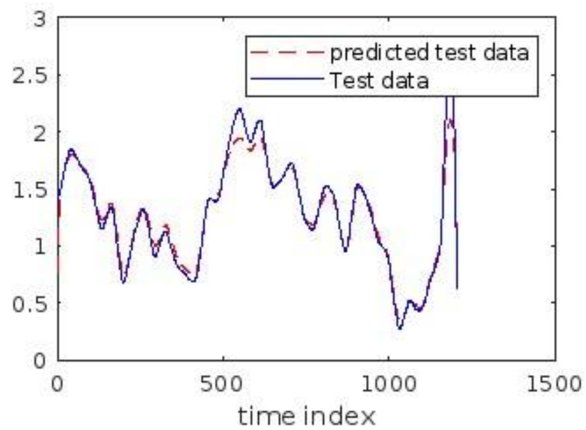


Figure 14: *Normalized daily opening prices time series forecasting results*

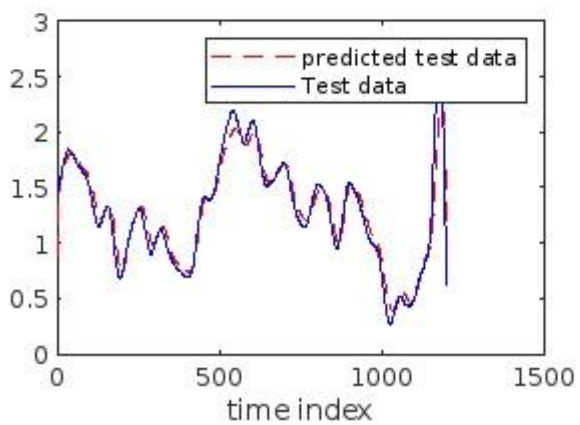


Figure 15: *Normalized weekly opening prices time series forecasting results*

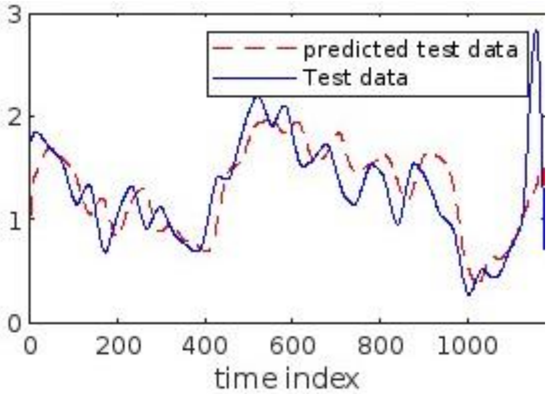


Figure 16: *Normalized monthly opening prices time series forecasting results*

It is apparent from Figure 14 and Figure 15 that the LSTM models can capture the general trend in the data and represent a good fit and thus can be confidently used for predicting future values. In Figure 16, though the prediction error is still quite small, yet there is a small shift between the actual observations and the predicted values.

In our second set of experiments, we tried feeding the LSTM models, whose parameters are listed in Table 3, and an ARIMA(1,1,1) model with the original data and predicting the residuals time series; that is, the time series resulting from the first differencing operation. To enhance the performance of the models in making predictions on weekly basis and monthly basis, we used data aggregation. By data aggregation, we mean summarizing the data of a week/month by one entry, which is taken to be the average of the observed values during a week/month. The obtained results are shown in Figures 17-19. The RMSE values are summarized in Table 4.

In Figure 17 and Figure 18, it clear that both the LSTM and ARIMA models succeeded in generating predictions following the same trend as that found in the original observations. However, there are clear fluctuations in the ARIMA model predictions for the monthly data series, as shown in Figure 19. From Table 4, it is clear that the LSTM models yield much smaller RMSE values.

Table 3: *Second set of LSTMmodels parameters settings*

Parameter	Setting		
Lag	Daily	Weekly	Monthly
Training algorithm	Adam's algorithm	Adam's algorithm	Adam's algorithm
Input layer neurons	1	1	1
Output layer neurons	1	1	1
Bi-LSTMlayer neurons	150	150	150
Max number of epochs	1000	1000	1000
Mini batch size	20	2	2
Initial learning rate	0.01	0.04	0.04
Learning rate drop factor	0.9	0.9	0.9
Learning rate drop period	50	100	100

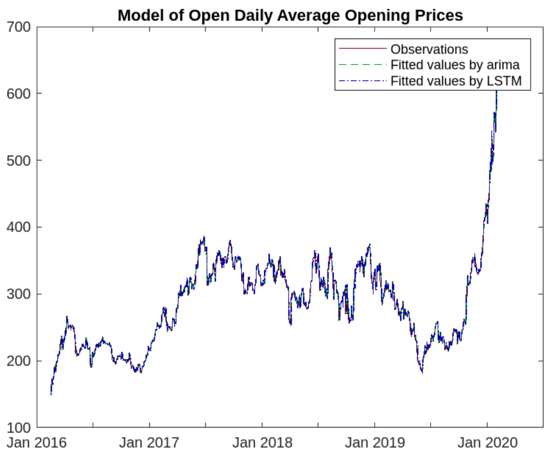


Figure 17: *Daily opening prices residuals time series forecasting results*

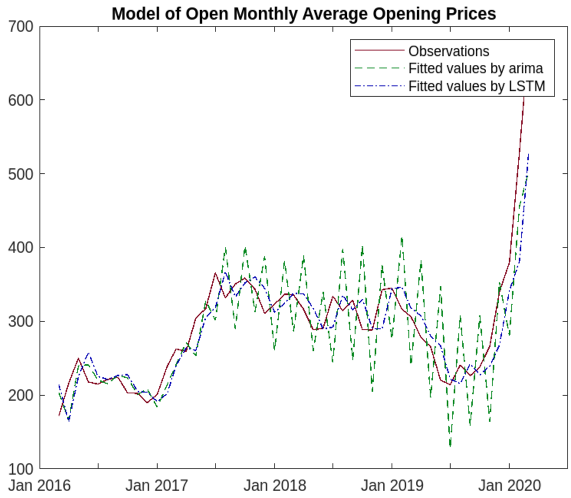


Figure 18: *Weekly opening prices residuals time series forecasting results*

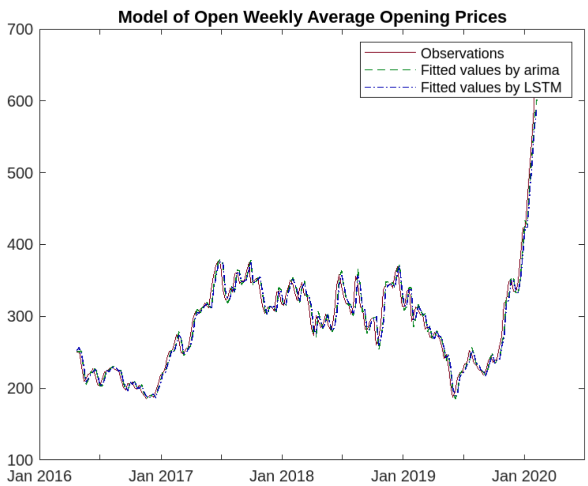


Figure 19: *Monthly opening prices residuals time series forecasting results*

Table 4: *LSTMversus ARIMA prediction results*

	Lag	1 day	week	month
RMSE	LSTM	0.6	1.3	3.2
	ARIMA	9	15.7	66.4

5. Conclusion

Stock price prediction is an important too for investors. In this paper, we compared the performance of LSTM models to traditional ARIMA models in predicting TESLA stock prices. LSTM models succeeded in producing more accurate predictions and capturing the complex patterns in the time series more closely. The enhancement in prediction capability has been verified both visually and quantitatively. Moreover, we considered simple data pre-processing techniques including normalization and denoising using the wavelet transform. Furthermore, it has been demonstrated that making monthly predictions is a more difficult task compared to making daily predictions. However, still the employed LSTM model was able to provide adequate future predictions.

Reference

[1] Khulood Albeladi, Bassam Zafar, and Ahmed Mueen. Time series forecasting using LSTMand ARIMA. International Journal of Advanced Computer Science and Applications, 14(1):313–320, 2023.

[2] Vaia I Kontopoulou, Athanasios D Panagopoulos, Ioannis Kakkos, and George K Matsopoulos. A review of arima vs. machine learning approaches for time series forecasting in data driven networks. Future Internet, 15(8):255, 2023

[3] Ayat Hamel and Baydaa Ismael Abdulwahhab. Time series forecasting using arima model. autoimmunity, 9:1, 2022.

[4] Arvydas Jadevicius and Simon Huston. Arima modelling of lithuanian house price index. International Journal of Housing Markets and Analysis, 8(1):135–147, 2015.

[5] Huizi Qian. Stock predicting based on LSTM and ARIMA. In 2022 2nd International Conference on Economic Development and Business Culture (ICEDBC 2022), pages 485–490. Atlantis Press, 2022.

[6] MK Ho, Hazlina Darman, and Sarah Musa. Stock price prediction using arima, neural network and LSTM models. In Journal of Physics: Conference Series, volume 1988, page 012041. IOP Publishing, 2021.

[7] Sun R.Q. Study of LSTMneural network-based price trend forecasting model for u.s. stock indexes. 2016.

[8] Jiajing Wang, Siyi Hong, Yuxin Dong, Zichao Li, and Jinxin Hu. Predicting stock market trends using LSTM networks: overcoming rnn limitations for improved financial forecasting. Journal of computer science and software applications, 4(3):1–7, 2024.

[9] Payal Soni, Yogya Tewari, and Deepa Krishnan. Machine learning approaches in stock price prediction: a systematic review. In Journal of Physics: Conference Series, volume 2161, page 012065. IOP Publishing, 2022.

[10] Akshay Sood. Interpreting Black-Box Models Using Hierarchical and Temporal Feature Abstractions. The University of Wisconsin-Madison, 2021.

[11] Guanze Shao. Prediction of stock prices based on the LSTM model. In 8th International Conference on Financial Innovation and Economic Development (ICFIED 2023), pages 377–387. Atlantis Press, 2023.

[12]Wei Jiang and Hans D Schotten. Recurrent neural networks with the long short-term memory for fading channel prediction. In 2020 IEEE 91st vehicular technology conference (VTC2020-Spring), pages 1–5. IEEE, 2020.

[13]Micah Mers, Zhongyu Yang, Yung-An Hsieh, and Yichang Tsai. Recurrent neural networks for pavement performance forecasting: review and model performance comparison. Transportation Research Record, 2677(1):610–624, 2023.

[14]Mohd Anul Haq, Ahsan Ahmed, Ilyas Khan, Jayadev Gyani, Abdullah Mohamed, El-Awady Attia, Pandian Mangan, and Dinagarapandi Pandi. Analysis of environmental factors using ai and ml methods. Scientific Reports, 12(1),13267, 2022.

[15]Lu Hou, Jinhua Zhu, James Kwok, Fei Gao, Tao Qin, and Tie-yan Liu. Normalization helps training of

quantized lstm. Advances in Neural Information Processing Systems, 32, 2019.

[16]Adam Hayes. Autoregressive integrated moving average (arima) prediction model. Investopedia. https://www.investopedia.com/terms/a/autoregressive-integratedmoving-average-arima. asp (accessed, 2023.

[17]Mubarak Daha Isa. Understanding time series forecasting with arima. https://medium.com/@mubarakdaha/

understanding-time-series-forecasting-with-arima-59cd7140d6c3. Accessed: 2024.07-09–

[18]Jamal Fattah, Latifa Ezzine, Zineb Aman, Haj El Moussami, and Abdeslam Lachhab. Forecasting of demand using arima model. International Journal of Engineering Business Management, 10:1847979018808673, 2018.

[19]A. Wan. Achapter 5: Box-jenkins (arima) forecasting. Available at: https://personal.cityu.edu.hk/msawan/teaching/ms6215/MS6215Ch5a.pdf, 2022. Accessed: 2024–11-06.

[20] Arief Agoestanto, Siti Rosidah, et al. Arima method with the software minitab and reviews to forecast inflation in semarang indonesia. Journal of Theoretical & Applied Information Technology, 94(1), 2016.

[21] Georgina M Tinungki. The analysis of partial autocorrelation function in predicting maximum wind speed. In IOP Conference Series: Earth and Environmental Science, volume 235, page 012097. IOP Publishing, 2019.

[22] Sana Prasanth Shakti, Mohan Kamal Hassan, Yang Zhenning, Ronnie D Caytiles, and NCSN Iyenger. Annual automobile sales prediction using arima model. Int. J. Hybrid Inf. Technol, 10(6):13–22, 2017.

[23] Philip Sedgwick. Pearson's correlation coefficient. Bmj, 345, 2012.

[24] Qi Tang, Ruchen Shi, Tongmei Fan, Yidan Ma, and Jingyan Huang. Prediction of financial time series based on LSTM using wavelet transform and singular spectrum analysis. Mathematical Problems in Engineering, 2021(1):9942410, 2021.