Comparing LSTM Models for Stock Market Prediction: A Case Study with Apple's Historical Prices

Ha Minh Tan, Le Gia Minh, Tran Cao Minh, Tran Thi Be Quyen, and Kien Cao-Van(✉)

Faculty of Information Technology, Nguyen Tat Thanh University, Ho Chi Minh City, Vietnam
cvkien@ntt.edu.vn

Abstract. Stock market prediction holds significant importance in the world of finance, captivating the attention of both investors and financial researchers. The integration of artificial intelligence and advancements in computational power has led to substantial improvements in predicting stock prices, surpassing the effectiveness of traditional programmed prediction methods. In this paper, we explore three distinct and innovative methods for stock price prediction: Long Short-Term Memory (LSTM), LSTM combined with Simple Moving Average (LSTM-SMA), and LSTM combined with Exponential Moving Average (LSTM-EMA). Our analysis is conducted using a comprehensive historical dataset of Apple's stock prices, and the performance of each model is rigorously evaluated using critical metrics, including Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R2 score. Additionally, the training time for each model is taken into account. The results show that all three models LSTM, LSTM-SMA, and LSTM-EMA give good prediction results for Apple's stock price, in which the LSTM model gives the best prediction results for the 21-day cluster. However, in terms of computational time, the LSTM-SMA model and LSTM-EMA model are more efficient than the LSTM model. These findings highlight the potential of integrating advanced techniques to achieve more accurate and efficient stock price predictions.

Keywords: Long Short-Term Memory · Simple Moving Average · Exponential Moving Average · Neural Network · Prediction · Stock Price

1 Introduction

Stock prediction is a significant challenge in the field of finance. Investors and risk managers alike desire the ability to accurately forecast the trends and fluctuations of stock prices in order to make intelligent and effective investment decisions. In the digital age, the development of technology has generated a vast amount of stock market data and advanced analytical tools, thereby creating

P. Cong Vinh and H. Mahfooz Ul Haque (Eds.): ICTCC 2023, LNICST 586, pp. 173–185, 2024.
https://doi.org/10.1007/978-3-031-59462-5_12

opportunities to apply database-driven prediction methods to provide forecasts for the stock market.

In recent years, many researchers have explored this topic from various perspectives. They have employed different approaches such as statistical methods [3,7,9,15], supervised learning methods, with Long Short-Term Memory (LSTM) being a typical example [4,8,14,16], and unsupervised learning [13]. Statistical methods require several factors to achieve good results, such as data reliability and adherence to standards and assumptions. On the other hand, neural networks, especially LSTM, have the capability to learn complex patterns and process time series data. LSTM is able to store long-term information and capture relationships in past data, enabling it to learn and predict trends, fluctuations, and complex patterns in stock market time series data.

When trained on historical stock market data, neural networks have the ability to outperform traditional methods such as simple statistical models in terms of prediction. Through continuous learning and updating during the training process, neural networks can enhance prediction capabilities and provide more accurate results. However, the effectiveness of both statistical methods and neural networks in stock market prediction can be influenced by various factors, including sample size, data quality, and training methods.

We conducted experiments on a historical dataset of Apple Inc. stock obtained from the website https://finance.yahoo.com/, spanning from January 2, 2019, to June 29, 2023, comprising 1,131 data points. This dataset provides detailed information on the price fluctuations of Apple Inc. stock over the past four years, enabling investors and financial analysts to analyze trends and forecast prices for future periods.

In this paper, we focus on researching and comparing the performance of three stock prediction methods: LSTM, LSTM combined with SMA technique, and LSTM combined with EMA technique. Our objective is to evaluate and compare the predictive capabilities of these methods, while determining which method is the most effective in predicting stock prices.

The remainder of the article is structured as follows: Sect. 2 describes the methodology of the applied techniques. Section 3 presents the experimental evaluation of the methods' results, and Sect. 4 concludes the paper.

2 Methodology

2.1 Description of Data

The historical stock price data for Apple was downloaded from the website https://finance.yahoo.com/ from January 02, 2019, to June 29, 2023, comprising 1,131 rows of data. The dataset contains information such as Date, Open, High, Low, Close, Adj Close, and Volume, which are used for analyzing and tracking stock prices and other assets in the financial market. Some of the stocks shown in Table 1. Moreover, the data has been checked and does not contain any empty, null, or NaN values.

Table 1. Dataset

Date	Open	High	Low	Close	Adj Close	Volume
02/01/2019	38.72	39.71	38.56	39.48	37.99	148,158,800
03/01/2019	35.99	36.43	35.50	35.55	34.21	365,248,800
...	...	...	...	...	...	...
28/06/2023	187.93	189.90	187.60	189.25	189.25	51,216,800
29/06/2023	189.08	190.07	188.94	189.59	189.59	46,347,300

In the world of trading, the final trading price signifies the latest value at which a stock is exchanged during regular trading hours. This closing price serves as a pivotal reference point for investors to monitor the stock's performance over an extended period. The objective of this study is to forecast the closing price of Apple's stock market. Figure 1 illustrates the closing price of Apple's stock market spanning from January 02, 2019, to June 29, 2023.

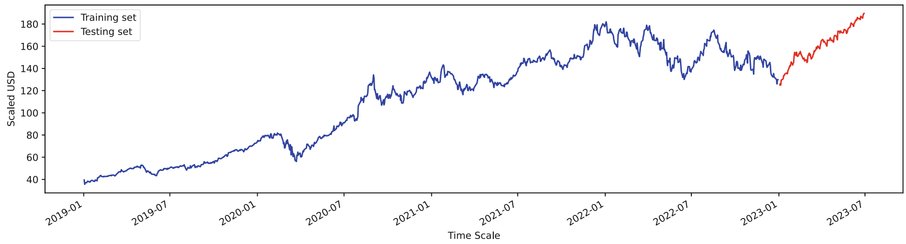

Fig. 1. Closing price of the Apple stock market.

A set of six additional variables has been developed to enhance the accuracy of predicting stock closing prices, as implemented in the research [17]. These newly introduced variables were utilized in the training process of the model. Here are the details of these fresh variables:

- Stock High minus Low price (H-L)
- Stock Close minus Open price (O-C)
- Stock price's seven days' moving average (7 DAYS MA)
- Stock price's fourteen days' moving average (14 DAYS MA)
- Stock price's twenty one days' moving average (21 DAYS MA)
- Stock price's standard deviation for the past seven days (7 DAYS STD DEV)

2.2 Min-max Scaling

Min-max scaling is a widely recognized technique used for normalization. It involves mapping variables to a specific range, typically [0, 1], where the minimum and maximum values of a variable are set to 0 and 1, respectively. The primary objective is to ensure that variables are measured on a consistent scale, enabling each variable to contribute proportionally during model fitting. This normalization method helps align the scales of different variables, facilitating fair comparisons and accurate analysis. The mathematical formula is,

$$x_{scale} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

where x_{scale} and x are scaled and the original input respectively. Similarly, $x_{\max}$ and $x_{\min}$ are the maximum and minimum value of each feature, respectively.

2.3 Moving Average

Moving Average (MA) is an essential technique for analyzing time series data as it effectively smooths the input data by averaging values over a specified time period, thereby revealing their underlying price trend. Subsequently, a new time series is generated based on these average values. There are several approaches to calculating MA, with the two most prevalent methods being the Simple Moving Average (SMA) and the Exponential Moving Average (EMA), both of which have been extensively applied [2,5] in this field.

SMA is a method to predict the value of the next data point based on the mean of the previous "n" data points. It involves calculating the average of the past "n" data points, denoted as P_1 to P_n, and using this average as the predicted value for the next data point. Namely,

$$SMA = \frac{P_1 + P_2 + \cdots + P_n}{n}. \tag{2}$$

The choice of "n" significantly impacts the precision of the prediction. A higher "n" means considering a more extended period in the past to compute the present value. For instance, with $n = 2$, the average of the past two days' stock prices is taken, while with $n = 50$, fifty days' worth of stock prices are considered. Naturally, using more data points provides more information about the stock's trends, leading to better predictions. However, using an excessively large "n" can also destabilize the model, as it smoothes out finer fluctuations, and looking too far back in time, such as the past 300 days, may not be optimal for prediction accuracy.

On the other hand, EMA is a method for calculating a moving average by giving greater weight to recent values rather than assigning equal weights to the entire data.

$$EMA = k * P_t + (1 - k) * EMA_{(t-1)}, \tag{3}$$

where P_t represents the price at time t, and k denotes the weight assigned to that particular data point. $EMA_{(t-1)}$ corresponds to the value computed from the previous $t-1$ data points. The number of time points in the EMA is denoted by N, and the weighting factor is calculated as $k = 2/(N + 1)$.

The EMA has an edge over the SMA as it exhibits greater responsiveness to price fluctuations. This characteristic renders it particularly valuable for short-term trading strategies.

2.4 Long Short Term Memory

The Recurrent Neural Network (RNN) was initially designed to handle sequential or time-related data effectively. However, when propagated through multiple time steps, the gradient may experience either an exponential decrease or a significant increase. The issue of diminishing gradients can result in ineffective weight updates and the loss of distant information, commonly referred to as vanishing gradients [10].

In 1997, Hochreiter and Schmidhuber introduced the LSTM model [11], a significant advancement in the realm of RNNs, specifically designed to address challenges related to vanishing gradients and long-term dependencies. While an RNN model comprises a sequence of recurrent neural network modules, the standard RNN models often employ a simple structure, typically using a *tanh* layer as the repeating module. In contrast, LSTM models also follow a sequential structure akin to RNNs but with slightly different configurations for the repeating modules. As shown in Fig. 2, LSTM incorporates four key components (cell, forget gate, input gate, output gate) that interact in a specialized manner to enhance its capabilities.

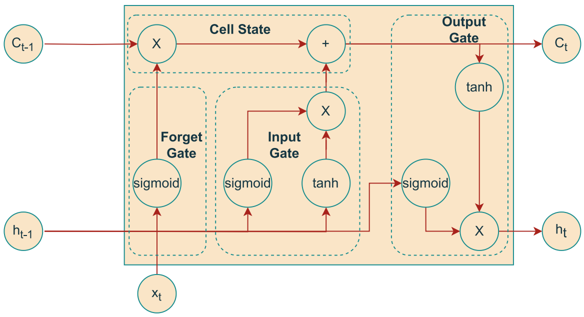

Fig. 2. Architecture of a cell Long Short-Term Memory

The sequence of steps in the LSTM model is as follows:

– The first step in LSTM determines which information is allowed to pass through the cell state. It is controlled by a sigmoid function in a layer called the forget gate. It takes inputs h_{t-1} and x_t and outputs a value between 0 and 1 for each value in the cell state C_{t-1}. A value of 1 indicates "keep all information", and 0 indicates "discard all of it".

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right), \tag{4}$$

where σ denotes the sigmoid function, W_f denotes the weight of forget gate, b_f denotes the bias of forget gate, x_t is input at time t, and h_{t-1} is the output of the hidden layer at time $t-1$.

- Next, we decide what information to store in the cell state. This step consists of two parts. The first part is a hidden layer of the sigmoid function called the input gate layer, which decides what values to update. Then, a *tanh* function generates a vector of a new state value C_t that can be added to the cell state. The results of these two layers are combined to form an update for the cell state.

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{5}$$

$$\tilde{C}_t = \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right) \tag{6}$$

where σ denotes the sigmoid function, W_f denotes the weight of forget gate, b_i denotes the bias of input gate, x_t is input at time t, h_{t-1} is the output of the hidden layer at time $t-1$, W_c is the weight of cell, and b_c is the bias of cell.

- At this point, we update the old cell state C_{t-1} to a new state C_t, which has been decided in the previous steps, and in this step, we simply carry out that decision.
- We multiply the old cell state by f_t, which corresponds to forgetting the parts that we decided to forget early on. The candidate element $i_t * \tilde{C}_t$ is a newly computed value corresponding to how much should be added to each cell state value.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t. \tag{7}$$

- Finally, we need to decide how much output to return. The output at this stage will be based on the cell state but will be a filtered version. First, we run it through a sigmoid layer where we decide which parts of the cell state will be in the output. Then, the cell state is passed through a tanh function (to squash the values between -1 and 1) and multiplied by the output of a sigmoid gate, thus outputting only the parts we decided.

$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{8}$$

$$h_t = o_t * \tanh\left(C_t\right), \tag{9}$$

where σ denotes the sigmoid function, W_o denotes the weight of output gate, and b_o denotes the bias of output gate.

Based on its operating mechanism, LSTM is considered superior to RNN. LSTM can access information from a larger set of data, making it highly suitable for long-term time series forecasting. The LSTM model has found applications in various fields due to its ability to capture long-term dependencies in sequential data. Some notable applications of LSTM include: Time Series Forecasting [1], Speech Recognition [6], Natural Language Processing [12], etc.

3 Experimental Results

During the model construction phase, we will establish three distinct models, each corresponding to one of the following methods: LSTM, LSTM combined with SMA, and LSTM combined with EMA. More precisely, the LSTM model will be trained using data organized into clusters of 7-day, 14-day, and 21-day intervals. For the LSTM with the SMA method, we will first apply a simple moving average to smooth the data within 7-day, 14-day, and 21-day clusters before feeding it into the LSTM model. On the other hand, the LSTM with EMA approach will involve smoothing the data within 7-day, 14-day, and 21-day clusters using exponential moving averages before inputting it into the LSTM model. These models will exhibit diverse performances and outcomes when applied. Through careful observation and evaluation, we aim to identify the optimal model for predicting the closing price of Apple's stock.

Parameter Configuration: The LSTM network will have a consistent setup across all three methods. The input layer will contain 128 units, with "return_sequences" set to True and the "input_shape" taking values from the training set. A Dropout rate of 0.2 will be applied. Intermediate layers 2, 3, and 4 will share the same configuration, each with 128 units and a Dropout rate of 0.2. The output layer of the LSTM will be a fully connected layer (Dense), resulting in a model size of 461,441 parameters.

Data Splitting: The training set will encompass closing price data from January 02, 2019, to December 30, 2022, while the test set will cover data from January 03, 2023, to June 29, 2023. To implement the early stopping method, Early Stopping callbacks with "monitor" set to 'loss' and a "patience" of 30 will be used. Additionally, the validation set will consist of 10% of the data for all methods.

The hardware setup for the project comprises a Ryzen 3 2200G processor, a 256 GB SSD, 16 GB RAM, and an RX570 graphics card. For coding and experimentation, we utilize the Google Colab v2.15.0 platform along with the Python programming language. Additionally, we incorporate essential libraries such as TensorFlow v2.6.0, Numpy v1.22.2, and Scikit-learn v1.0.2 to facilitate the construction and experimentation processes.

3.1 Evaluation Criteria

To assess the model's performance, we will use the following metrics: MAPE (Mean Absolute Percentage Error), MAE (Mean Absolute Error), RMSE (Root Mean Square Error), and R2 score. These metrics are commonly used in forecasting and statistics to evaluate the accuracy of prediction models. They measure the level of deviation between predicted values and actual values. With n representing the number of samples in the dataset, Y_t denoting the actual value of the t-th sample, and $\widehat{Y_t}$ representing the predicted value of sample t.

– **MAPE**, which measures prediction accuracy, calculates the ratio of the sum of absolute errors of predictions to the sum of the actual values. A lower MAPE value indicates a more accurate model assessment.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{Y_t - \widehat{Y}_t}{Y_t}\right| \tag{10}$$

– **MAE** quantifies the average deviation between predicted values and actual values, using absolute values for computation. A lower MAE value signifies a better model accuracy.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|Y_t - \widehat{Y}_t\right| \tag{11}$$

– **RMSE** assesses the accuracy of a prediction model in comparison to real data. A lower RMSE value indicates a higher level of model accuracy.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(Y_t - \widehat{Y}_t\right)^2} \tag{12}$$

– **R2 score** score evaluates how well a model fits the data. It is computed by taking the sum of the squares of the differences between the predicted and actual values, dividing it by the sum of the squares of the differences between the mean of the actual values and the actual values, and subtracting this result from 1. A higher R2 score indicates a better fit of the model to the data, serving as a metric for model accuracy assessment.

$$R2 = 1 - \frac{\sum_{t=1}^{n}\left(Y_t - \widehat{Y}_t\right)^2}{\sum_{t=1}^{n}\left(Y_t - \bar{Y}\right)^2}. \tag{13}$$

3.2 Analysis of Results

In this study, we conducted stock price predictions of Apple using three different methods: LSTM, LSTM combined with the SMA technique, and LSTM combined with the EMA technique. Specifically, we used data clusters of 7 days, 14 days, and 21 days to input into the prediction models and evaluated the results using several performance evaluation metrics such as MAPE, MAE, RMSE, and R2.

Subsequently, we compared the experimental results to gain a better understanding of the performance of each method. Additionally, testing with different data clusters allowed us to identify the most suitable data cluster to achieve the highest accuracy. The general results of the three methods can be found in Table 2.

Table 2. Experimental Results for Stock Price Prediction of Apple using Different Methods and Data Clusters.

Method	Cluster	Value				Epochs	Training time (seconds)
		MAPE	MAE	RMSE	R2		
LSTM	**7-days**	0.0237	3.8823	4.4520	0.9167	236	529.1066
	14-days	0.0171	2.7637	3.3151	0.9550	256	977.6710
	21-days	0.0140	2.2685	2.7940	0.9672	277	1637.5458
LSTM-SMA	**7-days**	0.0170	2.8032	3.5610	0.9487	206	150.6537
	14-days	0.0156	2.5766	3.3633	0.9546	258	192.8566
	21-days	0.0169	2.7848	3.6121	0.9480	205	142.0111
LSTM-EMA	**7-days**	0.0188	3.1369	4.0096	0.9357	149	114.1174
	14-days	0.0170	2.8194	3.6231	0.9463	166	107.8286
	21-days	0.0162	2.6887	3.5321	0.9504	182	130.1446

Below is a comment on the prediction results of three methods LSTM, LSTM combined with SMA and LSTM combined with EMA based on evaluation indicators and training time:

Prediction Method Using the LSTM Model: The best results are achieved when using a data cluster of 21 days, with MAPE = 0.0140, MAE = 2.2685, RMSE = 2.7940, R2 = 0.9672. This indicates that the LSTM model has the ability to accurately predict with a large data cluster.

Training time increases with the data cluster, ranging from 529 s for a 7-day cluster to 1637 s for a 21-day cluster. The LSTM model requires longer training time when using larger data clusters (Figs. 3 and 4).

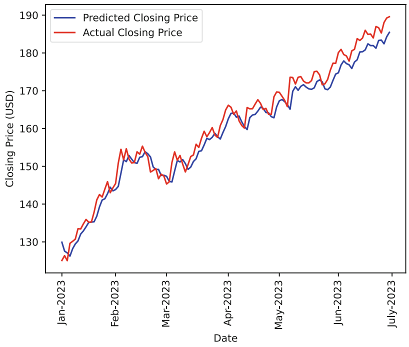

Fig. 3. Closing price Apple's stock is predicted by pure LSTM in 7-day clusters

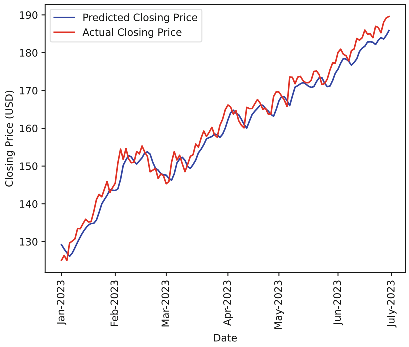

Fig. 4. Closing price Apple's stock is predicted by pure LSTM in 14-day clusters

Prediction Method Using LSTM Combined with SMA Technique: The results show that this model performs better than the pure LSTM method with smaller data clusters (Figs. 5, 6, 7 and 8).

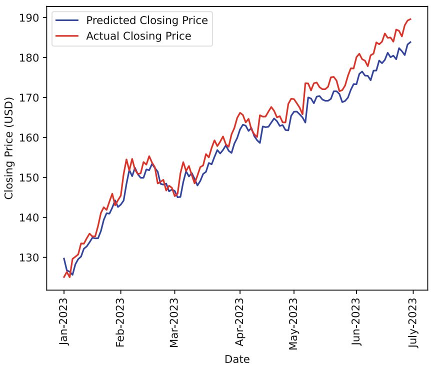

Fig. 5. Closing price Apple's stock is predicted by pure LSTM in 21-day clusters

The best results are achieved when using a data cluster of 14 days, with MAPE = 0.0156, MAE = 2.5766, RMSE = 3.3633, R2 = 0.9546.

The training time of the model with SMA technique is significantly faster compared to the pure LSTM model, ranging from 142 s (for a 21-day cluster) to 192 s (for a 14-day cluster).

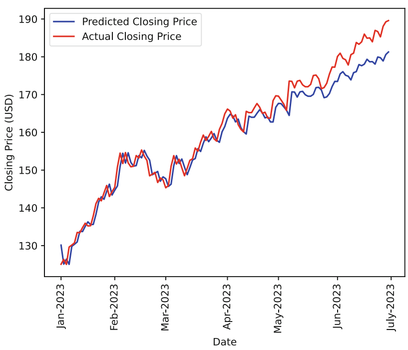

Fig. 6. Closing price Apple's stock is predicted by LSTM with SMA in 7-day clusters

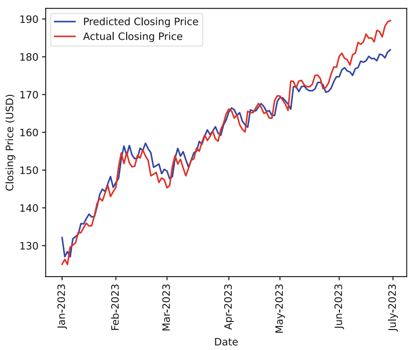

Fig. 7. Closing price Apple's stock is predicted by LSTM with SMA in 14-day clusters

Prediction Method Using LSTM Combined with EMA Technique: The results also show improvements compared to the LSTM method with smaller data clusters (Figs. 9, 10 and 11).

The best results are obtained when using a data cluster of 21 days, with MAPE = 0.0162, MAE = 2.6887, RMSE = 3.5321, R2 = 0.9504.

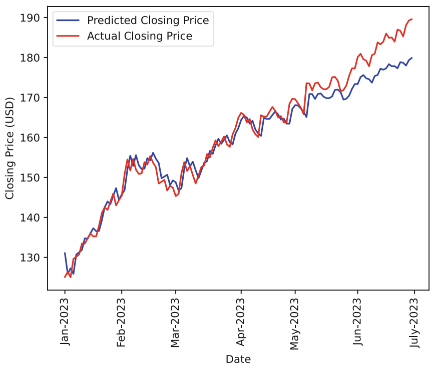

Fig. 8. Closing price Apple's stock is predicted by LSTM with SMA in 21-day clusters

The LSTM model combined with EMA technique is also faster than the pure LSTM model, with training times ranging from about 107 s (for a 14-day cluster) to 130 s (for a 21-day cluster).

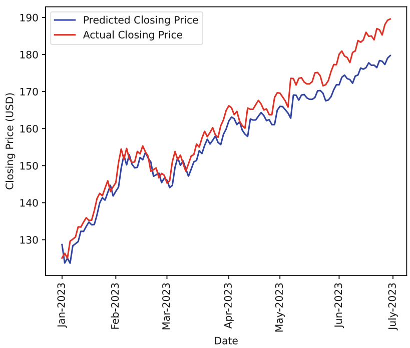

Fig. 9. Closing price Apple's stock is predicted by LSTM with EMA in 7-day clusters

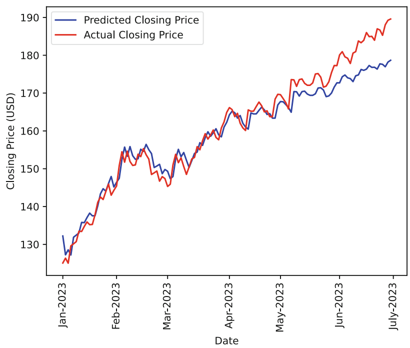

Fig. 10. Closing price Apple's stock is predicted by LSTM with EMA in 14-day clusters

In summary, the findings indicate that all three models, namely LSTM, LSTM-SMA, and LSTM-EMA, yield favorable predictions for Apple's stock price. Among them, the LSTM model stands out with the most accurate predictions within the 21-day cluster. Nevertheless, when considering computational time, both the LSTM-SMA and LSTM-EMA models outperform the LSTM model. Nonetheless, the choice of suitable techniques and data clusters for achieving precise predictions still hinges on the unique objectives and requirements of the model.

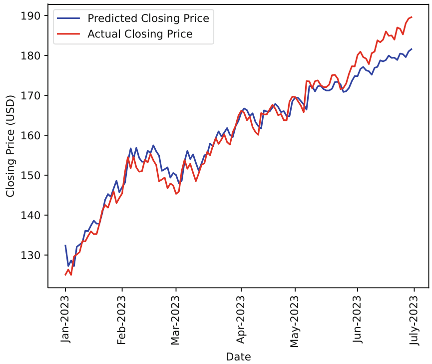

Fig. 11. Closing price Apple's stock is predicted by LSTM with EMA in 21-day clusters

4 Conclusions and Future Work

In this study, we have demonstrated the feasibility and performance of applying the data smoothing methods, SMA and EMA, to the Apple Stock dataset. Our results show promising values in terms of accuracy, time, training speed, and data processing when using SMA and EMA instead of directly inputting data into the LSTM model. For future research, we aim to enhance the capability of the LSTM with the EMA method to achieve even higher performance and potentially surpass the traditional LSTM's capabilities. Additionally, we will apply this approach to a broader range of datasets to provide more comprehensive and rigorous evaluations.

Acknowledgments. This research is funded by Nguyen Tat Thanh University, Ho Chi Minh city, Vietnam.

References

1. AL-Alimi, D., et al.: Tlia: time-series forecasting model using long short-term memory integrated with artificial neural networks for volatile energy markets. Appl. Energy **343**, 121230 (2023). https://doi.org/10.1016/j.apenergy.2023.121230
2. Altarawneh, G.A., Hassanat, A.B., Tarawneh, A.S., Abadleh, A., Alrashidi, M., Alghamdi, M.: Stock price forecasting for Jordan insurance companies amid the Covid-19 pandemic utilizing off-the-shelf technical analysis methods. Economies **10**(2), 43 (2022)
3. Ariyo, A.A., Adewumi, A.O., Ayo, C.K.: Stock price prediction using the Arima model. In: 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp. 106–112. IEEE (2014)
4. Bhandari, H.N., Rimal, B., Pokhrel, N.R., Rimal, R., Dahal, K.R., Khatri, R.K.: Predicting stock market index using LSTM. Mach. Learn. App. **9**, 100320 (2022)
5. Biswas, M., Shome, A., Islam, M.A., Nova, A.J., Ahmed, S.: Predicting stock market price: a logical strategy using deep learning. In: 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), pp. 218–223 (2021). https://doi.org/10.1109/ISCAIE51753.2021.9431817

6. Chien, J.T., Misbullah, A.: Deep long short-term memory networks for speech recognition. In: 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP), pp. 1–5 (2016). https://doi.org/10.1109/ISCSLP.2016.7918375
7. Fattah, J., Ezzine, L., Aman, Z., El Moussami, H., Lachhab, A.: Forecasting of demand using Arima model. Int. J. Eng. Bus. Manage. **10**, 1847979018808673 (2018)
8. Gülmez, B.: Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm. Expert Syst. Appl. **227**, 120346 (2023)
9. Haining, R.: The moving average model for spatial interaction. Trans. Inst. Br. Geograph. **3**, 202–225 (1978)
10. Hochreiter, S.: The vanishing gradient problem during learning recurrent neural nets and problem solutions. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **06**(02), 107–116 (1998). https://doi.org/10.1142/S0218488598000094
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
12. Kasthuri, E., Balaji, S.: Natural language processing and deep learning chatbot using long short term memory algorithm. Mater. Today Proc. **81**, 690–693 (2023)
13. Leangarun, T., Tangamchit, P., Thajchayapong, S.: Stock price manipulation detection using deep unsupervised learning: the case of Thailand. IEEE Access **9**, 106824–106838 (2021)
14. Md, A.Q., Kapoor, S., A.V., C.J., Sivaraman, A.K., Tee, K.F., H., S., N., J.: Novel optimization approach for stock price forecasting using multi-layered sequential LSTM. Appl. Soft Comput. **134**, 109830 (2023). https://doi.org/10.1016/j.asoc.2022.109830, https://www.sciencedirect.com/science/article/pii/S1568494622008791
15. Mondal, P., Shit, L., Goswami, S.: Study of effectiveness of time series modeling (Arima) in forecasting stock prices. Int. J. Comput. Sci. Eng. App. **4**(2), 13 (2014)
16. Tan, H.M., Wang, J.C.: Single channel speech separation using enhanced learning on embedding features. In: IEEE 10th Global Conference on Consumer Electronics (GCCE), pp. 430–431. IEEE (2021)
17. Vijh, M., Chandola, D., Tikkiwal, V., Kumar, A.: Stock closing price prediction using machine learning techniques. Proc. Comput. Sci. **167**, 599–606 (2020). https://doi.org/10.1016/j.procs.2020.03.326