# S&P 500 Stock Price Prediction using LSTM.

Zemeng Chen *

School of Science, Rensselear Polytechnic Institute, Troy, US

* Corresponding Author Email: chenz29@rpi.edu

**Abstract.** This paper conducts a comprehensive investigation into the effectiveness of LSTM neural networks in the realm of stock price prediction. By leveraging a combination of historical price data, technical indicators, and market sentiment features, the LSTM model adeptly captures both short and long-term patterns inherent in financial data. Through rigorous experimentation and analysis using real-world stock market data, the study illuminates the model's potential in unraveling complex relationships that drive market dynamics. Despite the challenges posed by the inherent volatility of financial markets, LSTM-based models exhibit promise in enhancing decision-making within trading contexts. It is important, however, to exercise caution when applying these models in highly unpredictable markets. The paper underscores the need for a balanced and informed approach. In summary, LSTM neural networks emerge as a valuable and versatile tool for refining stock price prediction methodologies. Their capacity to decipher intricate patterns positions them as a significant asset in the pursuit of more accurate and insightful financial predictions.

**Keywords:** LSTM neural networks; prediction accuracy; real-world data.

## 1. Introduction

Stock prices are influenced by various factors, primarily the business performance of a company and markets overseas. Investors use financial statements, stock market indexes, and news from diverse sources to make informed judgments [1]. However, determining the best option is challenging for investors based on this information since standard finance assumes investors are rational and unaffected by risk-return trade-offs and value-adding strategies. According to the efficient market theory, investors should consider all relevant information and maintain objectivity when evaluating securities and selecting profitable equities. It is important to recognize that the human brain has limitations in processing large amounts of information. Additionally, psychologists have revealed that individuals do not always behave in a strictly rational manner as commonly expected by economists. A recent study conducted by Hui-Chu Shu discovered a positive correlation between investor sentiment and security prices, particularly in equities and bill prices. This highlights the significant role of investor sentiment in determining equilibrium asset prices and returns [2].

The stock market index represents the value changes in the stock market and is compiled by either a stock exchange or financial service institution. Investors face market price risks due to stock price volatility. Although investors can comprehend changes in one stock price easily, understanding the changes in multiple stocks one by one is tedious and complicated. In order to adapt to this situation and meet the needs of investors, financial service institutions utilize their business expertise and market familiarity to create publicly released stock price indices, which serve as indicators of market price changes. This allows investors to evaluate the effectiveness of their investments and make predictions about the stock market's trends. At the same time, the press, company executives, and even political leaders also use this as a reference index to observe and predict the social, political, and economic development situation.

Time series modeling of financial data with increasing variation over time is gaining popularity. Since Engle and Bollerslev's development of the first Autoregressive Conditional Heteroscedasticity (ARCH) and Generalized ARCH (GARCH) models, these parametric models for financial asset volatility have undergone significant improvement [3]. These heteroscedastic time series models are particularly useful for simulating highly volatile financial market data. All technical term abbreviations are explained the first time they are used. Although time series data typically assumes a linear correlation structure, many financial observations display a non-linear dependency structure.

As a result, linear approximation models may not be suitable for those complex situations, and GARCH type models may not capture such nonlinear patterns. Thus, using linear approximation models for these complicated situations may not be fitting and may not show the nonlinear patterns that GARCH type models cannot capture. Compared to linear models, which offer inadequate forecasts, nonparametric models estimated through various techniques, including artificial intelligence (AI), can fit a dataset significantly better [4].

As stock price data is a time series, a model must be able to handle it. Schumaker et al. [5] and Ding et al. assumed the effect of prior events had a set duration while using SVM and deep neural networks [6]. While numerous events have a similar duration of impact, certain terms, such as "financial crisis," can have lasting effects. To reflect these impacts, a model must incorporate time series data. Additionally, a random forest model can be utilized to predict the future direction of stock market prices. Khaidemb et al. applied the random forest method to predict the future stock prices of AAPL, MSFT, and SAMSUNG, achieving a notable accuracy of 85%-95% [7]. Besides, ANN and RNN are also common methods for prediction in the financial market. Ticknor developed a Bayesian regularized artificial neural network that reduces the risk of overtraining and overfitting while increasing prediction accuracy and network generalization for forecasting stock market prices [8]. To assess the model's effectiveness, experiments were conducted on the stocks of Goldman Sachs Group Inc. and Microsoft Corp. The results indicate that the proposed model can replicate the performance of more intricate models without the need for data pre-processing, testing for seasonality, or analyzing cycles.

## 2. Method

### 2.1. Data Source and Description

The data is S&P 500 stock closing prices selected from the recent 5 years collected by Yahoo Finance. The data is automatically converted to time series object in R. In order to ease the following forecast work, the data is normalized, meaning that each number within the dataset is divided by the maximum value of this set. It has increasing trend, no seasonality, significantly non-stationary by analyzing Figure 1. Therefore, we take the log to smooth its volatility and first-order difference to stabilize it. Finally, the data after logarithm processing passed the autocorrelation and KPSS test.

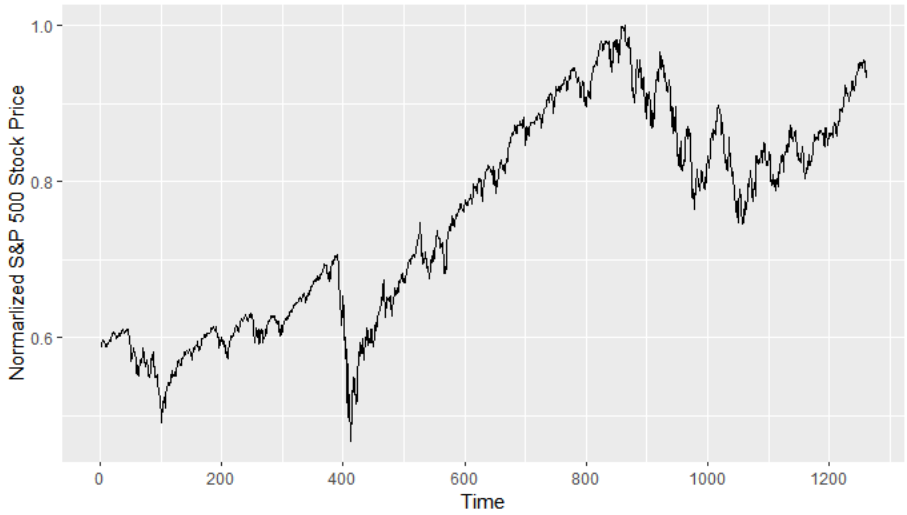


**Fig. 1** Normalized S&P 500 stock closing price versus time

### 2.2. Variable Selection and Description

There are many variables that need to be taken into account while analyzing whether a type of stock is profitable. In this essay the main focus is on the closing price of S&P 500. The maximum

price from the 5-year interval was $4796.56 per share, while the minimum price was $2237.4. The average stock price after calculation is $3624.02; moreover, the standard deviation is 650.46, which is large enough to claim that the price of stock is highly unstable [9]. Therefore, it may cause inaccuracy while making forecasts.

Consistent estimation (or consensus estimation) is the standard for evaluating estimators in large samples. When the sample size is not large, people tend to use small sample-based evaluation criteria. In this case, variance is used for unbiased estimation and mean square error is used for biased estimation.

Generally, when the sample size is fixed, the criterion used to evaluate the quality of a point estimation is always a function of the distance between the point estimation and the true value of the parameter. The most commonly used function is the square of the distance. Due to the randomness of the estimation, the expectation of this function can be obtained, which is the mean square error given by the following equation:

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{E}(\hat{\theta} - \theta)^2 \tag{1}$$

### 2.3. Methodology

Long short-term memory (LSTM) is a variant of RNN. Its core concept is cell state and "gate" structure [10]. Cell state is equivalent to the path of information transmission, so that information can be transmitted in the sequence. You can think of it as the "memory" of the network. In theory, cell state can transmit relevant information during sequence processing all the time. Therefore, even the information of the earlier time step can be carried to the cells of the later time step, which overcomes the influence of short-term memory. We add and remove information through the "gate" structure, which will learn which information to save or forget in the training process.

Forget Gate decides what information should be discarded or retained. The information from the previous hidden state and the current input information are transferred to the sigmoid function at the same time. The output value is between 0 and 1. The closer to 0, the more it should be discarded, and the closer to 1, the more it should be retained.

$$\mathrm{f_t} = \sigma(\mathrm{W_f} \cdot [\mathrm{h_{t-1}}, \mathrm{x_t}] + \mathrm{b_f}) \tag{2}$$

The input gate is used to update the cell state. First, the information of the previous hidden state and the current input information are transferred to the sigmoid function. Adjust the value between 0 and 1 to determine which information to update. 0 means unimportant, 1 means important. Secondly, the information of the previous hidden state and the current input information should be transferred to the tanh function to create a new candidate value vector. Finally, the output value of sigmoid is multiplied by the output value of tanh. The output value of sigmoid will determine which information in the output value of tanh is important and needs to be retained.

$$\mathrm{i_t} = \sigma(\mathrm{W_i} \cdot [\mathrm{h_{t-1}}, \mathrm{x_t}] + \mathrm{b_i}) \tag{3}$$

$$\tilde{C_t}=\tanh(\mathrm{W_C} \cdot [\mathrm{h_{t-1}}, \mathrm{x_t}] + \mathrm{b_C}) \tag{4}$$

The cell state of the previous layer is multiplied by the forgetting vector point by point. If it is multiplied by a value close to 0, it means that this information needs to be discarded in the new cell state. Then add this value and the output value of the input gate point by point to update the new information found by the neural network to the cell state. At this point, the updated cell state is obtained.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C_t} \tag{5}$$

The output gate is used to determine the value of the next hidden state, which contains the previously entered information. First, we pass the previous hidden state and current input to the sigmoid function, and then pass the newly obtained cell state to the tanh function. Finally, the output of tanh is multiplied by the output of sigmoid to determine the information that the hidden state should

carry. Then take the hidden state as the output of the current cell and transfer the new cell state and new hidden state to the next time step.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = O_t \odot \tanh C_t \tag{7}$$

## 3. Results and Discussion

### 3.1. LSTM Network Regression

One of the simplest ideas in time series prediction is to find the relationship between current and past data ($X_t, X_{t-1}, \ldots$) and future data ($X_{t+1}$), which is usually expressed as a regression problem. Regression theory must be used to understand the time series prediction problem, and an LSTM network will be built to make predictions. In the first method, the data from month t-1 are used to forecast the data from month t.

However, since training a neural network model involves some unpredictability, it is vital to standardize the environment in which random numbers are generated. Set.seed(7) is a function in R that simulates a random environment. The stock price of the S&P 500 exhibits an upward trend, no seasonality, no periodicity, and substantial non-stationarity, as can be shown by carefully examining the image.

The first objective is to separate the data into a training set and a test set, which make up respectively 4/5 and 1/5 of the total data set. The "scale" of data affects the LSTM network. The normalization described above, which scales the data between 0 and 1, is preferable. The formula used for normalization is, ∀ x in X, where X is the actual stock price:

$$x_{normalized} = \frac{x}{\max(x) - \min(x)} \tag{8}$$

Two matrices are created from the input, namely "historical data" (as the prediction factor) and "future data" (as the prediction goal), in order to train the neural network to preprocess the data. Here, we make predictions using historical information from the previous month. It is evident from the above and below that translation misalignment nearly always occurs; the only time the actual and anticipated curves agree is between 12/2018 and 06/2021 as shown in Figure 2. Through the built-in function evaluate () in R language, we can know that the training score is 3433.9377 MSE (58.5998 RMSE) and the test score is 3679.0157 MSE (60.6549 RMSE).

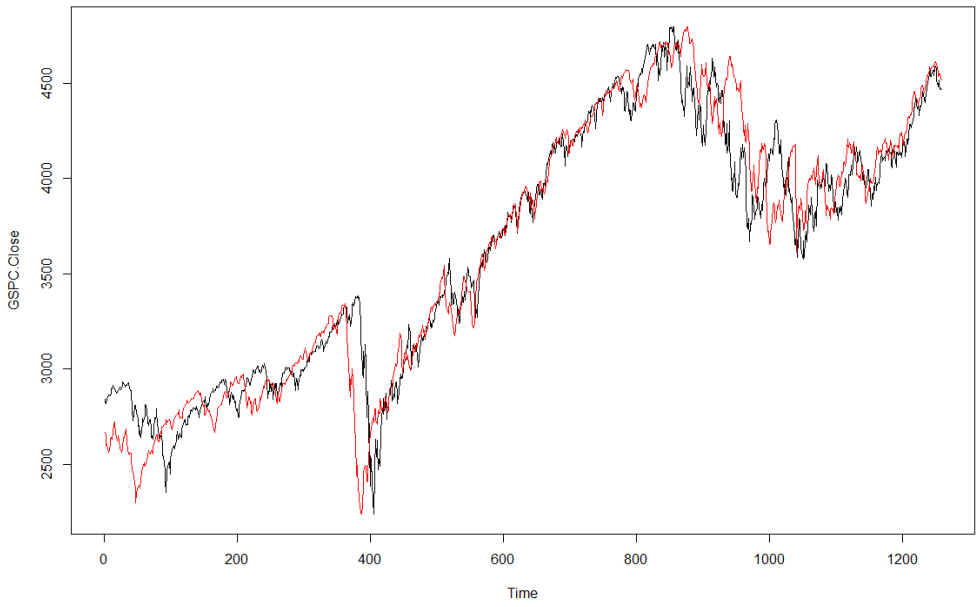


**Fig. 2** Model of look_back = 1

### 3.2. LSTM Network combined with Window Method

In contrast, the above example demonstrates how challenging it is for the neural network model to identify the structural properties of "seasonality" and "periodicity" if $X_t$ is simply utilized to forecast $X_{t+1}$. As a result, it is important to train the model with more historical data (including a whole cycle) and attempt widening the "window". As a result, the machine now requires a 30-day window of data to look back on.

Similar to multi-layer perceptron regression, the outcome is the same. The new model essentially solves the "translation dislocation" phenomena while maintaining the ability to recognize the two key trends of linear development and progressive rise in volatility as shown in Figure 3. The built-in function evaluate () of the R programming language allows us to determine that the test score is 3725.5603 MSE (61.0374 RMSE) and the training score is 2439.4888 MSE (49.3912 RMSE).

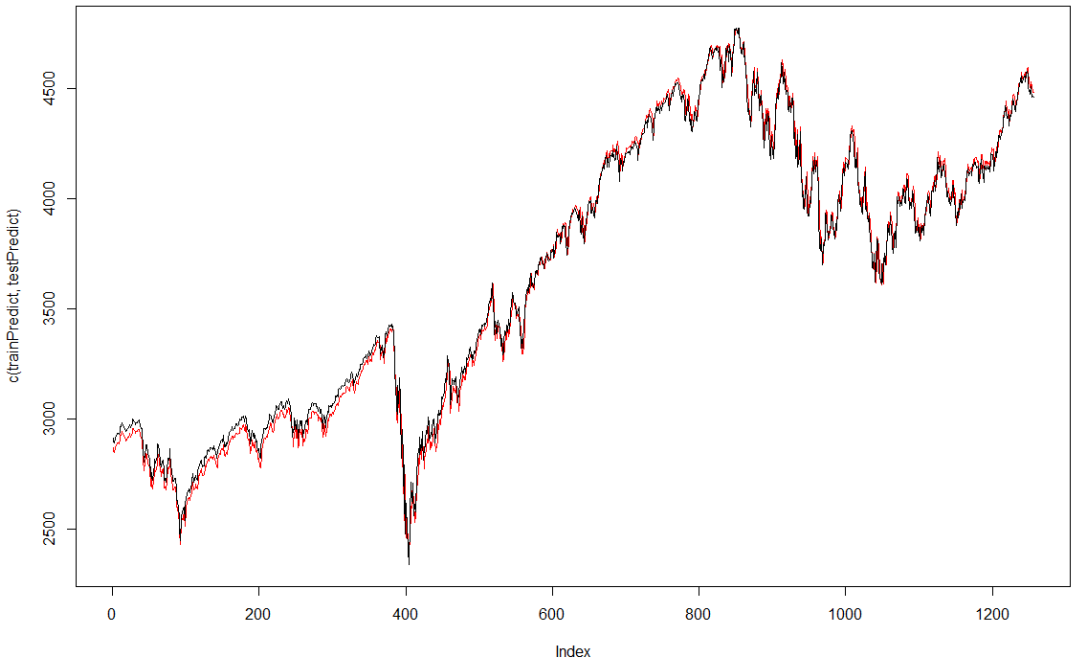


**Fig. 3** Model of look_back = 30

Upon careful examination of the visual representation provided in the aforementioned figure, it becomes readily apparent that the previously encountered issue of translation misalignment, stemming from the training process of the model, has undergone a significant enhancement. This improvement can be attributed to the refined training procedures and the utilization of advanced methodologies. Moreover, a noteworthy inference can be drawn from the analysis, specifically highlighting the distinctive presence of monthly cyclicity inherent in the closing price trends of the S&P 500 index. This intriguing observation gains further credence from the fact that the model, trained with a look_back value of 30, demonstrates a notably enhanced congruence with the actual price trajectory. This congruence, when juxtaposed with the visual representation, elucidates a more accurate alignment with the real-world market dynamics, subsequently emphasizing the practical significance of the model's performance in capturing these cyclical patterns within the S&P 500's closing prices.

### 3.3. Discussion

The association between stock prices and different factors that influence them is frequently nonlinear. LSTMs can capture such nonlinearities by acquiring detailed data patterns, which paves the way for more precise predictions. Furthermore, LSTMs have the capacity to learn pertinent

features from input data, eliminating the prerequisite for manual feature engineering. In the context of stock price forecasting, where determining the most relevant features is arduous, this is a crucial advantage. Importantly, Long Short-Term Memory networks (LSTMs) contain memory cells that enable the storage and retrieval of information for extended periods. This memory function assists the network in recognizing recurrent patterns and trends within the stock market.

However, sudden and dramatic fluctuations in the market can occur as a result of unforeseen events. Like all predictive methods, LSTM models may find it difficult to make precise forecasts during periods of extreme instability. Deep learning models, including LSTMs, may suffer from overfitting if regularization is not properly implemented. To mitigate this issue, techniques such as dropout and early stopping should be used. Moreover, stock prices are affected by numerous factors, such as economic indicators, company news, geopolitical events, and others. Integrating these factors into the model can be challenging and intricate.

Predicting S&P 500 stock prices using LSTM networks presents a promising avenue for enhancing stock market analysis and decision-making. These networks can capture temporal dependencies, handle nonlinear relationships, and automatically learn relevant features from historical price data. Nonetheless, challenges, such as data quality, market volatility, and the consideration of multiple influencing factors, must be carefully addressed. As deep learning methods keep advancing, more study and development of LSTM models may potentially result in enhanced predictions, ultimately assisting investors and analysts in maneuvering the intricate landscape of financial markets.

## 4. Conclusion

In conclusion, a significant advancement in the field of financial forecasting is the use of LSTM networks to predict stock prices. The ability of LSTMs to capture complex temporal correlations in stock data has proven to be a valuable asset in achieving more accurate forecasts compared to traditional methods. However, it is important to recognize the inherent challenges of financial markets, such as their inherent volatility and susceptibility to unanticipated events, which can affect the reliability of any forecasting model.

While LSTMs provide promising results, they are not immune to limitations. The complexity of training and tuning LSTM models, coupled with the risk of overfitting, requires careful attention during model development. In addition, even the most sophisticated algorithms cannot completely eliminate the unpredictability of financial markets. Therefore, it is critical to integrate LSTM predictions into a comprehensive investment strategy, as well as thorough risk management and fundamental analysis.

In the ever-evolving world of technology and finance, LSTM-based stock price prediction models provide powerful tools for traders, investors, and financial analysts. As research and innovation continue, refining these models and combining them with other advanced techniques can produce more robust and reliable forecasts. Ultimately, while LSTM enhances our predictive capabilities, making informed decisions in the financial world requires a holistic approach that recognizes both the potential of artificial intelligence and the inherent uncertainty of markets.

## References

[1] R Akita, A Yoshihara, T Matsubara, K Uehara. Deep learning for stock prediction using numerical and textual information. 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, 2016, 1-6.

[2] Shu H. Investor mood and financial markets. Journal of Economic Behavior and Organization, 2010, 76(2): 267–282.

[3] Hajizadeh E, et al. A hybrid modeling approach for forecasting the volatility of S&P 500 index return. Expert Systems With Applications, 2012, 39(1): 431–436.

[4] Hansen P R, Lunde A. A Forecast Comparison of Volatility Models: Does Anything Beat a GARCH(1,1). Social Science Research Network, 2001.

[5] Schumaker R P, Johnson J W. An Investigation of SVM Regression to Predict Longshot Greyhound Races. Communications of the IIMA, 2014, 8(2).

[6] Ding C, Bao T, Huang H. Quantum-Inspired Support Vector Machine. Journal of Latex Class Files, 2019.

[7] Basak S, et al. Predicting the direction of stock market prices using tree-based classifiers. The North American Journal of Economics and Finance, 2019, 47: 552-567.

[8] Ticknor J L. A Bayesian regularized artificial neural network for stock market forecasting. Expert Systems With Applications, 2013, 40(14): 5501-5506.

[9] S&P 500: ^GSPC. Yahoo Finance, 2023. https://finance.yahoo.com/quote/%5EGSPC/history?period1=1533600000&period2=1691366400&interval=1d&filter=history&frequency=1d&includeAdjustedClose=true

[10] Zhou S K, et al. Handbook of Medical Image Computing and Computer Assisted Intervention. In Elsevier eBooks, 2020.