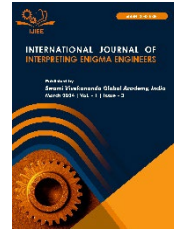




AI FRONTIER TRADING INSIGHTS WITH NEURAL NETWORK MARKET PREDICTIONS



Kranthi Kumar Mandalaju, Aishwarya Govindkar

Original Article

Manager, SAP CRM at Accenture, Pune.

Software Engineer-2 at Dell Technologies, Hyderabad.

*Corresponding Author's Email: kranthi.244379@gmail.com, aishwaryagovindkar@gmail.com

Abstract

Estimation of stock prices is very important for traders and investors to make intelligent decisions. In order to understand the relationship between previous stock prices and future prices, this paper gives an overview of Recurrent Neural Networks (RNNs) that are used in predicting stock prices. Recurrent Neural Networks are designed to identify relationships between past and future states of a time series and non-linear patterns in sequential data, in opposite to traditional methods like multiple linear regression. After collecting and preprocessing previous stock data, the RNN model is trained to estimate future stock prices. By differentiating actual values with expected values, the model's accuracy is evaluated and market trends are displayed. This method is used on international markets as well as the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE) in India, giving investors access to useful data to help them make data-driven decisions.

Keywords: *Stock Price Prediction, Recurrent Neural Network, Long Short Term Memory, Historical Data, Market Trends.*

Introduction

With the current unpredictability in the financial markets, traders and investors who wish to make wise judgments are finding that accurate stock price prediction techniques are more and more necessary. The aggressive changing and complex character of stock price is frequently above the reach of traditional methodologies. As a result, complex techniques like as Recurrent Neural Networks (RNNs) have introduced as powerful tools that are capable of simulating the time dependent relationships and non-linear patterns found in financial data. This study uses RNNs to analyze previous stock prices and other financial data in order to give predictions that help guide investing strategies. By accounting for factors including past stock prices, trade volumes, and market trends, the RNN model increases the accuracy of stock price predictions.

Literature Survey

According to the paper the model independent stock price prediction could be done by deep learning methods. For short term future values it uses sliding window method and evaluates three distinct designs for NSE listed companies. The percentage error applied for examining the performance of these models shows how important the stock market price is in influencing investor profits [1]. It analyses whether customer sentiment and market trends affect the stock sentiment and compares the model against traditional machine learning algorithms [2]. It could be difficult to predict stock market volatility because of elements like physical conditions, mood of the investor and rumors. By using machine learning algorithms to previous stock price data, data analysis can help in recognising trends and producing accurate

predictions[3]. In order to estimate future stock values with better accuracy, the study examines the utilization of machine learning techniques, particularly Recurrent Neural Networks and Long Short Term Memory models[4]. Using Recurrent Neural Network, Long Short Term Memory(LSTM), and Bi-Directional Short Term Memory(BI-LSTM), this study presents an innovative framework for estimating stock prices. When the hyper parameters are neatly adjusted the proposed model can accurately predict future stock patterns. Using a dataset which is openly available for stock market with open, high, low, closing prices the RMSE for each model was measured[5]. The NIFTY 50 index records from 2008 to 2020 were used in this study to offer a variety of deep learning algorithms for stock predicting price of stock. The model contains long and short term memory networks and two convolutional neural networks. The most rapid model is the univariate CNN, while the best accurate model is the univariate encoder decoder convolutional LSTM, based on the results[6]. In this study, nine predictors were used to estimate the closing price of the S & P 500 index using long short term memory (LSTM). The single layer LSTM model gives greater precision according to the results[7]. The global economy public attitude, supply and demand ratio, market trends, and financial information represent a few of the factors that affect prices in the stock market, making it unpredictable. However, modern tools like machine learning and data mining can assess huge amount of data and create exact forecasting models. The study gives a supervised machine learning model for stock price estimation utilizing sample data of stock closing prices[8]. The study estimates the final stock price of companies in distinct regions for next day using Random Forest and Artificial Neural Network techniques. The model uses their efficacy in predicting stock closing prices using financial data like open, high, low, close prices. They are evaluated using RMSE and MAPE[9]. Demand and current data, particularly opinions posted on social media are important variables in estimating stock prices. 87.6% accuracy in stock price prediction can be obtained through sentimental analysis using the Naive Bayes and Random Forest algorithms. A linear regression model with values of determination of 0.9989 and 0.9983 is used in this study to the Indonesian stock market market[10-13].

Proposed Methodology

The primary goal of our paper is to predict future stock prices using previous data so that investors can make well informed decisions. Time series forecasting is an ideal for Recurrent Neural Networks(RNN) accompanied by Long Short Term(LSTM) layers. The method analyzes patterns and trends to make predictions regarding future stock movements by using previous stock prices and a range of market factors as training data. The capacity of the model to identify time related relationships in the data is made easier by using LSTM networks, and they also permits the model to make accurate predictions that can help in the design of trading techniques. Figure 1 specifies the architecture of RNN model.

The architecture starts from loading previous data of stock prices, usually from a CSV file. The closing price of the stocks is taken as the target feature. Dates are transformed into datetime objects and set as index for easier time series analysis. The data is converted into sequences where each sequence to forecast the future day’s stock price. Further data is sent to LSTM model for training.

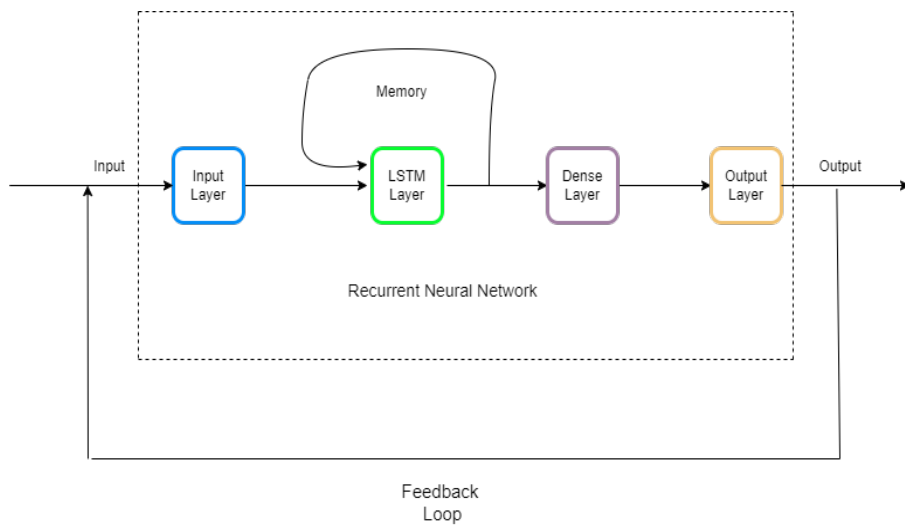


Figure 1 : Recurrent Neural Network’s System Architecture

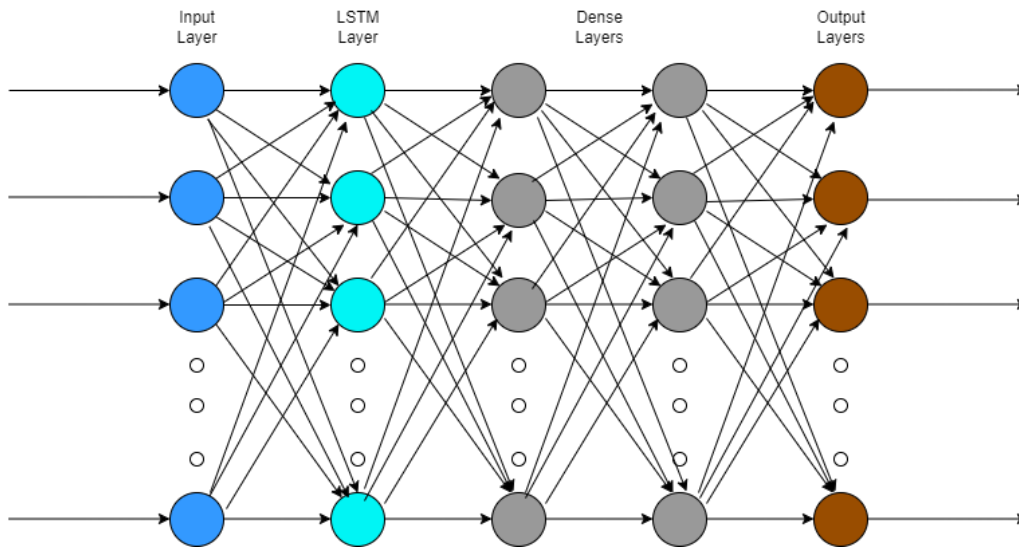


Figure 2 : Layers of LSTM Model

Figure 2 shows the three different layers of LSTM model which are : Input Layer, LSTM Layers, Dense Layers. Collecting the prepared time series data is the input layer. Both of the LSTM layers in the architecture have 50 units each. Sequences to be sent onto the next LSTM layer are given back by the first LSTM layer, but the second does not, preparing the data for the dense layer. The Dense layer gives the predicted stock price by using a single unit and a fully connected layer.

Methodology

Step 1: The NSE closing prices of Tata Global Beverages Limited have been included in the stock price data that has been gathered. The proposed model uses this data as its input data. The historical closing prices are presented in Figure 3 highlighting the patterns and trends over time.



Figure 3 : History of closing stock prices of Tata Global Beverages Limited

Step 2: The data is then preprocessed by handling null values and normalizing the data

a) Handling Null Values :

Firstly any null(missing) values in the dataset are examined. The dataset's consistency remains unchanged by replacing any null values that are found with the mean of the relevant columns. In mathematical terms, if X is the dataset with columns b_1, b_2, \dots, b_m and rows a_1, a_2, \dots, a_n , then for each null value q , a function $f(q)$ is defined as $f(q)$ equals to 1 if q is null otherwise 0. If $f(q) = 1$, the null value is replaced by the mean of the corresponding column.

b) Data Normalization:

Using MinMaxScalar, the closing prices are scaled from 0 to 1 to normalize the results. During training, this scaling improves the model's stability.

Step 3: This dataset split as two parts such as training and testing sets. The training set is again divided into training set and validation set. 20% of the data is utilized for testing, 80% of the data is used for training. An additional 20% of the training data utilized for validation and the rest 80% is used for training the model. In mathematical terms, the split is as follows if dataset X has n samples then, Training Set = $0.7n$, Validation Set = $0.2n$, Test Set = $0.1n$.

Step 4: LSTM Model Architecture:

Two LSTM layers and a Dense layer are placed in order of construction in a sequential approach. LSTM Layer 1 comprise of 50 units, which also provides stacking sequences. No sequences are given back by the 50 unit LSTM Layer 2. Dense Layer is the layer that predicts the closing price using only one unit. The loss function is Mean Squared Error(MSE) and Adam optimizer to train the model.

Step 5: Proposed model is then trained using the training data with 10 epochs and a batch size 32, including a validation split of 20%. After training, the model is assessed on the test set. The following metrics are calculated using (1),(2),(3),(4)and (5).

$$\text{Mean Absolute Error (MAE)} = \frac{1}{n} \sum_{i=1}^n |b_i - \hat{b}_i| \dots (1)$$

it calculates the mean magnitude of errors in a set of estimations, without taking consideration their direction.

$$\text{Mean Squared Error (MSE)} = \frac{1}{n} \sum_{i=1}^n (b_i - \hat{b}_i)^2 \dots (2)$$

it gives mean squared variation of actual and predicted values.

$$\text{Root Mean Squared Error (RMSE)} = \sqrt{MSE} \dots (3)$$

it gives the MSE sqareroot, giving error in the same units as the target variable.

$$R^2 \text{ Score} = 1 - \frac{\sum_{i=1}^n (b_i - \hat{b}_i)^2}{\sum_{i=1}^n (b_i - \bar{b}_i)^2} \dots (4)$$

it measures how the model's estimations match the actual data.

$$\text{Mean Absolute Percentage Error (MAPE)} = \frac{1}{n} \sum_{i=1}^n \left| \frac{b_i - \hat{b}_i}{b_i} \right| \times 100 \dots (5)$$

it gives the accuracy of the predictions as a percentage.

Step 6 : This test set is used to asses the final proposed model. Accuracy of proposed model is calculated by using the equation (6).

$$\text{Acc} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} 1(\hat{b}_i = b_i) \dots (6)$$

Result and Discussion

After training Recurrent Neural Network(RNN) model on NSE-Tata-Global-Beverages-Limited dataset, we analyzed its performance using a variety of statistical metrics. 80% of stock data is utilized for model training and remaining data is utilized for model testing. The main conclusions are as follows:

The R^2 score measures how well the model estimates closing price, a score closer to 1 indicates better-fitness of model. The R^2 score of the proposed model is approximately 0.54. The model achieved a precision of 97% approximately on the test set. The MAE of the actual and predicted values was 4.3 approximately. This measure shows the mean absolute variation of the actual closing prices and the values estimated by the model. A lower MAE indicates a more accurate model. The MSE of the model was approximately 34.8 and the RMSE was approximately 5.9. While RMSE is the MSE square root of actual and predicted values, it gives a clearer metric in the same unit as the target variable which is the closing price. Better model performance is shown by lower MSE and RMSE values. The MAPE was approximately 2%. Mean Absolute Percentage Error gives the error as percentage, in order to make it easier to explain the model's performance across distinct scales of data. Figure 4 shows a line graph comparing original stock prices with predicted stock prices visualizing how well the model's estimated prices placed with the actual prices.

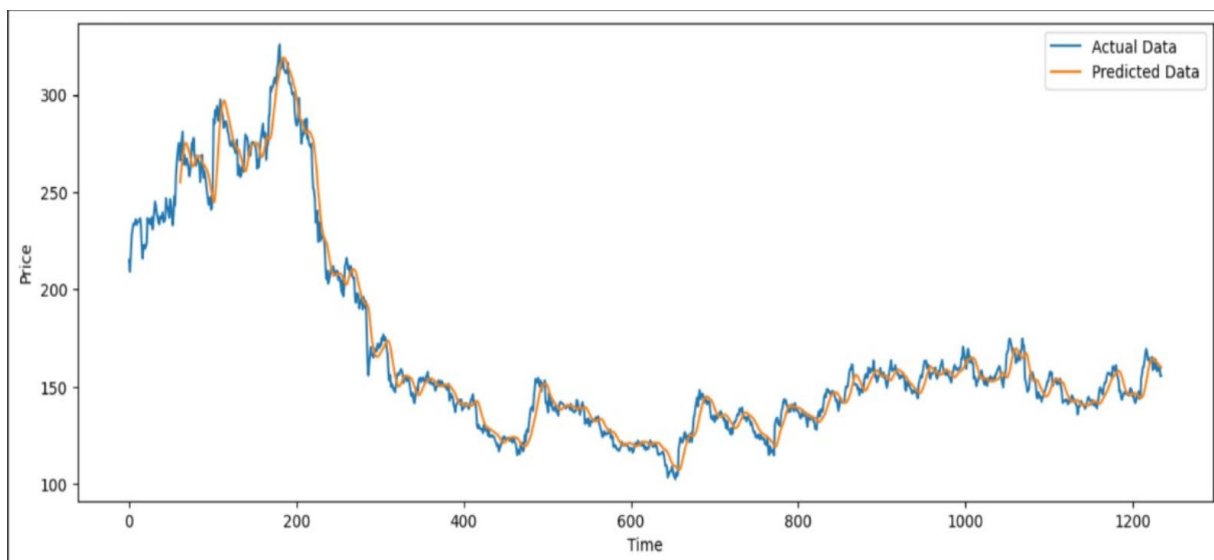


Figure 4 : Line graph comparing actual stock prices with predicted stock prices

Conclusion

In conclusion, we have shown how to implement Recurrent neural Networks(RNNs) to estimate stock prices through data driven approach. The model proposed, successfully captures the complex patterns and temporal correlations in stock price movements by using previous stock data and financial indicators. This enables us to give accurate predictions that can help investors in making well informed decisions. By using modern machine learning techniques, this approach gives the potential to enhance investment strategies, reduce financial risks and improve portfolio management.

In the future, development may concentrate on adding real-time data from live market feeds, integrating advanced deep learning architectures, and expanding the dataset to include a wider range of financial indicators and market patterns. The model's application can be expanded by developing a simple user interface for financial analysts and investors, which would make it a better tool for complexity analysis of the stock market.

References

- [1] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 2017, pp. 1643-1647, doi: 10.1109/ICACCI.2017.8126078. keywords: {Companies;Time series analysis;Logic gates;Predictive models;Forecasting;Data models;Machine learning;Time series;Stock market;RNN;LSTM;CNN}
- [2] Pawar, K., Jalem, R.S., Tiwari, V. (2019). Stock Market Price Prediction Using LSTM RNN. In: Rathore, V., Worrying, M., Mishra, D., Joshi, A., Maheshwari, S. (eds) Emerging Trends in Expert Applications and Security. Advances in Intelligent Systems and Computing, vol 841. Springer, Singapore. https://doi.org/10.1007/978-981-13-2285-3_58
- [3] Ghosh, A., Bose, S., Maji, G., Debnath, N., & Sen, S. (2019, September). Stock price prediction using LSTM on Indian share market. In Proceedings of 32nd international conference on (Vol. 63, pp. 101-110).
- [4] Moghar, A., & Hamiche, M. (2020). Stock market prediction using LSTM recurrent neural network. *Procedia computer science*, 170, 1168-1173.
- [5] Sunny, M. A. I., Maswood, M. M. S., & Alharbi, A. G. (2020, October). Deep learning-based stock price prediction using LSTM and bi-directional LSTM model. In 2020 2nd novel intelligent and leading emerging sciences conference (NILES) (pp. 87-92). IEEE.
- [6] Mehtab, S., & Sen, J. (2020, November). Stock price prediction using CNN and LSTM-based deep learning models. In 2020 International Conference on Decision Aid Sciences and Application (DASA) (pp. 447-453). IEEE.
- [7] Bhandari, H. N., Rimal, B., Pokhrel, N. R., Rimal, R., Dahal, K. R., & Khatri, R. K. (2022). Predicting stock market index using LSTM. *Machine Learning with Applications*, 9, 100320.
- [8] Jahan, I., & Sajal, S. (2018). Stock price prediction using recurrent neural network (RNN) algorithm on time-series data. In 2018 Midwest instruction and computing symposium. Duluth, Minnesota, USA: MSRP.
- [9] Mehar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, Arun Kumar, "Stock Closing Price Prediction using Machine Learning Techniques", International Conference on Computational Intelligence and Data Science (ICCIDS 2019)
- [10] Kishor Kumar Reddy C and Vijaya Babu B, "ISLGAS: Improved Supervised Learning in Quest using Gain Ratio as Attribute Selection Measure", International Journal of Control Theory and Applications, 2016
- [11] Sreelatha, Gavini, A. Vinaya Babu, and Divya Midhunchakkarvarthy. "A Survey on Cloud Attack Detection using Machine Learning Techniques." *International Journal of Computer Applications* 975 (2020): 8887.
- [12] Cakra, Y. E., & Trisedya, B. D. (2015, October). Stock price prediction using linear regression based on sentiment analysis. In 2015 international conference on advanced computer science and information systems (ICACSIS) (pp. 147-154). IEEE.
- [13] Sreelatha, G., Vinaya Babu, A., Midhunchakkarvarthy, D. (2021). Extended Equilibrium-Based Transfer Learning for Improved Security in Cloud Environment. In: Suma, V., Chen, J.IZ., Baig, Z., Wang, H. (eds) *Inventive Systems and Control. Lecture Notes in Networks and Systems*, vol 204. Springer, Singapore. https://doi.org/10.1007/978-981-16-1395-1_4