IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-4, April 2019 E-ISSN: 2347-2693

Stock Market Prediction using Deep Neural Networks

Rohit Kumar[1*], Rohit Gajbhiye[2], Isha Nikhar[3], Dyotak Thengdi[4], Sofia Pillai[5]

[1,2,3,4,5]Dept. of Computer Science & Engineering G.H. Raisoni College of Engineering, Nagpur

Corresponding Author: kumar_rohit.ghrcecs@raisoni.net

DOI: https://doi.org/10.26438/ijcse/v7i4.2428 | Available online at: www.ijcseonline.org

Accepted: 11/Apr/2019, Published: 30/Apr/2019

***Abstract*—**Stock Market Prediction is the demonstration of attempting to decide the future estimation of an organization stock or other money related instrument exchanged on a trade. Prediction on stock market is a great challenge as it is complex, dynamic and non-linear in nature. The main focus is on closing price of next day. High, Low, Volume is of importance but the closing price is of more value. There are numerous instances of Machine Learning algorithms possessed the capacity to achieve attractive outcomes while doing that kind of forecast. In this paper, the LSTM networks are used to predict future closing price of stock market based on the price history, alongside with technical analysis indicators. For that objective, a forecast model was built, and a series of experiments were performed and their outcomes were examined against various measurements to survey if this kind of calculation presents and enhancements when contrasted with other Machine Learning techniques.

***Keywords*—**Stock Market, Prediction, LSTM**.**

I. INTRODUCTION

Deep Learning has been ended up being an inconceivable machine learning gadget of late, and it has a wide combination of employments. Regardless, employments of deep learning in the field of computational store are as yet limited. In this undertaking, we execute Long Short-Term Memory (LSTM) arrange, a period arrangement rendition of Deep Neural Networks, to gauge the stock cost for different organization (NSE). LSTM was first made by Hochreiter and Schmidhuber (1997). Throughout the years, it has been associated with various issues that incorporate successive data, and research has shown powerful applications in such fields as natural language preparing and discourse acknowledgment.

The info highlights we use are classified into three classes: (1) the historical data (OHLC Variables), (2) technical indicators, and (3)stock market index. These highlights reflect everyday estimations of these factors, and the system predicts the closing price of next day.

Our trial is made out of three stages. To start with, we pick a model and then trained the network. Second, we make a figure on a test set with the picked model. Third, given the readied framework, we take a gander at the benefit of an algorithmic trading philosophy dependent on the figure made by the model.

Whatever is left of this paper is sorted out as pursues: Section II discusses existing papers and the characteristics and weaknesses of their models. Section III depicts the dataset used in the preliminary. Section IV clears up the models. Segment V describes the trading procedure. Segment VI outlines the preliminary and the results. Segment VII is the end.

II. RELATED WORK

D. Kumar and S. Murugan have presented another strategy by consolidating time-arrangement information with ANN [1]. This expectation demonstrates depends on a feed-forward ANN with back proliferation. The execution of the estimate show is poor down using a couple of components. The key factor incorporates Mean Absolute Error (MAE), Mean Absolute Rate Error (MAPE), Percentage Mean Absolute Deviation (PMAD) and Mean Squared Error (MSE). This execution is controlled by BSE100 and NIFTY MIDCAP50.

K. Abhishek, A. Khairwa, T. Pratap and S. Prakash have developed an anticipating model for the Microsoft Partnership. This guaging model incorporates a two-advance procedure [2]. In the underlying advance, ANN experiences planning with hereditary calculation. This calculation prepares the gauge model to perceive the new examples. In a second step the results are gotten by looking at the readied desire model to the dataset.

M. Hagenan, M. Liebmann, M. Hedwig and D. Neumann have investigated distinctive roads in regards to the dataset of NSE and news data from web based life [3]. They arranged the word mix framework with German adhoc messages and stock expenses. The substance mining is evaluated by Chi-Based part assurance. The Support Vector Machine calculation is used to foresee the stock expense. The game plan precision is upheld by info based feature assurance.

III. DATASET AND FEATURES

A. Dataset

Our dataset is the historical data of various companies for at least 3 years, sourced from either NSE or Yahoo Finance. So as to look at the strength of the models in various timeframes, the dataset is divided into two periods.

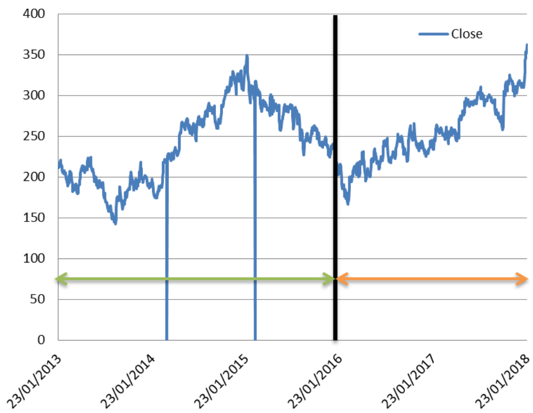


Fig. 1: Stock price of ICICI Bank from 23/01/2013 to 23/01/2018. The green arrows indicate the training set and the orange arrows indicate the testing set.

The period from 23/02/2013 to 23/01/2016. Period II ranges from 24/01/2016 to 23/01/2018. The historical data of ICICI Bank is shown in Figure 1, which outlines how we split the information on the diverse sets.

B. Input Features

The information highlights comprise of three arrangements of factors. The first set is the historical data of organizations including OHLC factors. These highlights give essential data about the stock. The second set is the technical indicators that show different attributes of the stock's conduct. The third set is about indexes: NIFTY 50 and VIX. Figure 2 portrays the subtleties of these factors. The majority of the data sources and yield are scaled somewhere in the range of 0 and 1 before we feed them into the models.

Input Features	Definition
	Category 1 - Historical Data
Open/Close Price	Nominal daily open/close price
High/Low Price	Nominal daily open/close price
Volume	Daily trading volume
	Category 2 - Technical Indicator
MACD	Moving Average Convergence Divergence: a trend-following momentum indicator
BOLL	Bollinger Band: a set of lines plotted two standard deviations away from a simple moving average
RSI	Relative Strength Index: a technical indicator used in the analysis of financial markets
CCI	Commodity channel index: an oscillator used to identify cyclical trends
ROC	Rate of Change: a momentum oscillator that measures the percent change in price from one period to the next
WPR	Williams' Percent Range: a measurement of the buying and selling pressure
	Category 3 – General Market Index
NIFTY 50	National Stock Exchange Fifty: National Stock Exchange of India's benchmark broad based stock market index for the Indian equity market
VIX	CBOE Volatility Index: a popular measure of the stock market's expectation of volatility

Fig. 2: Descriptions of the input features

IV. METHODS

A. Long Short-Term Memory

Long Momentary Memory (LSTM) was first made by Hochreiter and Schmidhuber (1997) as a variety of Monotonous Neural System (RNN). Consistently, LSTM has been associated with various issues that incorporate back to back data, and research has illustrated productive applications in such fields as basic vernacular handling, talk affirmation, and DNA plan.

Like RNN, LSTM has a repetitive structure where every cell yields expectation $\hat{y}_t$ as well as exchanges initiation h_t to the following cell. The striking component of LSTM is its capacity to store, forget, and read data from the long-haul state of the shrouded components, and these errands are accomplished through three kinds of entryways. In the forget gate, a cell gets long-term state c_{t-1}, holds a few bits of the data by sum f_t, and afterward includes new recollections that the information entryway chose. The *input gate* figures out what parts of the changed info g_t should be added to the LTS c_t. This procedure refreshes LTS c_t, which is specifically transmitted to the following cell. At last, *output gate* changes the refreshed LTS c_t through tanh(.), channels it by o_t, and produces the yield $\hat{y}_t$, which is likewise sent to the following cell as the STS h_t.

The conditions for LSTM calculations are given by

$$i_t = \sigma(W_{xi}^T \cdot x_t + W_{hi}^T \cdot h_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf}^T \cdot x_t + W_{hf}^T \cdot h_{t-1} + b_f)$$
$$o_t = \sigma(W_{xo}^T \cdot x_t + W_{ho}^T \cdot h_{t-1} + b_o)$$
$$g_t = \tanh(W_{xg}^T \cdot x_t + W_{hg}^T \cdot h_{t-1} + b_g)$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes g_t$$
$$\hat{y}_t = h_t = o_t \otimes \tanh(c_t)$$

Where is component insightful augmentation, (.) is the strategic capacity, and tanh(.) is the hyperbolic tan function. The three entryways open and close as per the estimation of the door controllers f_t, i_t, and o_t, which are all completely associated layers of neurons. The scope of their yields is [0, 1] as they utilize the strategic capacity for initiation. In each entryway, their yields are sustained into component savvy duplication activities, so if the yield is near 0, the door is limited and less memory is put away in c_t, while if the yield is near 1,the gateway is even more commonly open, allowing more memory to course through the entryway. Given LSTM cells, generally used to stack various layers of the cells to make the model further to no doubt get nonlinearity of the data. Figure 3 outlines how calculation is completed in a LSTM cell.

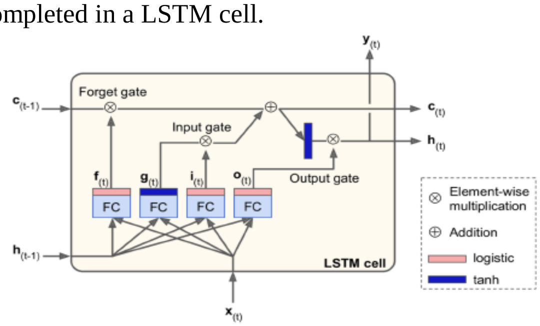


Fig. 3: LSTM cell [3]

We pick Mean Squared Error (MSE) with L2-regularization on the loads for the cost capacity:

$$L(\theta) = \frac{1}{m}\sum_{t=1}^{m}(y_t - \hat{y}_t)^2 + \lambda\left(||W_i||_2 + ||W_f||_2 + ||W_o||_2\right)$$

Where Ɵ is the arrangement of parameters to be prepared including the load grids for each door {$W_i = (W_{xi}; W_{hi})$, $W_f = (W_{xf}; W_{hf})$, $W_o = (W_{xo}; W_{ho})$}, and the inclination terms {b_i, b_f, b_o, b_g}.

Since this is a RNN, LSTM is prepared by means of *Back propagation Through Time*. The key idea is that for each cell, we at first unroll the settled number of past cells and after that apply forward feed and back propagation to the unrolled cells. The quantity of unrolled cells is another hyper parameter that should be chosen notwithstanding the quantity of neurons and layers.

B. Linear Regression

LR is the most generally utilized statistical technique; it is an approach to demonstrate a connection between two arrangements of factors. The outcome is a linear regression condition that can be utilized to make predictions about information.

Regression investigation is utilized to discover conditions that fit information. When we have the regression condition, we can utilize the model to make predictions. One kind of regression investigation is linear examination. At the point when a connection coefficient demonstrates that information is probably going to most likely foresee future results and a disperse plot of the information seems to shape a straight line, you can utilize basic linear regression to locate a prescient capacity. On the off chance that you review from basic polynomial math, the condition for a line is y = mx + b. This article tells you the best way to take information, ascertain linear regression, and discover the condition y' = a + bx.

Linear regression is an approach to show the relationship between two factors. You may likewise perceive the condition as the slant equation. The condition has the frame Y=a+bX, where Y is the reliant variable (that is the variable that goes on the Y pivot), X is the free factor (for example it is plotted on the X hub), b is the incline of the line and is the y-intercept.

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

The initial phase in finding a linear regression condition is to decide whether there is a relationship between the two factors.

C. Reinforcement Learning (RL)

RL is a computational way to deal with knowing and monetizing objective coordinated learning and basic leadership [6]. It is recognized from other computational methodologies by its accentuation learning by the person from direct cooperation with its condition as appeared in Figure 4.

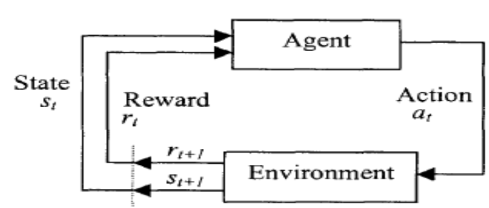


Fig. 4: The operator condition association in reinforcement learning.

At each discrete time step t, the specialist detects the current state st, picks a present activity at, and performs it. The condition reacts by giving the operator a reward rt+1 = r(st, at,) and by creating the succeeding state st+1= δ(st,at). Here the capacities S and r are the bit of the condition and are not really known to the specialist. In MDP (Markov choice process), the capacities δ(st,at) what's more, r(st, at) depend just on the present state and activity not on prior states or activities. The errand of the specialist is to learn an arrangement, π: S→A, where S is the arrangement of states and An is the set of activities, for choosing its next activity dependent on the current watched state s; that is, π(st)=a,. An ideal approach is a strategy that can expand the conceivable reward from a state, called esteem, Vπ(s) for all states. (1) is a regular meaning of this:

$$V^{\pi}(s_t) = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} . \qquad (1)$$

Here γ (0 < γ < 1) is a consistent that decides the relative estimation of postponed versus prompt prizes. $V^{\pi(s)}$ means the conceivable aggregate reward accomplished by following an self-assertive arrangement π from a discretionary beginning state. At that point an ideal arrangement, π*, is characterized as pursues:

$$\pi^* = \arg\max_{\pi} V^{\pi}(s), (\forall s) . \qquad (2)$$

An iterative procedure in Figure 5, called Generalized Policy Emphasis (GPI), is important to get π*.

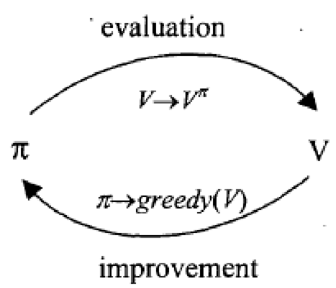


Fig. 5: Generalized policy iteration

V. TRADING STRATEGY

We consider a straightforward algorithmic trading methodology based on the prediction by the model. At day t, a financial specialist purchases one offer of a stock if the anticipated cost is higher than the current actual closing price. Something else, the person moves one offer of a stock. The methodology s_t can be portrayed as:

$$s_t = \begin{cases} +1 & \text{if } \hat{y}_{t+1} > y_t \\ -1 & \text{if } \hat{y}_{t+1} \leq y_t \end{cases}$$

where y_t is the current balanced closing price of a stock what's more, $\hat{y}_{t+1}$ is the anticipated price by the model. Utilizing the pointer variable S_t, we can figure an everyday return of the system at day t + 1:

$$r_{t+1} = s_t \times \log\left(\frac{y_{t+1}}{y_t}\right) - c$$

where c means transaction cost, and the combined come back from t = 0 to m is

$$r_0^m = \sum_{t=0}^{m-1} r_t .$$

VI. EXPERIMENT AND RESULTS

A. Setup for the Experiment

We utilize *TensorFlow* to play out our analysis on the dataset. The choices of hyper parameters for LSTM Model are recorded in Table I:

Categories	Choice
Library	*TensorFlow*
Optimizer	*AdamOptimizer*
No. of Hidden Layers	*1*
No. of Neurons	*500*
No. of Epochs	*3000*

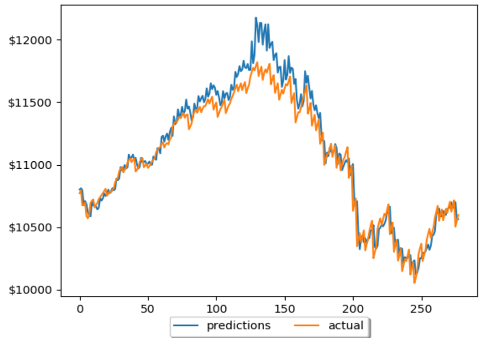
Table II: Hyper parameters for the LSTM Model

We pick one layer as per the system. As to number of unrolled cells,10 (days) is thought to be adequate for the LSTM to foresee the next day's stock closing price and stay away from the disappearing slope issue. The quantity of neurons is dictated by attempt and mistake.

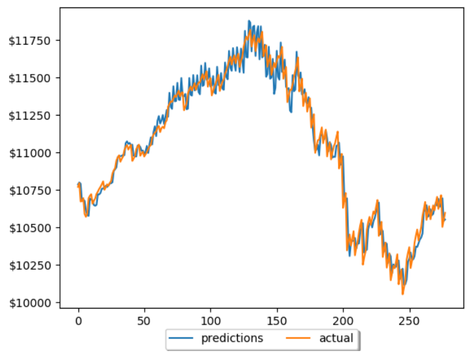
Since the architecture of the LSTM is 1 layers with 500 neurons, which is profound and wide, it is vital to present regularization as examined in Section IV so as to abstain from over fitting and enhance the prescient precision. For every period, we increment no. of significant lots of informational collection and train the system on various epochs.
AdamOptimizer is picked since it is sensible for profound learning issues with huge dataset and various parameters. As for the parameters of AdamOptimizer, we use the default regards given by Tensorflow.

B. The results of the experiment

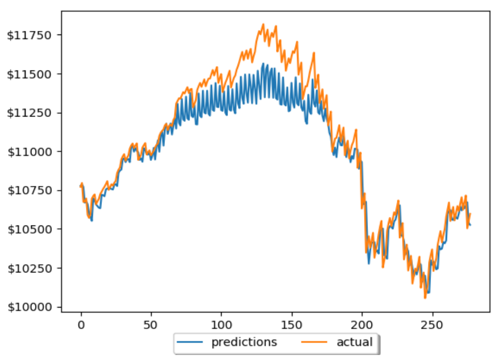
Figure 6 demonstrates the predicted product price by the LSTM model for different epochs along with the actual closing price of an organization. The predicted prices of the LSTM are nearer to the actual price. It is essential to note that the LSTM is by all accounts ready to foresee the stock price all the more precisely when the price does not display a clear pattern.



(a) One thousand epochs



(b) Two thousand epochs



(c) Three thousand epochs

Fig. 6: Predicted and actual price of ICICI Bank for 3 different epochs

To additionally assess the predictive execution of the models, we ascertain the estimations and analyse the productivity of the algorithmic exchanging methodology.

Mean Squared Error of Predicted Price:

$$\mathrm{MSE} = \frac{1}{m}\sum_{t=1}^{m}(y_t - \hat{y}_t)^2$$

where y_t and $\hat{y}_t$ mean the actual and predicted prices of ICICI at day t. The MSE of the two models are recorded in Table II. The MSE of the LSTM on the test sets turns out to be little and lower. This outcome substantiates that the LSTM accomplishes higher prescient precision.

Period	Train Error	Test Error
I	0.00124	0.00564
II	0.00051	0.00762
III	0.00057	0.01126

Table II: The MSE of the LSTM for the three periods

VII. CONCLUSION

In this undertaking, we actualize Long Short-Term Memory Network to anticipate the stock cost of an organization and apply the prepared network to the algorithmic exchanging issue. The LSTM can precisely anticipate the following day's end cost of an organization particularly when the stock cost is absence of a pattern. The procedure dependent on the forecast by the LSTM makes promising aggregate day by day returns, outflanking the other two procedures. These outcomes, in any case, are constrained in a few regards. We expect that it is constantly conceivable to exchange at the balanced shutting value each day, which isn't attainable practically speaking. However, our examination illustrates the capability of LSTM in securities exchange expectation and algorithmic exchanging.

REFERENCES

[1] Kumar, D. Ashok, and S. Murugan. "Performance analysis of Indian stock market index using neural network time series model." In Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on, pp. 72-78. IEEE, 2013.

[2] Abhishek, Kumar, Anshul Khairwa, Tej Pratap, and Surya Prakash. "A stock market prediction model using ANN." In Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on, pp. 1-5. IEEE, 2012.

[3] Hagenau, Michael, Michael Liebmann, Markus Hedwig, and Dirk Neumann. "Automated news reading: Stock price prediction based on financial news using context-specific features." In System Science (HICSS), 2012 45th Hawaii International Conference on, pp. 1040-1049.IEEE, 2012.

[4] US. Sutton and A.G. Barto, Reinforcement Learning, The MIT Press, 1998.

[5] R.S. Sutton, "Learning to foresee by the technique for worldly contrasts," Machine Learning, 3:9-44, 1988.

[6] P.J. Werbos, “Building and understanding adaptive systems: A statistical/numericaI approach to factory automation and brain research,” IEEE Transactions on Systems, Man, and Cybernetics, 17:7-20, 1987.

[7] S.M. Kendall and K. Ord, Time Series, Oxford University Press, New York, 1997.

[8] N. BaBa and M. Kozaki, “An intelligent forecasting system of stock price using neural networks,” Proc. IJCNN, Baltimore, Maryland, 652-657, 1992.

[9] R.J. Kuo, “A decision support system for the stock market through integration of fuzzy neural networks and fuzzy Delphi,” Applied Arlijkial Intelligence, 6501 -520, 1998.

[10] Sheta, A. F., Ahmed, S. E. M., & Faris, H. (2015). A comparison between regression, artificial neural networks and support vector machines for predicting stock market index. Soft Computing, 7, 8.

[11] oone, L., Giorno, C., & Richardson, P. (1998), Stock market fluctuations and consumption behaviour: some recent evidence (No. 208). OECD Publishing.

[12] Gharehchopogh, F.S., Mohammadi, P., &Hakimi, P. (2012). Application of Decision Tree Algorithm for Data Mining in Healthcare Operations: A Case Study. International Journal of Computer Applications, 52(6), 21-26.

