Stock Market Prediction Using LSTM

Utkarsh Anuse[1], Toshal Agashe[2], Jatin Bhangaonkar[3], Prof. Madhuri Ghuge[4]
Dept of Computer Engineering
Bharti Vidyapeeth College of Engineering, Navi Mumbai.
India - 400614

Abstract- *Stock Market Prediction is a software model to analyze trends in stock market and predict their values in the near future. In a economically volatile market like stock market, it is necessary to predict future trends. It is important to have a secure prediction of the values of the stocks. Predicting such an uncertain market requires advanced algorithms of machine learning. This paper contains research of different data mining and machine learning algorithms such as ANN (artificial neural networks). The results of this study will show how the algorithm of machine learning can be used in predicting stock values which will have the success of making a profit.*

Keywords- stock market prediction, artificial neural networks, stock forecast, market trends

I. INTRODUCTION

In today's fast growing market, predicting stock values is miraculous achievement. Over the years, stock brokers and investors have faced a lot of trouble in determining stock values. The market is as uncertain as the world is. As it is impossible to predict what will happen next in the real world, the same is for stock market trends. The method of trying to determine the future price of a company stock or any financial commodity traded in a market is called Stock market prediction. Significant profits could be yielded by the successful prediction of a stock's future price . The efficient-market-hypothesis suggests that stock price changes are not based on newly revealed information instead they reflect all currently available information and thus are inherently unpredictable. Some disagree and have various methods and technologies which allows them to gain future price information. The efficient market hypothesis posits newly revealed information about a company's prospects is almost immediately reflected in the current stock price. Fundamental Analyst evaluate a company's past performance as well as the credibility of its accounts. Many performance ratios are created that enables the analyst to assess the validity of a stock, such as the P/E ratio. Fundamental analysis is built on the notion that human society needs money to make progress and if a corporation functions well, it should be rewarded with additional capital and result in overflow in stock price. Fundamental analysis is widely used by fund managers as it is made from publicly available information like financial statement analysis.

Another aspect of fundamental analysis is top-down analysis from first analyzing the global economy, followed by country analysis and then sector analysis, and finally the company level analysis.

II. LITERATURE SURVEY

The initial focus of our literature survey was to explore generic online learning algorithms and see if they could be adapted to our use case i.e., working on real-time stock price data. These included Online AUC Maximization [1], Online Transfer Learning [2], and Online Feature Selection [3]. However, as we were unable to find any potential adaptation of these for stock price prediction, we then decided to look at the existing systems [4], analyze the major drawbacks of the same, and see if we could improve upon them. We zeroed in on the correlation between stock data (in the form of dynamic, long-term temporal dependencies between stock prices) as the key issue that we wished to solve. A brief search of generic solutions to the above problem led us to RNN's [5] and LSTM [6]. After deciding to use an LSTM neural network to perform stock prediction, we consulted a number of papers to study the concept of gradient descent and its various types. We concluded our literature survey by looking at how gradient descent can be used to tune the weights of an LSTM network [8] and how this process can be optimized. [9]

III. STUDY

The basic idea behind this project is to collect sufficient amount of informative data and predicting the future values based upon that data. It is feasible and possible to determine a stock trend a year from now than in the immediate hour. It is practical to predict whether the stock value will rise or fall. But predicting the accurate value that it will hold in the future is a bit tricky and unreliable. Data mining as part of information mining is a significant breakthrough for this model. Big heap of data about stock market attributes can be stored in datamarts. This data can be put into machine learning algorithms to predict a new value. Various algorithms and test cases contributed to analyze the spontaneity and unpredictability of the stock market.

IV. METHODOLOGY

Different types of neural networks can be developed by the combination of various factors like network topology, training method etc. For this model, we have considered Recurrent Neural Network and Long Short-Term Memory. This section we will discuss the methodology of our system. Our system consists of several stages which are as follows:-

Stage 1: Raw Data:

In this stage, the historical stock data is collected from https://www.quandl.com/data/NSE and this historical data is used for the prediction of future stock prices.

Stage 2: Data Preprocessing:

The pre-processing stage involves

a) Data discretization: Part of data reduction but with particular importance, especially for numerical data.
b) Data transformation: Normalization.
c) Data cleaning: Fill in missing values.
d) Data integration: Integration of data files.[10]

After the dataset is converted into a clean dataset, the dataset is divided into training and testing sets so as to evaluate. Here, the training values are taken as the more recent values. The data to be tested is kept as 5-10 percent of the total dataset.

Stage 3: Feature Extraction:

In this layer, only the features that are to be fed to the neural network are selected. The feature will be chosen from date, open, high, low, close, and volume.

Stage 4: Training Neural Network:

In this stage, the data is fed to the neural network and trained for prediction assigning random biases and weights. Our LSTM model constitutes of a sequential input layer followed by 2 LSTM layers and dense layer with ReLU activation and then finally a dense output layer with linear activation function.[10]

V. SYSTEM DESIGN

This model is a Python tool for stock exploration Its capabilities can be used to identify pattern of data change and study overall stock trend. It can also analyze frequent deflections in a particular commodity. The main USP of this model is that it can predict future stock values with almost 80% accuracy. The use of LTSM as a machine learning algorithm is a major module of this model. The data from various data marts .

VI. OVERALL SYSTEM ARCHITECTURE

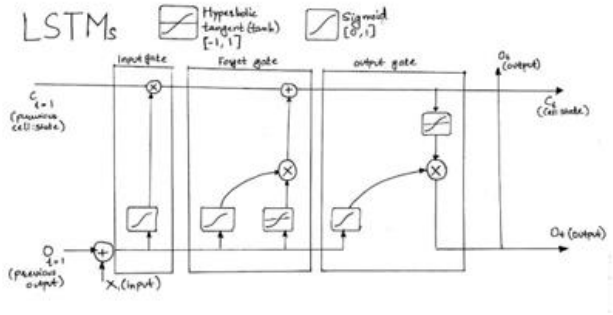


Figure 1 An LSTM memory cell

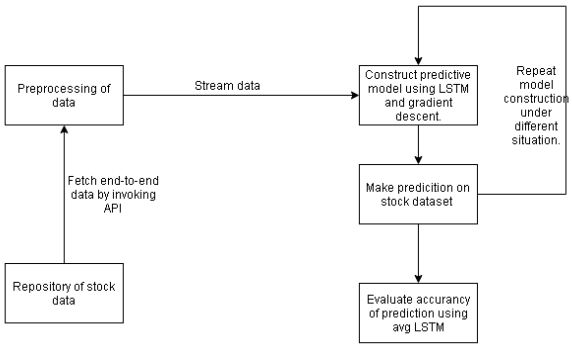


Figure 2 Architecture

The stock prediction system depicted in Figure has three main components. A brief explanation of each is given below:

1. Obtaining dataset and preprocessing

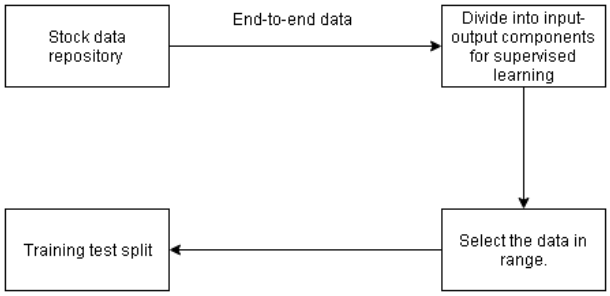


Figure 3 Data preprocessing

Main stock market data (for end-of-day prices of various companies) was obtained from two primary sources: Yahoo Finance and Google Finance. These two websites offer

URL-based APIs from which historical stock data for various companies can be obtained for various companies by simply specifying some parameters in the URL.

The obtained data contained five features:

1. Date: of the observation
2. Opening price: of the stock
3. High: highest intra-day price reached by the stock
4. Low: lowest intra-day price reached by the stock
5. Volume: number of shares or contracts bought and sold in the market during the day
6. OpenInt i.e., Open Interest: how many futures contracts are currently outstanding in the market

In [1]:

	Date	Open	High	Low	Close	Volume	Ex-Dividend	Split Ratio	Adj. Open	Adj. High	Adj. Low	Adj. Close	Adj. Volume	ds	y	Daily Change
0	1986-03-13	25.50	29.25	25.5	28.00	3582600.0	0.0	1.0	0.058941	0.067609	0.058941	0.064720	1.031789e+09	1986-03-13	0.064720	0.005779
1	1986-03-14	28.00	29.50	28.0	29.00	1070000.0	0.0	1.0	0.064720	0.068187	0.064720	0.067031	3.081600e+08	1986-03-14	0.067031	0.002311
2	1986-03-17	29.00	29.75	29.0	29.50	462400.0	0.0	1.0	0.067031	0.068765	0.067031	0.068187	1.331712e+08	1986-03-17	0.068187	0.001156
3	1986-03-18	29.50	29.75	28.5	28.75	235300.0	0.0	1.0	0.068187	0.068765	0.065876	0.066454	6.776640e+07	1986-03-18	0.066454	-0.001734
4	1986-03-19	28.75	29.00	28.0	28.25	166300.0	0.0	1.0	0.066454	0.067031	0.064720	0.065298	4.789440e+07	1986-03-19	0.065298	-0.001156

Figure 4 Dataset

The above data was then transformed into a format suitable for use with our prediction model by performing the following steps:

1. Transformation of time-series data into input-output components for supervised learning
2. Scaling the data to [-1, +1] range.

2.Construction of prediction model

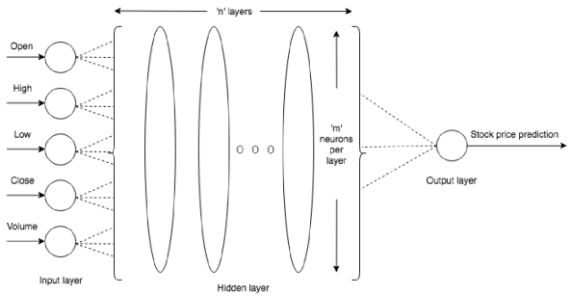


Figure 5 Prediction model

The input data is split into training and test datasets; our LSTM model will be fit on the training dataset, and the accuracy of the fit will be evaluated on the test dataset. The LSTM network is constructed with one input layer having five neurons, 'n' hidden layers (with 'm' LSTM memory cells per layer), and one output layer (with one neuron). After fitting the model on the training dataset, hyper-parameter tuning is done using the validation set to choose the optimal values of parameters such as the number of hidden layers 'n', number of neurons 'm' per hidden layer, batch size, etc.

3. Apply LSTM Algorithm [11] :-

Algorithm 1: LSTM stock prediction algorithm

Input: Historical stock price data

Output: Prediction for stock prices based on stock price variation

1. Start
2. Stock data is taken and stored in a numpy array of 3 dimensions (N,W,F) where :
   - N is number of training sequences,
   - W is sequence length
   - F is the number of features of each sequence.
3. A network structure is built with [1,a,b,1] dimensions, where there is 1 input layer, *a* neurons in the next layer, *b* neurons in the subsequent layer, and a single layer with a linear activation function.
4. Train the constructed network on the data
5. Use the output of the last layer as prediction of the next time step.
6. Repeat steps 4 and 5 until optimal convergence is reached.
7. Obtain predictions by providing test data as input to the network.
8. Evaluate accuracy by comparing predictions made with actual data.
9. End

4. Predictions and accuracy

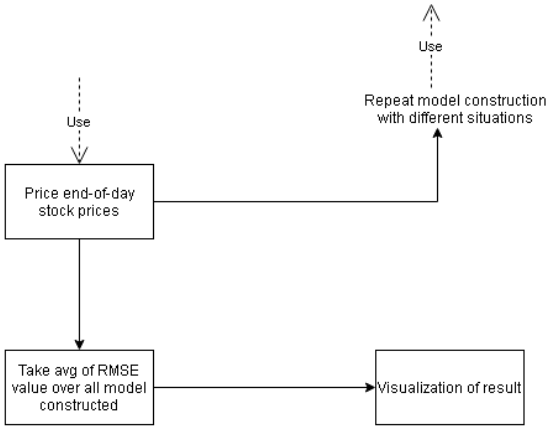


Figure 6 RMSE flow

Once the LSTM model is fit to the training data, it can be used to predict the end-of-day stock price of an arbitrary stock.

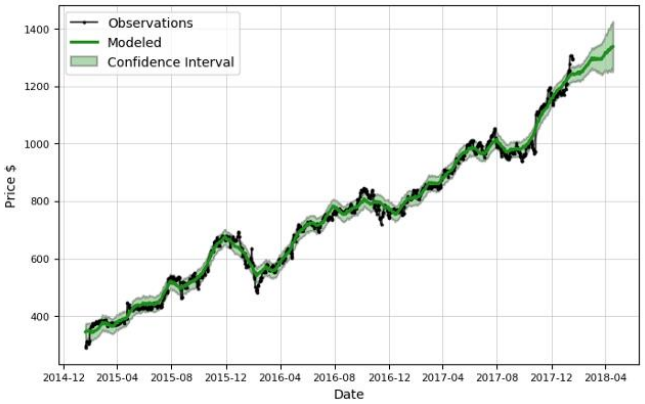
This prediction can be performed in two ways:

1. Static – a simple, less accurate method where the model is fit on all the training data. Each new time step is then predicted one at a time from test data.
2. Dynamic – a complex, more accurate approach where the model is refit for each time step of the test data as new observations are made available.

The accuracy of the prediction model can then be estimated robustly using the RMSE (Root Mean Squared Error) metric. This is due to the fact that neural networks in general (including LSTM) tend to give different results with different starting conditions on the same data.

We then repeat the model construction and prediction several times (with different starting conditions) and then take the average RMSE as an indication of how well our configuration would be expected to execute on unseen real-world stock data . That is, our predictions will be compared with actual trends in stock price movement that can be inferred from historical data.

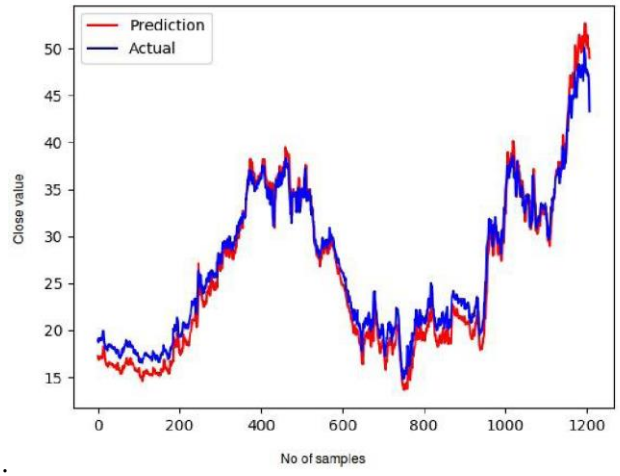
VII. VISUALIZATION OF RESULTS



.Figure 7 Visualize

In this prediction, the green line, indicates a confidence interval. This represents the model's uncertainty in the forecast. In this case, the confidence interval width is set at 80%, meaning we expect that this range will contain the actual value 80% of the time. As the time increases, the confidence interval grows wide further because the predicted value has more uncertainty as it gets further away from the data. Any time we make a prediction we must include a confidence interval so that it can be evaluated for accuracy.

It shows the actual and the predicted closing stock price of the company Alcoa Corp, a large-sized stock. Our model was trained with a batch size of 512 and 50 epochs, and the predictions made closely matched the real stock prices, as observed in the graph



.

Figure 8.1

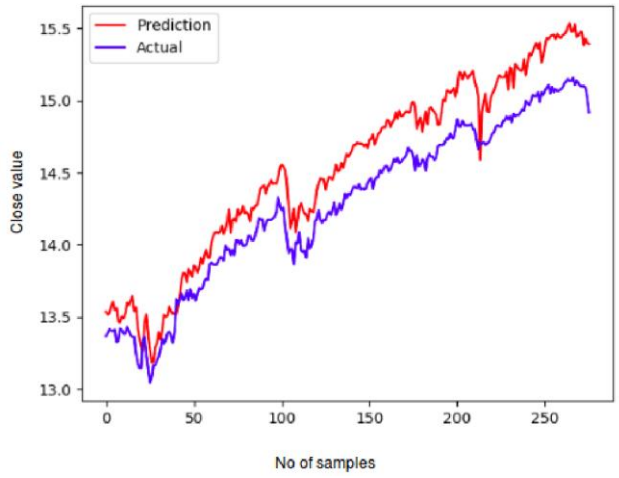


Figure 8.2

VIII. CONCLUSION

The popularity of stock market is growing rapidly, which is encouraging traders, brokers and researchers to find out new methods for the prediction of future stock trends. This forecasting techniques are not only helping the researchers but it also helps stock brokers and any person dealing with the stock market. In order to predict the stock indices, a system model with good accuracy is required.

REFERENCES

[1] Ding, Y., Zhao, P., Hoi, S. C., Ong, Y. S. "An Adaptive Gradient Method for Online AUC Maximization" In AAAI (pp. 2568-2574). (2015, January).

[2] Zhao, P., Hoi, S. C., Wang, J., Li, B. "Online transfer learning". Artificial Intelligence, 216, 76-102. (2014)
[3] Nazar, Nasrin Banu, and Radha Senthilkumar. "An online approach for feature selection for classification in big data." Turkish Journal of Electrical Engineering & Computer Sciences 25.1 (2017): 163-171.
[4] Kai Chen, Yi Zhou and Soulas, Eleftherios, and Dennis Shasha. "Online machine learning algorithms for currency exchange prediction." Computer Science Department in New York University, Tech. Rep 31 (2013).
[5] Pascanu, Razvan, Tomas Mikolov, and Yoshua Bengio. "On the difficulty of training recurrent neural networks." International Conference on Machine Learning. 2013.
[6] Suresh, Harini, et al. "Clinical Intervention Prediction and Understanding using Deep Networks." arXiv preprint arXiv:1705.08498 (2017).
[7] Recht, Benjamin, et al. "Hogwild: A lock-free approach to parallelizing stochastic gradient descent." Advances in neural information processing systems. 2011.
[8] Zhu, Maohua, et al. "Training Long Short-Term Memory With Sparsified Stochastic Gradient Descent." (2016)
[9] Ruder, Sebastian. "An overview of gradient descent optimization algorithms." arXiv preprint arXiv:1609.04747.(2016).
[10] Aniruddha Choudhury "Stock Market Prediction by Recurrent Neural Network on LSTM Model "
[11] Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". *Neural Computation*. **9** (8): 1735–1780.