STOCK MARKET PRICE PREDICATION USING MACHINE LEARNING

Project report submitted in partial fulfilment of Industry Oriented Mini Project

for

VII semester, B. Tech in Computer Science and Engineering

By

B. YASHASWINI (18135A0505)

B. SAI KIRAN (18135A0507)

B. MANJUNADH (18135A0508)

Under the Esteemed Guidance of

K . Suma Sree , [M. Tech]

Assistant Professor

Computer Science Dept.

GVPCOE(A)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING (Autonomous)

Approved by AICTE, New Delhi and Affiliated to JNTU-Kakinada
Re-accredited by NAAC with "A" Grade with a CGPA of 3.47/4.00
Madhurawada, Visakapathanam-530 048

CERTIFICATE

This is to certify that the project report entitled **" Stock market price prediction using machine learning "** being submitted by

B. YASHASWINI (18135A0505)

B. SAI KIRAN (18135A0507)

B. MANJUNADH (18135A0508)

in partial fulfilment for the mini project in Computer Science and Engineering to the Jawaharlal Nehru Technological University, Kakinada is a record of bona fide work carried out under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

PROJECT GUIDE:

Mrs. Suma Sree
Assistant Professor
GVPCE(A)

HEAD OF DEPARTMENT:

DR. P. Krishna Subba Rao
Professor
GVPCE(A)

ACKNOWLEDGEMENT

We thank **Prof. A. B. K. Rao** principal, **Gayatri Vidya Parishad College of Engineering(A)** for extending his utmost support and cooperation in providing all the provisions for the successful completion of the project.

We consider it our privilege to express our deepest gratitude to **Dr. P.K. Subba Rao** Professor, Head of the Department, Computer Science and Engineering, for his valuable suggestions and constant motivation that greatly helped the project work to get successfully completed.

Thank you, Sir.

We also thank all the members of the staff in Computer Science Engineering for their sustained help in our pursuits.

We thank all those who contributed directly or indirectly in successfully carrying out his work.

By,

B. Yashaswini (18135A0505)

B. Sai kiran (18135A0507)

B. Manjunadh (18135A0508)

ABSTRACT

Stock Market Price Prediction Using Machine Learning

Predicting stock market prices is a complex task that traditionally involves extensive human-computer interaction. Due to the correlated nature of stock prices, conventional batch processing methods cannot be utilized efficiently for stock market analysis. Stock price prediction is one among the complex machine learning problems. It depends on a large number of factors which contribute to changes in the supply and demand. In Stock Market Prediction, the aim is to predict the future value of the financial stocks of a company. The recent trend in stock market prediction technologies is the use of machine learning which makes predictions based on the values of current stock market indices by training on their previous values. Machine learning itself employs different models to make prediction easier and authentic. Stock prices are represented as time series data and neural networks are trained to learn the patterns from trends. This paper focuses on the use learning algorithm that utilizes a kind of recurrent neural network (RNN) called Long Short Term Memory (LSTM) to predict stock values. This will provide more accurate results when compared to existing stock price prediction algorithms. The network is trained and evaluated for accuracy with various sizes of data, and the results are tabulated.

INDEX

Contents :

1. INTRODUCTION

Stock price is the price of a single stock among the number of stocks sold by a company listed in public offering. Having stocks of a public company allows you to own a portion of it. Original owners of the company initially sell the stocks to get additional investment to help the company grow. This initial offering of stocks to the public is called Initial Public Offering (IPO). Stock prices change because of the supply and demand. Suppose, if many people are willing to buy a stock, then the price goes up as there is more demand. If more people are willing to sell the stock, the price goes down as there is more supply than the demand. Though understanding supply and the demand is relatively easy, it is hard to derive what factors exactly contribute to the increase in demand or supply. These factors would generally boil down to socioeconomic factors like market behavior, inflation, trends and more importantly, what is positive about the company in the news and what's negative.

The stock market is a vast array of investors and traders who buy and sell stock, pushing the price up or down. The prices of stocks are governed by the principles of demand and supply, and the ultimate goal of buying shares is to make money by buying stocks in companies whose perceived value (i.e., share price) is expected to rise. Stock markets are closely linked with the world of economics.

Economics and stock prices are mainly reliant upon subjective perceptions about the stock market. It is near impossible to predict stock prices to the T, owing to the volatility of factors that play a major role in the movement of prices. However, it is possible to make an educated estimate of prices. Stock prices never vary in isolation: the movement of one tends to have an avalanche effect on several other stocks as well . This aspect of stock price movement can be used as an important tool to predict the prices of many stocks at once. Due to the sheer volume of money involved and number of transactions that take place every minute, there comes a trade-off between the accuracy and the volume of predictions made; as such, most stock prediction systems are implemented in a distributed, parallelized fashion . These are some of the considerations and challenges faced in stock market analysis.

Recurrent neural networks (RNN) have proved one of the most powerful models for processing sequential data.

Long Short-Term memory is one of the most successful RNNs architectures. LSTM introduces the memory cell, a unit of computation that replaces traditional artificial neurons in the hidden layer of the network. With these memory cells, networks are able to effectively associate memories and input remote in time, hence suit to grasp the structure of data dynamically over time with high prediction capacity.

2. LITERATURE SURVEY

Forecasting stock return is an important financial subject that has attracted researchers' attention for many years. It involves an assumption that fundamental information publicly available in the past has some predictive relationships to the future stock returns. In order to be able to extract such relationships from the available data, data mining techniques are new techniques that can be used to extract the knowledge from this data. For that reason, several researchers have focused on technical analysis and using advanced math and science.

Some models have been proposed and implemented using the above mentioned techniques, the authors of Tsang, P.M., Kwok ,P., Choy, S.O., Kwan, R., Ng, S.C., Mak, J., Tsang, J., Koong, K., and Wong, T. made an empirical study on building a stock buying/selling alert system using back propagation neural networks (BPNN), the empirical results showed that the implemented system was able to predict short-term price movement directions with accuracy about 74%.

The model of Wang, J.L., Chan, S.H. (2006) "Stock market trading rule discovery using two-layer bias decision tree", applied the concept of serial topology and designed a new decision system, namely the two layer bias decision tree, for stock price prediction.

The authors Enke, D., Thaworn wong, S. presented an approach that used data mining methods and neural networks for forecasting stock market returns. An attempt has been made in this study to investigate the predictive power of financial and economic variables by adopting the variable relevance analysis technique in machine learning for data mining. The authors examined the effectiveness of the neural network models used for level estimation and classification.

El - Baky et al., proposed a new approach for fast forecasting of stock market prices. The proposed approach uses new high speed time delay neural networks (HSTDNNs). The authors used the MATLAB tool to simulate results to confirm the theoretical computations of the approach.

3. SOFTWARE REQUIREMENTS ANALYSIS

3.1 **FUNCTIONAL REQUIREMENTS**

3.1.1 **NUMPY**

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array. Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Num array was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project.

Operations using NumPy

Using NumPy, we can perform the following operations:

1. Mathematical and logical operations on arrays.
2. Fourier transforms and routines for shape manipulation.
3. Operations related to linear algebra. NumPy has in-built functions for linear algebra and random number generation.

ND Array Object

The most important object defined in NumPy is an N-dimensional array type called nd array. It describes the collection of items of the same type. Items in the collection can be accessed using a zero-based index.

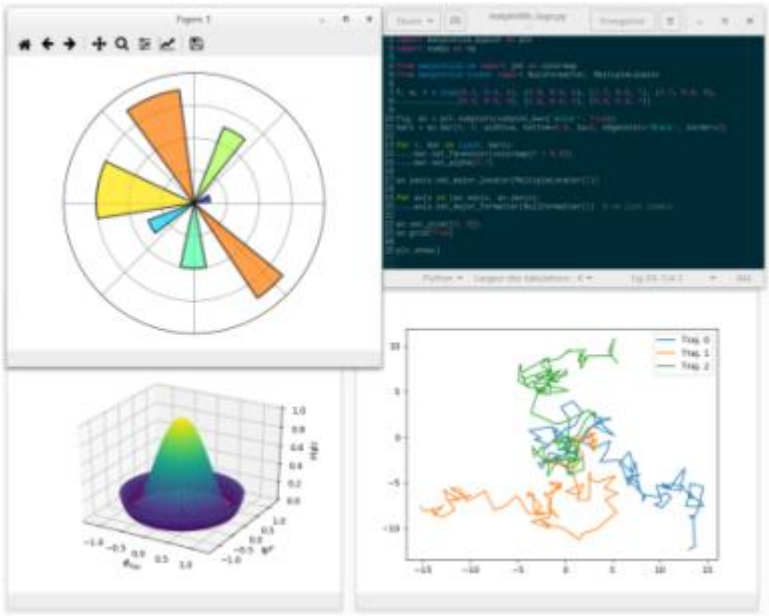
Every item in a nd array takes the same size of block in the memory. Each element in nd array is an object of data-type object (called dtype).

The basic nd array is created using an array function in NumPy as follows:

SYNTAX: numpy.array()

3.1.2 **MATPLOT LAB :**

Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.



3.1.3 **PANDAS :**

Pandas has been one of the most popular and favourite data science tools used in Python programming language for data wrangling and analysis. Data is unavoidably messy in real world. And **Pandas** is seriously a game changer when it comes to cleaning, transforming, manipulating and analyzing data.

3.1.4 **SKLEARN :**

Scikit-learn is one the most popular ML **libraries**. It supports many supervised and unsupervised **learning** algorithms. ... It adds a set of algorithms for common **machine learning** and data mining tasks, including clustering, regression and classification.

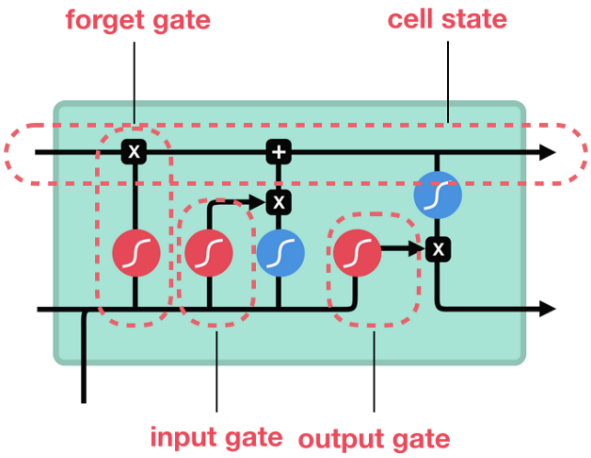
3.1.5 **KERAS :**

Keras is a model-level library, providing high-level building blocks for developing deep learning models. It does not handle itself low-level operations such as tensor products, convolutions and so on. Instead, it relies on a specialized, well-optimized tensor manipulation library to do so, serving as the "backend engine" of Keras. Rather than picking one single tensor library and making the implementation of Keras tied to that library, Keras handles the problem in a modular way, and several different backend engines can be plugged seamlessly into Keras.

3.1.6 **LSTM**

LSTM is a special type of RNN. These networks are proficient in learning about long-term dependencies. They are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!. Long Short-Term Memory models are extremely powerful time-series models. They can predict an arbitrary number of steps into the future. LSTMs overcame the problem of RNNs struggling with handling “Long term dependencies”. The Core idea of LSTM is A memory cell which can maintain its state over time, consisting of an explicit memory and gating units which regulate the information flow into and out of the memory.

Architecture of LSTM Cell



The architecture is composed of a **cell** (the memory part of the LSTM unit) and three "regulators", usually called gates, of the flow of information inside the LSTM unit:

- input gate
- output gate
- forget gate

3.1.6.1 **Input Gate**

To update the cell state, we have the input gate. First, we pass the previous hidden state and current input into a sigmoid function. That decides which values will be updated by transforming the values to be between 0 and 1. 0 means not important, and 1 means important. You also pass the hidden state and current input into the tanh function to squish values between -1 and 1 to help regulate the network. Then you multiply the tanh output with the sigmoid output. The sigmoid output will decide which information is important to keep from the tanh output.

3.1.6.2 **Output Gate**

The output gate decides what the next hidden state should be. The hidden state is used for predictions. First, we pass the previous hidden state and the current input into a sigmoid function. Then we pass the newly modified cell state to the tanh function. We multiply the tanh output with the sigmoid output to decide what information the hidden state should carry. The output is the hidden state. The new cell state and the new hidden is then carried over to the next time step

3.1.6.3 **Forget Gate**

This gate decides what information should be thrown away or kept. Information from the previous hidden state and information from the current input is passed through the sigmoid function. Values come out between 0 and 1. The closer to 0 means to forget, and the closer to 1 means to keep.

Thus, LSTM networks are ideal for exploring how variation in one stock's price can affect the prices of several other stocks over a long period of time. They can also decide (in a dynamic fashion) for how long information about specific past trends in stock price movement needs to be retained in order to more accurately predict future trends in the variation of stock prices.

Advantages of LSTM

The main advantage of an LSTM is its ability to learn context specific temporal dependence. Each LSTM unit remembers information for either a long or a short period of time (hence the name) without explicitly using an activation function within the recurrent components. An important fact to note is that any cell state is multiplied only by the output of the forget gate, which varies between 0 and 1. That is, the forget gate in an LSTM cell is responsible for both the weights and the activation function of the cell state. Therefore, information from a previous cell state can pass through a cell unchanged instead of increasing or decreasing exponentially at each time-stepor layer, and the weights can converge to their optimal values in a reasonable amount of time. This allows LSTM's to solve the vanishing gradient problem – since the value stored in a memory cell isn't iteratively modified, the gradient does not vanish when trained with backpropagation.

Additionally, LSTM's are also relatively insensitive to gaps (i.e., time lags between input data points) compared to other RNN's.

3.2 **METHODOLOGY**

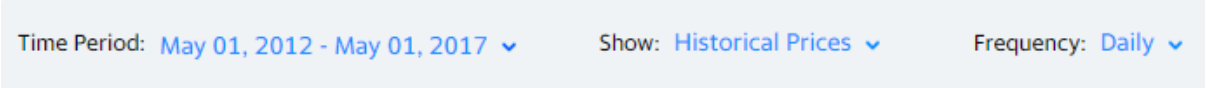
Various types of neural networks can be developed by the combination of different factors like network topology, training method etc. For this experiment, we have considered Recurrent Neural Network and Long Short-Term Memory.

We operate on the Stock of Google Inc.

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 of google is collected from https://finance.yahoo.com/quote/GOOG/history?p=GOOG and this historical data is used for the prediction of future stock prices. We have collected data from 2012 to 2015



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.

So, the final obtained data looks like this

	Date	Open	High	Low	Close	Volume
0	01-03-2012	325.25	332.83	324.97	663.59	73,80,500
1	01-04-2012	331.27	333.87	329.08	666.45	57,49,400
2	01-05-2012	329.83	330.75	326.89	657.21	65,90,300
3	01-06-2012	328.34	328.77	323.68	648.24	54,05,900
4	01-09-2012	322.04	322.29	309.46	620.76	1,16,88,800
...	...	...	...	...	...	...
1273	1/25/2017	829.62	835.77	825.06	835.67	14,94,500
1274	1/26/2017	837.81	838.00	827.01	832.15	29,73,900
1275	1/27/2017	834.71	841.95	820.44	823.31	29,65,800
1276	1/30/2017	814.66	815.84	799.80	802.32	32,46,600
1277	1/31/2017	796.86	801.25	790.52	796.79	21,60,600

After the dataset is transformed 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. Testing data is kept as 5-10 percent of the total dataset.

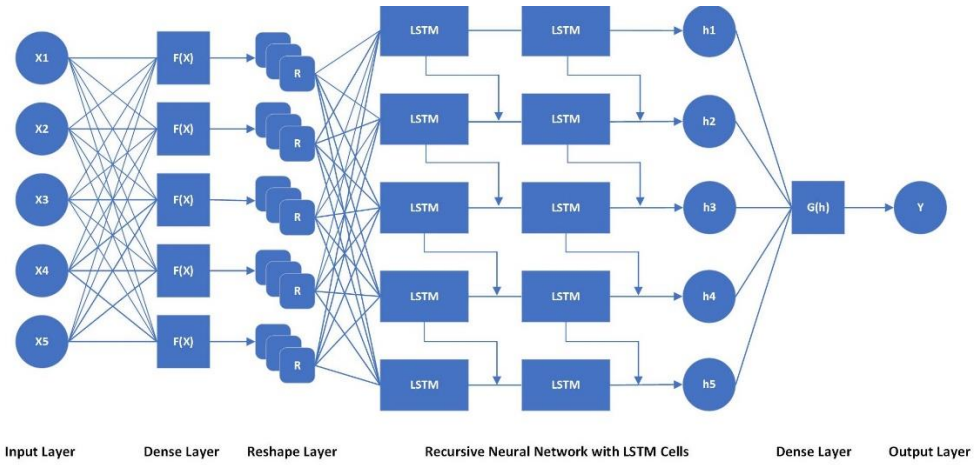
Stage 3 : Feature Extraction

In this layer, only the features which are to be fed to the neural network are chosen. We will choose the feature from Date, open, high, low, close, and volume.

We chose Open feature to work with LSTM's

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 is composed of a sequential input layer followed by 4 LSTM layers and dense layer with sigmoid activation and then finally a dense output layer with linear activation function.



Stage 5 : Output Generation

In this layer, the output value generated by the output layer of the RNN is compared with the target value. The error or the difference between the target and the obtained output value is minimized by using back propagation algorithm which adjusts the weights and the biases of the network.

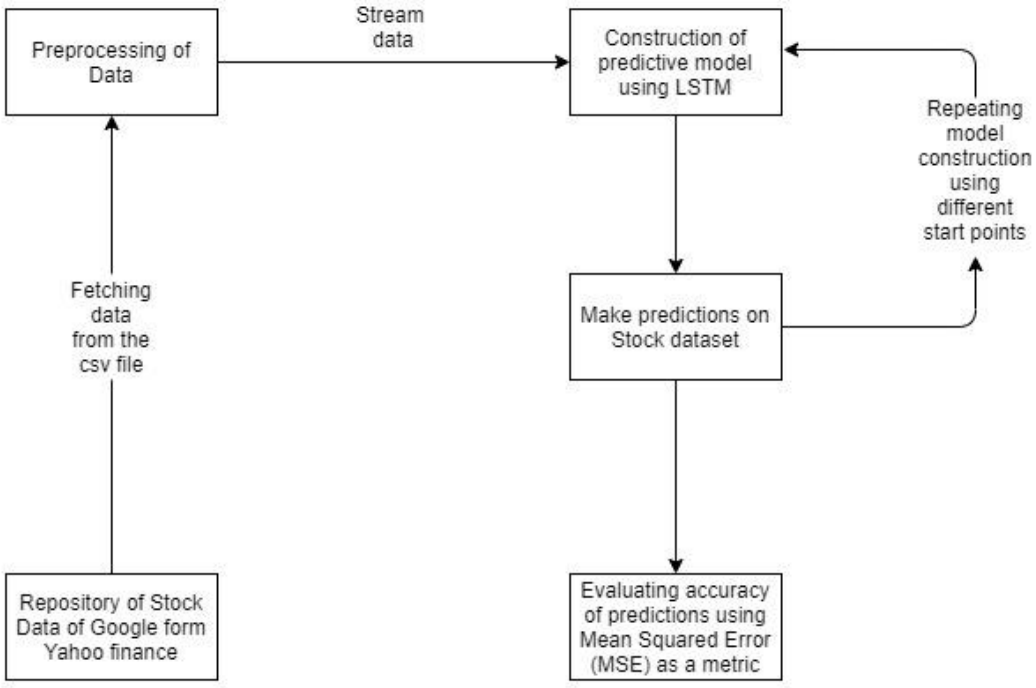
ANAYSIS

For analyzing the efficiency of the system, we are used the Mean Squared Error (MSE). The error or the difference between the target and the obtained output value is minimized by using MSE value. MSE is the mean/average of the square of all of the error. The use of MSE is highly common and it makes an excellent general purpose error metric for numerical predictions. MSE amplifies and severely punishes large errors.

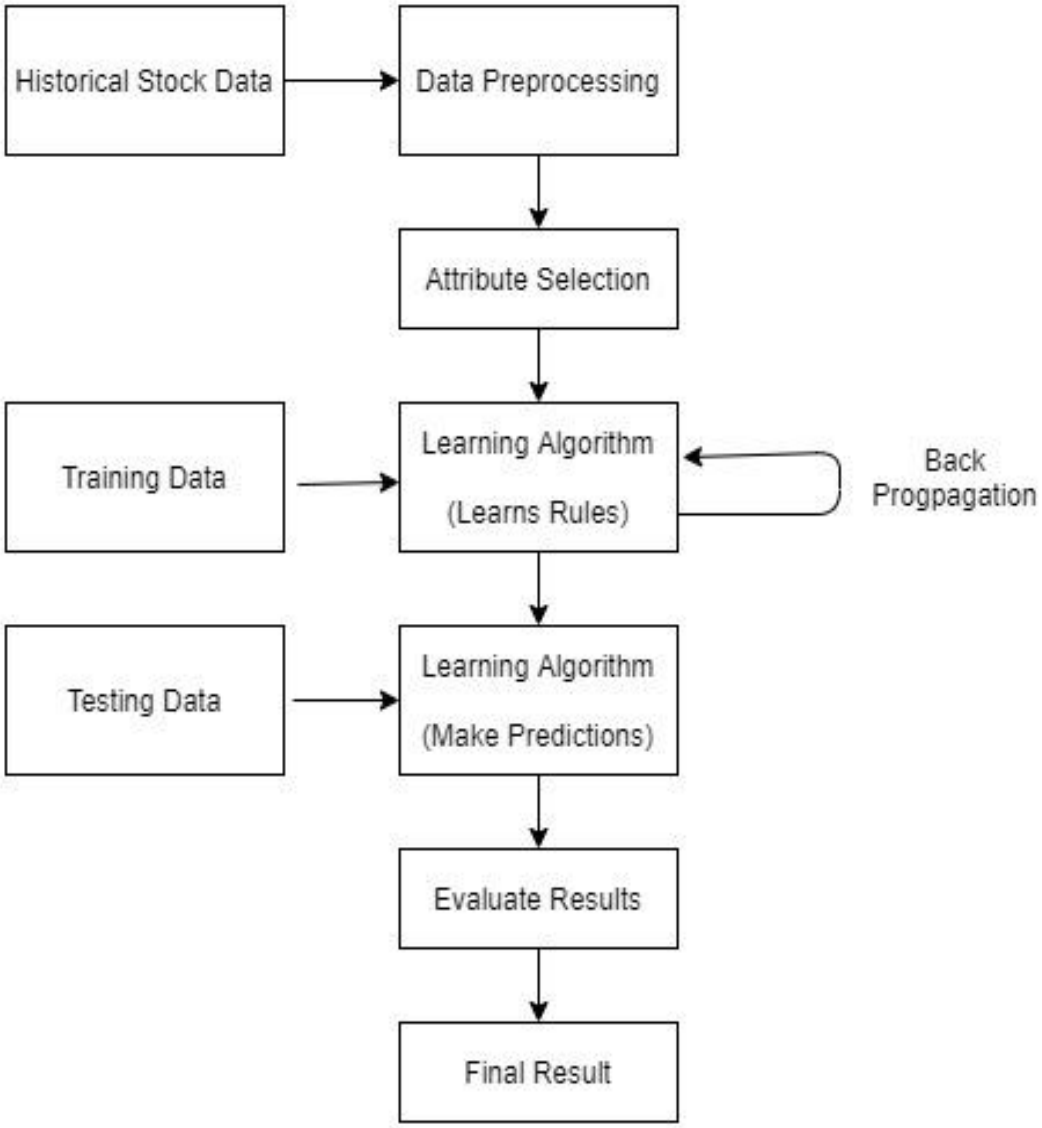
4 . SOFTWARE DESIGN

4.1 **Architecture Design:**

The below figure gives an overview of the Stock price prediction model. The system first fetches data from the repository of Stock Data of Google from yahoo finance. The data obtained may contain many unimportant and null values, the data obtained is preprocessed and is sent the LSTM model, In the LSTM model, the data is trained over many epochs, the predicted data is compared with actual data and the weights of the nodes in the network are updated accordingly, thus creating a neural network capable of predicting



4.2 **Control Flow Diagram**



5. TESTING

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

5.1 **Purpose of Testing**

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn't work. Testing is the process of executing a program with the intent of finding errors.

5.1.1 **Testing Objectives**:

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, we can say,

• Testing is a process of executing a program with the intent of finding an error.

• A successful test is one that uncovers an as yet undiscovered error.

• A good test case is one that has a high probability of finding error, if it exists.

• The software more or less confirms to the quality and reliable standards.

5.2 **Levels of Testing**

In order to uncover the errors, present in different phases we have the concept of levels of testing. The basic levels of testing are as shown below...

5.2.1 **System Testing**

The philosophy behind testing is to find errors. Test cases are devised with this in mind. A strategy employed for system testing is code testing.

5.2.2 **Code Testing**

This strategy examines the logic of the program. To follow this method we developed some test data that resulted in executing every instruction in the program and module i.e. every path is tested. Systems are not designed as entire nor are they tested as single systems. To ensure that the coding is perfect two types of testing is performed or for that matter is performed on all systems.

5.2.3 **White Box Testing**

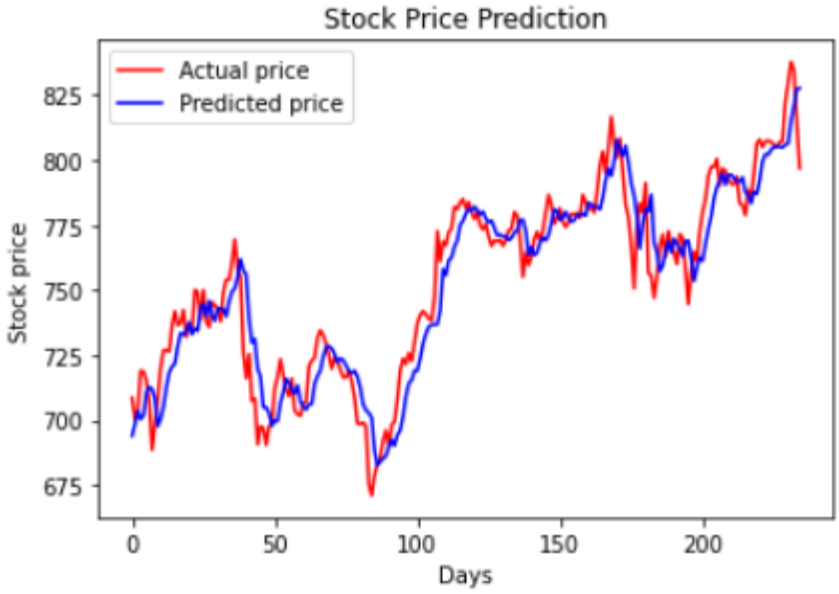
This is a unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors. I tested step wise every piece of code, taking care that every statement in the code is executed at least once. The white box testing is also called Glass Box Testing. I have generated a list of test cases, sample data. which is used to check all possible combinations of execution paths through the code at every module Level.

5.2.4 **Black Box Testing**

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details at statement level. Here the module will be treated as a block box that will take some input and generate output. Output for a given set of input combinations are forwarded to other modules.

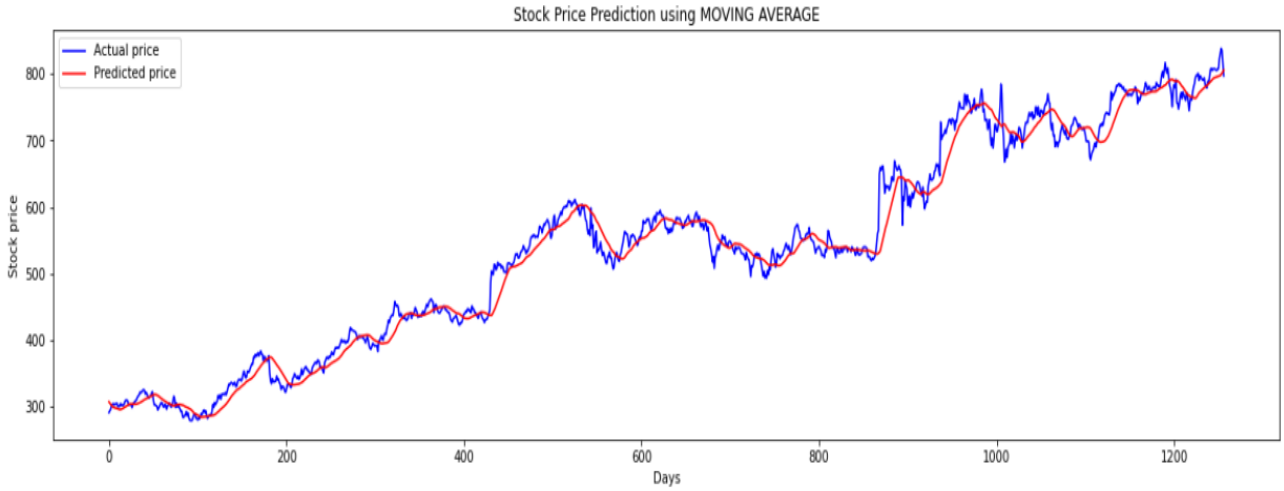
6. OUTPUT SCREENSHOTS

6.1 **Visualization of TEST DATA by LSTM**



6.1.1 **MSE Value of LSTM :** 9.975059700162928

6.2 **Visualization of DATA by Moving Average**



6.2.1 **MSE Value of Moving Average :** 21.164953903476306

7. CONCLUSION

The popularity of stock market trading is growing rapidly, which is encouraging researchers to find out new methods for the prediction using new techniques. The forecasting technique is not only helping the researchers but it also helps investors and any person dealing with the stock market. In order to help predict the stock indices, a forecasting model with good accuracy is required.

In this work, we have used one of the most precise forecasting technology using Recurrent Neural Network and Long Short-Term Memory unit which helps investors, analysts or any person interested in investing in the stock market by providing them a good knowledge of the future situation of the stock market.

8. FUTURE ENHANCEMENTS

Several future enhancements can be made, by increasing the richness of the data and the size of the dataset we might be able to predict the stock prices at a greater precision. There is also good scope for new algorithms for predicting time series forecasting which may help in predicting stock prices. Using bigger dataset can improve the output quality of predicted stock price.

9. REFERENCES

IRJET paper of “Stock price prediction using Long Short Term Memory” by Raghav Nanda Kumar, Uttamraj K, Vishal R, Y V Lokeshwari

International Journal on “Stock Market Forecasting Techniques” by Vivek Rajput, Sarika Bobde

https://towardsdatascience.com/predicting-stock-price-with-lstm-13af86a74944

https://www.researchgate.net/publication/328930285_Stock_Market_Prediction_Using_Machine_Learning