INTERNSHIP REPORT

PROJECT TITLE: STOCK PRICE PREDICTION

A Report Submitted to
Jawaharlal Nehru Technological University Kakinada, Kakinadain partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

P. NAISHID

(20KN1A05C1)

Under the esteemed guidance of

M.Rahul

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NRI INSTITUTE OF TECHNOLOGY

Autonomous

(Approved by AICTE, Permanently Affiliated to JNTUK, Kakinada) Accredited by NBA (CSE, ECE & EEE), Accredited by NAAC with 'A' GradeISO 9001: 2015 Certified Institution

Pothavarappadu (V), (Via) Nunna, Agiripalli (M), Krishna Dist., PIN: 521212, A.P, India.

2022-2023

NRI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution, Approved by AICTE, Permanently Affiliated to JNTUK, Kakinada)

Accredited by NBA (CSE, ECE & EEE), Accredited by NAAC with ‘A’ Grade
ISO 9001: 2015 Certified Institution

Pothavarappadu (V), (Via) Nunna, Agiripalli (M), Krishna Dist., PIN: 521212, A.P, India.

CERTIFICATE

This is to certify that the “**Internship report**” submitted by P. NAISHID (**20KN1A05C1**) is work done by him and submitted during YEARS academic year, in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY inCOMPUTER SCIENCE AND ENGINEERING,** at **BLACKBUCK ENGINEERS PVT LTD, Road No:36, Jubilee Hills, Hyderabad, Telangana.**

Kathyayani

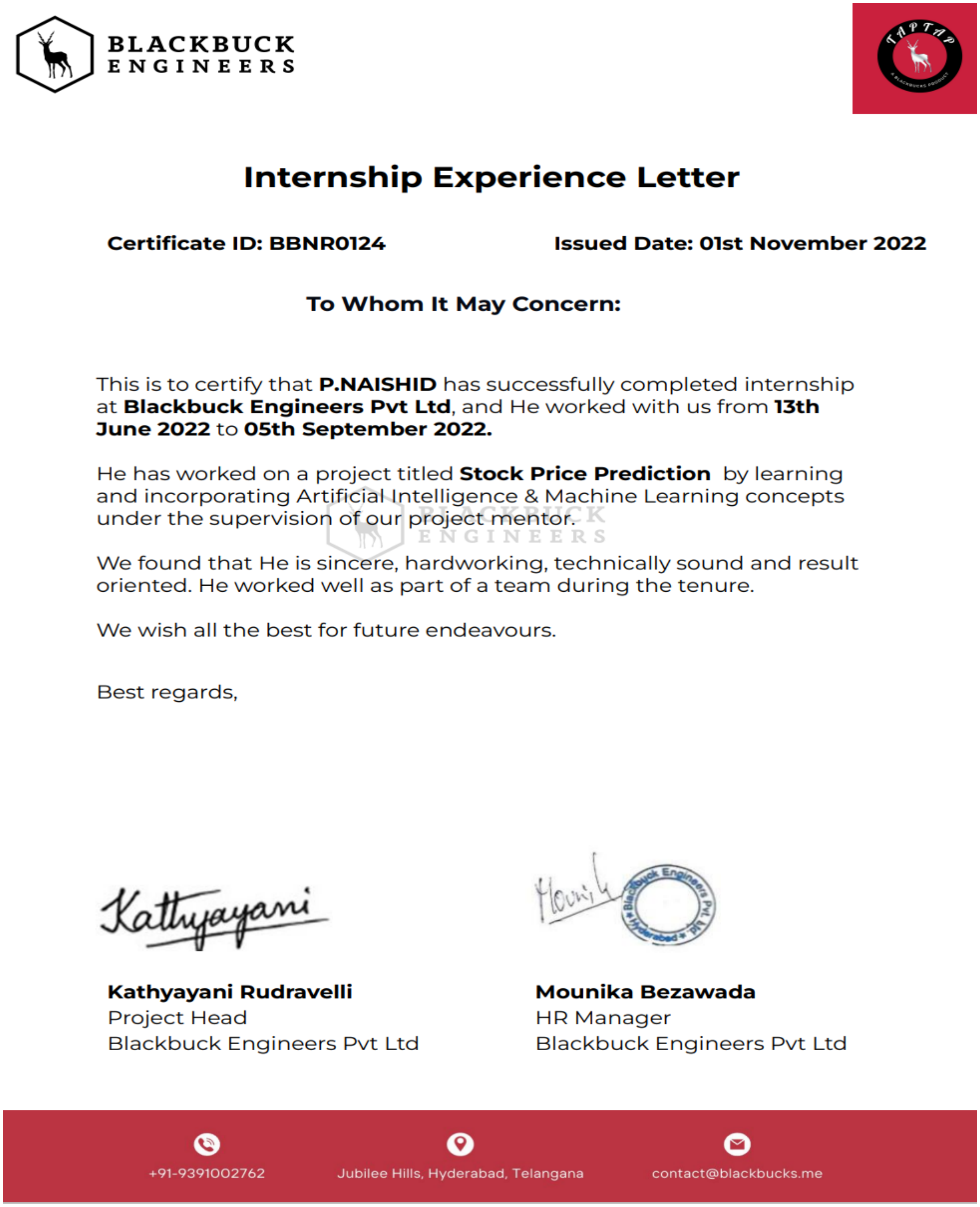
INTERNSHIP COORDIANTOR

Head of the Department

(Dr. D. SUNEETHA)

EXTERNAL EXAMINER

CERTIFICATE OF INTERNSHIP



BLACKBUCK ENGINEERS

TAPTAP

A BLACKBUCKS PRODUCT

Internship Experience Letter

Certificate ID: BBNR0124 **Issued Date: 01st November 2022**

To Whom It May Concern:

This is to certify that **P.NAISHID** has successfully completed internship at **Blackbuck Engineers Pvt Ltd**, and He worked with us from **13th June 2022** to **05th September 2022.**

He has worked on a project titled **Stock Price Prediction** by learning and incorporating Artificial Intelligence & Machine Learning concepts under the supervision of our project mentor.

We found that He is sincere, hardworking, technically sound and result oriented. He worked well as part of a team during the tenure.

We wish all the best for future endeavours.

Best regards,

Kathyayani

Mounika

Blackbuck Engineers Pvt. Ltd • Hyderabad •

Kathyayani Rudravelli
Project Head
Blackbuck Engineers Pvt Ltd

Mounika Bezawada
HR Manager
Blackbuck Engineers Pvt Ltd

☎ +91-9391002762 Jubilee Hills, Hyderabad, Telangana ✉ contact@blackbucks.me

ACKNOWLEDGEMENT

I take this opportunity to thank all who have rendered their full support to my work. The pleasure, the achievement, the glory, the satisfaction, the reward, the appreciation and the construction of my project cannot be expressed with a few words for their valuable suggestions.

I am expressing my heartfelt thanks to **Head of the Department, Dr. D. SUNEETHA** garu for her continuous guidance for completion of my Project work.

I am extending our sincere thanks to **Dean of the Department, Dr. K. V. SAMBASIVA RAO** for his continuous guidance and support to complete my project successfully.

I am thankful to the **Principal, Dr. C. NAGA BHASKAR** garu for his encouragement tocomplete the Project work.

I am extending my sincere and honest thanks to the **Chairman, Dr. R. VENKATA RAOgaru & Secretary, Sri K. Sridhar** garu for their continuous support in completing the Project work.

Finally, I thank the Administrative Officer, Staff Members, Faculty of Department of CSE, NRI Institute of Technology and my friends, directly or indirectly helped us in the completion of this project.

P. NAISHID
(20KN1A05C1)

ABSTRACT

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. Useful prediction systems allow traders to get better insights about data such as: future trends. Also, investors have a major benefit since the analysis give future conditions of the market. The objective is to predict the stock prices in order to make more informed and accurate investment decisions. We propose a stock price prediction system that integrates mathematical functions, machine learning, and other external factors for the purpose of achieving better stock prediction accuracy and issuing profitable trades. In this work we use Machine learning architectures Long Short-Term Memory (LSTM) for the price forecasting of NSE listed companies and differentiating their performance. The previous price of a stock is crucial in predicting its future price. While predicting the actual price of a stock is an uphill climb, we can build a model that will predict whether the price will go up or down.

Organization Information:

Blackbuck Engineers is started in 2013 with the aim of creating a great ecosystem of academia, research, industry, and individuals. Blackbucks is a premier partner to Govts International Institute of Digital Technologies, and IITs. Blackbuck delivers the TAPTAP AI Driven employability platform to transform the journey of students towards their dream goals while helping **HRs hire right students**.

Blackbuck has the largest chain of excellence in emerging tech across India.

Blackbucks runs post-graduation programs in AI, ML and Data Science

www.theblackbucks.com

Programs and opportunities:

This ground-up approach helps us deliver not only the solution to our clients but also add value to at the core Blackbuck Engineers which operates in Five specific domains namely TapTap - AI Driven, Post-Graduation Programs, Center of Excellence, Virtual Programming Labs and Happie Days - A social Networking site for the students. TapTap offer services in Campus Recruitment drives for the Employers as well as College authorities. Recruiters can Conduct Customized Online Assessments secured with Best-in-class Proctoring and Schedule the end-to-end hiring process. Under each division we further provide specific industry solutions on focused domains with cutting edge technologies. Blackbuck Engineers emphasize on building relationships with our clients by delivering projects on time and within budget

Learning Objectives/Internship Objectives

- Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships in order to receive real world experience and develop their skills.

- An objective for this position should emphasize the skills you already possess in the area and your interest in learning more

- Internships are utilized in a number of different career fields, including architecture, engineering, healthcare, economics, advertising and many more.

- Some internship is used to allow individuals to perform scientific research while others are specifically designed to allow people to gain first-hand experience working.

- Utilizing internships is a great way to build your resume and develop skills that can be emphasized in your resume for future jobs. When you are applying for a Training Internship, make sure to highlight any special skills or talents that can make you stand apart from the rest of the applicants so that you have an improved chance of landing the position.

WEEKLY OVERVIEW OF INTERNSHIP ACTIVITIES

1st WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	13.06.2022	Monday	Introduce the Topic & the Problem Statement
	14.06.2022	Tuesday	Introduce the Topic & the Problem Statement
	15.06.2022	Wednesday	Introduce the Topic & the Problem Statement
	16.06.2022	Thursday	Introduce the Topic & the Problem Statement
	17.06.2022	Friday	Introduce the Topic & the Problem Statement

2nd WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	20.06.2022	Monday	Abstract Building
	21.06.2022	Tuesday	Abstract Building
	22.06.2022	Wednesday	Abstract Building
	23.06.2022	Thursday	Abstract Building
	24.06.2022	Friday	Abstract Submission

3rd WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	27.06.2022	Monday	Abstract Submission
	28.06.2022	Tuesday	Abstract Submission
	29.06.2022	Wednesday	Explain your Approach to Solving Problem
	30.06.2022	Thursday	Explain your Approach to Solving Problem
	01.07.2022	Friday	Explain your Approach to Solving Problem

4th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	04.07.2022	Monday	Explain your Approach to Solving Problem
	05.07.2022	Tuesday	Explain the Structure of the Project
	06.07.2022	Wednesday	Explain the Structure of the Project
	07.07.2022	Thursday	Explain the Structure of the Project
	08.07.2022	Friday	Explain the Structure of the Project

5th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	11.07.2022	Monday	Data Preprocessing
	12.07.2022	Tuesday	Data Preprocessing
	13.07.2022	Wednesday	Data Preprocessing
	14.07.2022	Thursday	Data Preprocessing
	15.07.2022	Friday	Data Preprocessing

6th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	18.07.2022	Monday	Perform Analysis
	19.07.2022	Tuesday	Perform Analysis
	20.07.2022	Wednesday	Perform Analysis
	21.07.2022	Thursday	Perform Analysis
	22.07.2022	Friday	Perform Analysis

7th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	25.07.2022	Monday	PPT Preparation
	26.07.2022	Tuesday	PPT Preparation
	27.07.2022	Wednesday	PPT Preparation
	28.07.2022	Thursday	PPT Preparation
	29.07.2022	Friday	PPT Preparation

8th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	01.08.2022	Monday	PPT Submission
	02.08.2022	Tuesday	PPT Submission
	03.08.2022	Wednesday	Mid Review
	04.08.2022	Thursday	Mid Review
	05.08.2022	Friday	Mid Review

9th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	08.08.2022	Monday	Mid Review
	10.08.2022	Tuesday	Mid Review
	11.08.2022	Wednesday	Building & Applying Algorithm
	12.08.2022	Thursday	Building & Applying Algorithm

10th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	16.08.2022	Tuesday	Building & Applying Algorithm
	17.08.2022	Wednesday	Building & Applying Algorithm
	19.08.2022	Friday	Building & Applying Algorithm

11th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	22.08.2022	Monday	Concluding Project
	23.08.2022	Tuesday	Concluding Project
	24.08.2022	Wednesday	Concluding Project
	25.08.2022	Thursday	Concluding Project
	26.08.2022	Friday	Concluding Project

12th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	29.08.2022	Monday	Final Review
	30.08.2022	Tuesday	Final Review
	01.09.2022	Wednesday	Final Review
	02.09.2022	Thursday	Final Review
	05.09.2022	Friday	Final Review

INDEX

1. INTRODUCTION

One of the trending topics in financial markets is stocks. The stock market is a trading platform where different investors sale and purchase their shares according to stock availability. A stock or a share which is also known as a company's equity is referred to as a financial instrument that is used to represent an ownership in a company that represents a proportional assertion on its assets and earnings. Stock price prediction is an example of time series forecasting that promptly examine previous data and estimate future data values. A correct prediction of stocks can lead to huge profits for the seller and the broker. Recently, many traders have been employing the machine learning models for the stock price prediction because of its rising popularity as a result of its proficiency and effectiveness in the prediction. Frequently, it is brought that the prediction is chaotic rather than random, which means it can be predicted by carefully analyzing the history of respective stock market.The typical scientific stock price forecasting procedures are focused on the statistical analysis of stock data. We have selected the Machine learning techniques here because they provide us with better results as compared to any other prediction model. Here will develop a stock data predictor program that uses previous stock prices and data will be treated as training sets for the program to predict the stock prices of a particular share this program develops a procedure. This paper majorly deals with predicting the stock prices using LSTM.

In present days so many people's are interested in investing money in stock market for earning more in short period.

Here ,in stock market consist of many number of company shares along with prices in stock market every minute stock price will changes depending on the company environment and country economic structure decisions. In stock market there are many brokers for handing the stocks buying and selling between the clients and company. In previous year's there is difficult to predict the stock market because lack of technology and knowledge but in present days technology will increases day by day for that we can predict the stock market easily when compare to past here we can predict stock price by analyzing the previous data by using machine learning techniques. From these techniques we can use neural network(it's means that it is interconnected with networking it look like human neural brain structure) and
Simple moving average method.

2. LITERATURE SURVEY

Money-related trade judgment-making is a strengthening and difficult errand of fiscal data guessing. Figure about securities trade with high exactness improvement return advantage for examiners of the stocks. In perspective on the snare of budgetary trade financial data, expansion of productive models for forecast conclusion is very difficult, and it must be precise. This consider attempted to make models for guess of the securities trade and to pick whether to buy/hold the stock using data mining and ML techniques like Decision Tree (DT), Naive Bayes, k-Nearest neighbors (k-NN), Support Vector Machine(SVM), Artificial Neural Network(ANN) and Random Forest has been used for progressing of gauge model. Particular pointers are resolved from the stock prices set up on timetable data and it is used as commitments of the proposed guess models. Ten years of securities trade data has been used for sign gauge of stock. Based on the instructive accumulation, these models can make buy/hold signal for monetary trade as a yield. The rule target of this errand is to deliver yield signal(buy/hold) as per customers essential like mean be contributed, time term of endeavor, least advantage, most prominent hardship, using data mining and ML techniques. Forecasting the way of stock prices is a widely deliberate subject in many fields including trading, finance, statistics and computer science. Depositors in the stock market can maximize their yield by export or selling their investment if they can determine when to enter and exit a position. Specialized traders typically use essential and/or technical analysis to inspect stocks in making venture decisions. The objective function is to maximize medium to longer term profits based on S&P500 stock market index. The inputs are the technical pointers data and the economic indicators data. Three models (neural network, soft max logistic regression, decision forest) are then used to predict the buy/sell decisions.

3. PROBLEM STATEMENT

The aim of the project is to examine a number of different forecasting techniques to predict the future stock returns to construct a portfolio of multiple stocks in order to diversify the risk. We do this by applying supervised learning methods for stock price forecasting by interpreting the data.

The main objective of this project is to find the best model to predict the value of the stock market.

To examine a number of different forecasting techniques to predict future stock market.

To determine the values that particular stock will have in near future.

The Stock price prediction task is interesting as well as it divides researchers and academics into two groups those who believe that we can devise mechanisms to predict the market and those who believe that the market is efficient and whenever new information comes up the market absorbs it by correcting itself.

4. METHODOLOGY

Some of the methodology's are
ANN
RNN
CNN

Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) widely used for the prediction of stock price movements. Every algorithm has its way of learning patterns and then predicting. Artificial Neural Network (ANN) is a popular and most recent method which incorporates technical analysis for making predictions in financial markets. ANN includes a set of threshold functions. These functions trained on historical data after connecting each other with adaptive weights and they are used to make future predictions. The ANN models can be even more important for real scenarios, especially in situations where some of the emerging contaminants breakdown into metabolites during the wastewater treatment processes. In these cases that are characterized by a nonlinear behavior, the ANN model can be used to optimize the operating conditions in order to maximize the adsorption of these contaminants and their metabolites.

Recurrent Neural Network (RNN)

Recurrent Neural Network can memorize previous input from memory when huge set of sequential data is given. RNN are more complex. They save the output of processing nodes and feed the result back into the model (they did not pass the information in one direction only). This is how the model is said to learn to predict the outcome of a layer. Each node in the RNN model acts as a memory cell, continuing the computation and implementation of operations. If the network's prediction is incorrect, then the system self-learns and continues working towards the correct prediction during back propagation. all the inputs and outputs are independent of each other, but in cases like when it is required to predict the next word of a sentence, the previous words are required and hence there is a need to remember the previous words. Thus RNN came into existence, which solved this issue with the help of a Hidden Layer. The main and most important feature of RNN is Hidden state, which remembers some information about a sequence.

Convolution Neural Network(CNN)

Convolution Neural Network is a feed-forward neural network. Like the traditional architecture of a neural network including input layers, hidden layers and output layers, convolutional neural network also contains these features and the input of the layer of convolution are the output of the previous layer of convolution or pooling. Of course, they still have some unique features such as pooling layers, full connection layers, etc. The number of hidden layers in a convolution neutral network is more than that in a traditional neural network, which, to some extent, shows that the capability of the neural network. The more the hidden layers are, the higher feature it can extract and recognize from the input. People always use convolutional neural network in computer vision, such as face recognition, image classification.

5. ANALYSIS

Fundamental Analysis

Fundamental analysis involves the examination of economic factors that influence the price of a stock. Such factors include a balance sheet and income statement. The balance sheet is a financial statement that provides information about a company's assets, liabilities as well as the equity of their shareholders at a specific point in time. Basically, the balance sheet gives intel into what a company owns and owes and the amount investors have invested in it. The income statement is another type of financial statement that gives a synopsis of a company's performance by providing information about their revenues, expenses, and net profit/loss over time. These reports are released quarterly throughout the year. Because fundamental analysis relies on the reports that are issued on the basis of a slower time frame, this type of analysis is often used to project long-term price movements.

Technical Analysis

The goal of technical analysis is to anticipate what other stock holders are thinking based on available information about the price and volumes of stock. Technical analysts use a number of different types of indicators calculated from the past history of stock price and volume to predict future prices. Overall, the key to technical analysis is trend. Practitioners of technical analysis argue that trends in stock prices are caused by an imbalance between the supply and demand of stocks, which is reflected in the bid and ask prices. From the noisy data of stock prices, technical analysts attempt to extract patterns. Two examples of such stock charts are shown below. Fig. 1 represents historical price data for stocks for the IT sector for one year. Fig. 1,2 and 3 represent candlestick charts. Each bar or "candle" represents one day's high, low, and closing prices. The additional lines on the charts are examples of different technical indicators.

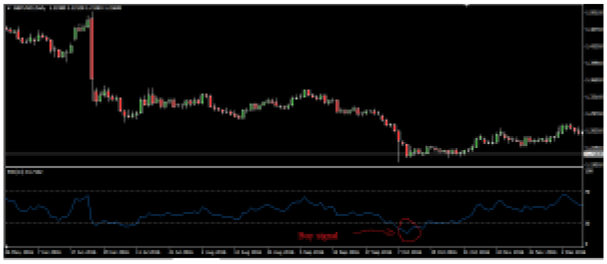


Fig. 1: Relative Strength Index

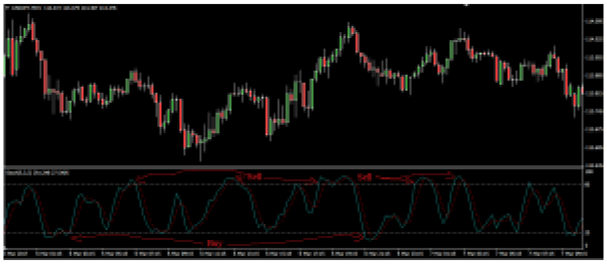


Fig. 2: Oscillator

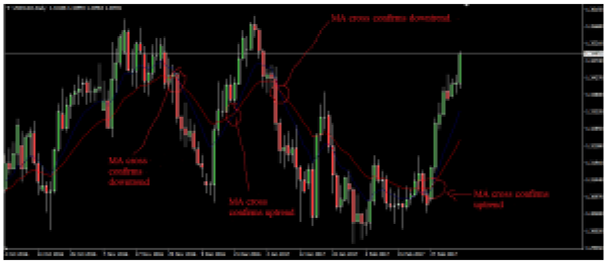


Fig. 3: Moving Average

Quantitative Technical Analysis

This qualitative aspect to our second school of thought is what differentiates it from our next methodology. As suggested by its name, this form of stock prediction relies on quantitative methods of prediction rather than visualizations on graphs. Specifically, the usages of machine learning algorithms are explored to predict future stock prices.

6.COMPARISIONS

RNN vs ANN:

- RNN is the most advanced and complex neural network where as ANN is One of the simplest types of neural networks
- In RNN Information flows in different directions, which gives it its memory and self-learning features where as in ANN information flows in one direction only
- RNN's commendable feature is to memorize and self learning where as for ANN , the commendable feature is Ability to work with incomplete knowledge and high fault tolerance.

RNN VS LSTM:

- Recurrent Neural Networks RNNs
- RNNs have feedback loops in the recurrent layer. This lets them maintain information in 'memory' over time. But, it can be difficult to train standard RNNs to solve problems that require learning long-term temporal dependencies.
- This is because the gradient of the loss function decays exponentially with time (called the vanishing gradient problem).

- Long Short-Term Memory LSTM
- LSTM networks are a type of RNN that uses special units in addition to standard units. LSTM units include a 'memory cell' that can maintain information in memory for long periods of time. This memory cell lets them learn longer-term dependencies.
- LSTMs deal with vanishing and exploding gradient problem by introducing new gates, such as input and forget gates, which allow for a better control over the gradient flow and enable better preservation of "long-range dependencies"..

RNN vs CNN:

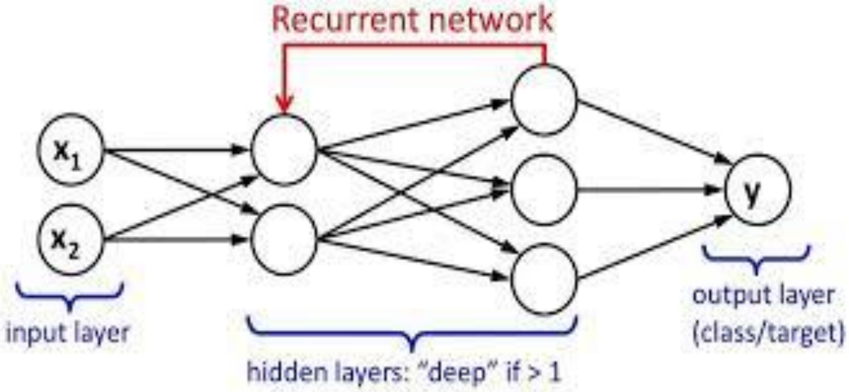
- RNN uses Natural language processing including sentiment analysis and speed recognition where as CNN uses Computer vision including image recognition.
- Data type of RNN is trained with sequence data where as in CNN data type relies on image data.
- RNN's commendable feature is to memorize and self learning where as for CNN the commendable feature is the accuracy in recognizing images.

7. EXISTING SYSTEM

Recurrent Neural Network

A recurrent neural network (RNN) is a class of advanced artificial neural network (ANN) that involves directed cycles in memory. One goal of recurrent neural networks is the ability to build on earlier types of networks with fixed-size input vectors and output vectors. In a recurrent neural network (RNN); connections between nodes form a directed graph along a sequence which allows exhibiting dynamic temporal behavior for a time sequence. Suppose one wants to predict the next word in a sentence or to predict the next day stock price etc. by using Machine Learning. The simplest form has an input layer which receives the input, a hidden layer where the activation is applied and an output layer where one finally receives the output. In more complex forms, where multiple hidden layers are present, the input layer receives the input, the first hidden layer applies its activations, these activations are sent to the next hidden layer, and each successive layers activation are sent through the layers to finally produce the output. Each hidden layer has its own weights and bias. For this, each layer behaves independently and, unless they have the same weights and bias, these hidden layers cannot be combined with one another. So, a recurrent neuron stores the state of a previous input and combines it with the current input, thereby preserving some relationship of the current input with the previous input.

The purpose of RNN is to process sequence data. In the traditional neural network model, from the input layer to the hidden layer to the output layer, the layers are fully connected, and the nodes between each layer are not connected. But this kind of ordinary neural network is powerless for many problems. The reason why RNN is called recurrent neural network means that the current output of a sequence is also related to the previous output. The specific form of expression is that the network will memorize the previous information and apply it to the calculation of the current output, that is, the nodes between the hidden layers are no longer unconnected but connected, and the input of the hidden layer not only includes the output of the input layer It also includes the output of the hidden layer at the previous moment. In theory, RNN can process sequence data of any length.

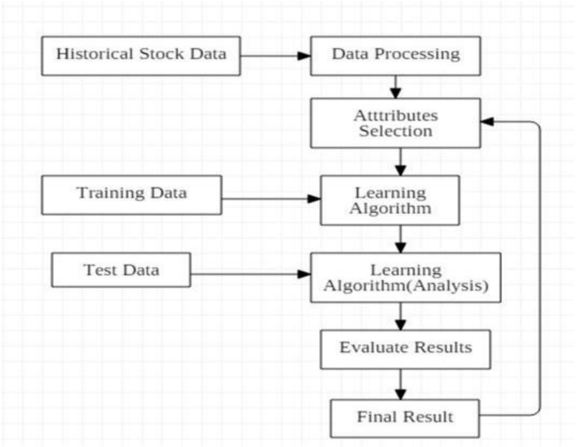


8. PROPOSED SYSTEM

Proposed work of a Stock Price Predictor Using LSTM

Working process is as follows

Firstly, we have to collect data that is historical data of stocks from the source.



After collecting data we will normalize the data. Normalization is one of the data preprocessing techniques which is used to organize data into database. Data normalization is to guarantee the quality of data before it is fed to any learning algorithm.

After normalizing data, scaling of data is done that is we scale data in some range of values for each input in order to minimize the data.
Now we have to divide the entire dataset into training set and test set.

Training set is a subset of original data which is used to training the data on LSTM Model. Testing dataset is independent of training data set which is used for testing accuracy and back propagation on testing dataset.

Train all modules using training data set

Predict the stocks using training model

There are many algorithms compared best model between those algorithms.

On choosing the best model Save the training model

Finally display model to website and show the predictions of Stock Prices.

Understanding LSTM Model

Stocks are volatile. Stock prices change because of supply and demand. To predict accurate stock price is extremely challenging because of multiple factors such as global economic conditions, companies financial performance, politics and so on.

So we use Long Short Term Memory Model which gives accurate results. LSTM is a special kind of recurrent neural network and it is designed to overcome long term dependencies.

It is commonly used for processing and predicting time series data. LSTM works on data sequence and composed of cells. It has three gates namely,

Input Gate:

It decides how much information from current input flows to cell state.

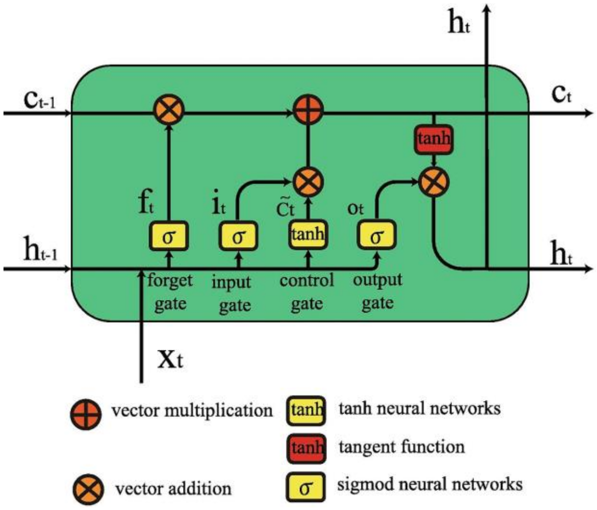
Forget Gate:

It controls if the memory cell is reset to zero i.e., it determines to what extent to forget the previous data.

Output Gate:

It decides how much information from current cell date flows into hidden state so that LSTM can only pick Long Term Memory or Short Term Memory.

Each of these gates take hidden state, Hidden state can be processed to obtain more meaningful data.



LSTMs work in a three-step process

The first step in LSTM is to decide which information to be omitted from the cell in that particular time step. It is decided with the help of a sigmoid function. It looks at the previous state (ht-1) and the current input xt and computes the function.

There are two functions in the second layer. The first is the sigmoid function, and the second is the tanh function. The sigmoid function decides which values to let through (0 or 1). The tanh function gives the weightage to the values passed, deciding their level of importance from -1 to 1.

The third step is to decide what will be the final output. First, you need to run a sigmoid layer which determines what parts of the cell state make it to the output. Then, you must put the cell state through the tanh function to push the values between -1 and 1 and multiply it by the output of the sigmoid gate.

With this basic understanding of LSTM, you can dive into the hands-on demonstration part of this tutorial regarding stock price prediction using machine learning.

9.SOFTWARE AND HARDWARE REQUIREMENTS

Software tools required for stock price prediction:

The tools required are

1. Jupyter notebook
2. NumPy
3. Pandas
4. Seaborn
5. Matplotlib

Jupyter notebook:

The Jupyter Notebook is a web-based interactive computing platform. It allows user to compile all aspects of data project in one place. Notebook documents are both human-readable documents containing the analysis description and the results (figures, tables, etc…) as well as executable documents which can be run to perform data analysis.

NumPy:

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely. NumPy stands for Numerical Python. It is open-source software and is flexible. It uses less memory for storing data.

Pandas:

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

Seaborn:

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas. Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs, so that we can switch between different visual representations for same variables for better understanding of dataset.

Matplotlib:

Matplotlib is a python library used to create 2D graphs and plots by using python scripts. It has a module named pyplot which makes things easy for plotting by providing feature to control line styles, font properties, formatting axes etc. It supports a very wide variety of graphs and plots namely - histogram, bar charts, power spectra, error charts etc. It is used along with NumPy to provide an environment that is an effective open-source alternative for MATLAB.

Hardware requirements:

Hardware requirements are

- Machine learning is a very CPU intensive program-sequel thing to be running, so be prepared to shell out a lot of money for a good enough system. Here are some system requirements to adhere to (parenthesis are what you can maybe get away with)

- Quad core Intel Core i7 Skylake or higher (Dual core is not the best for this kind of work, but manageable)

- 16GB of RAM (8GB is okay but not for the performance you may want and or expect)

- M.2 PCIe or regular PCIe SSD with at least 256GB of storage, though 512GB is best for performance. The faster you can load and save your applications, the better the system will perform. (SATA III will get in the way of the system's performance)

- Premium graphics cards, so things with GTX 980 or 980Ms would be the best for a laptop, and 1080s or 1070s would be the best for the desktop setup**.**

10. ADVANTAGES AND DISADVANTAGES

Advantages:

- Stock markets help companies to raise capital.
- It helps generate personal wealth.
- Stock markets serve as an indicator of the state of the economy.
- It is a widely used source for people to invest money in companies with high growth potential.

Disadvantages:

- However accuracy would decrease when setting more levels of stock market movement.
- The average of prediction accuracies using Decision Tree as the classifier are 43.44%, 31.92%, and 12.06% for “two levels,” “three levels,” and “five levels,” respectively.
- These results indicate that the stock price is unpredictable when traditional classifier is used.

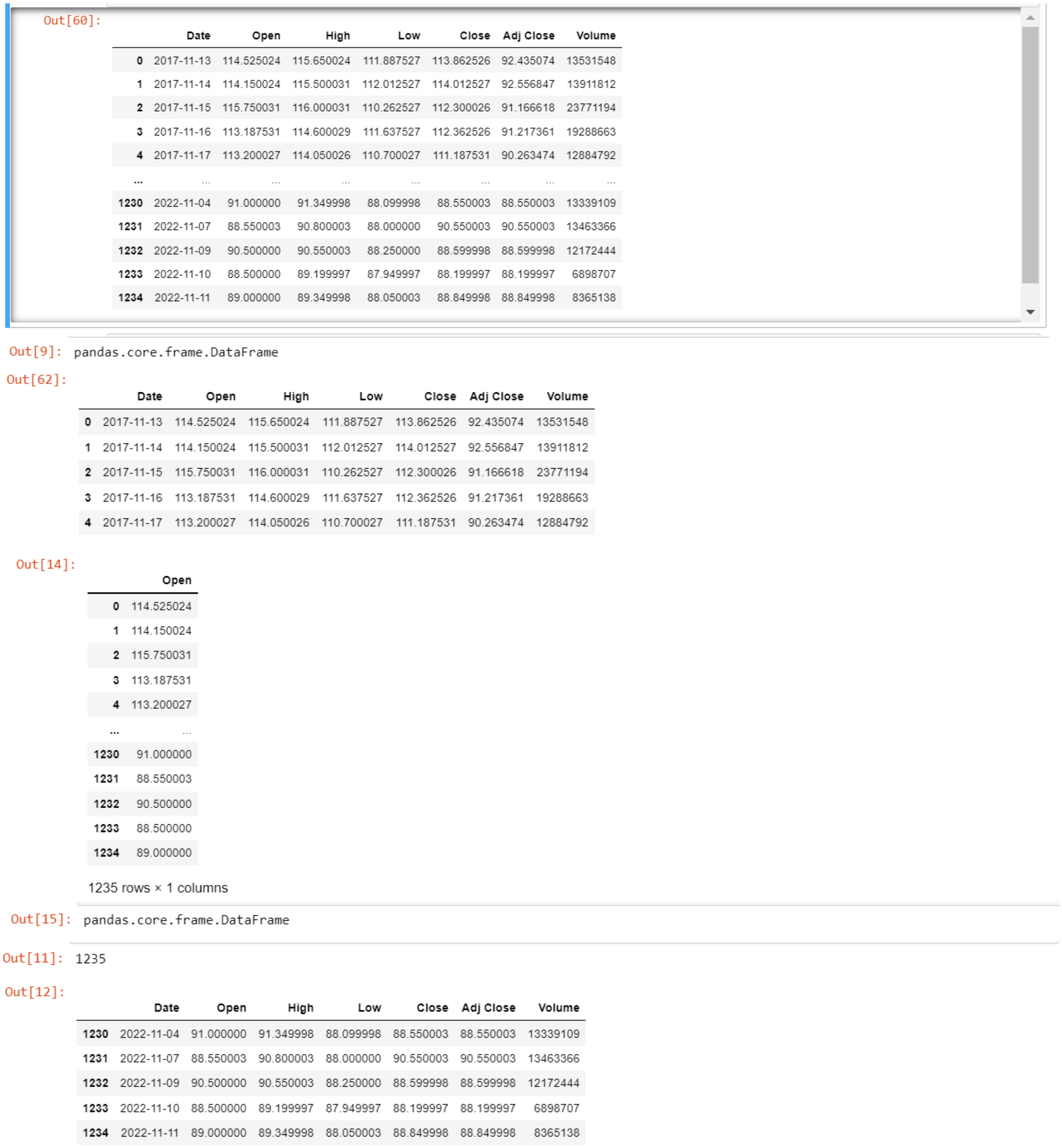
11. CODE

```
import pandas as pd
import numpy as np
data = pd.read_csv ('GAIL.NS.csv')
stock symbol = 'GAIL.NS'
type (data)
data.head()
len (data)
data.tail()
opn = data [['Open']]
type(opn)
opn.plot()
import matplotlib.pyplot as plt
ds = opn.values
type (ds)
plt.plot (ds)
from sklearn.preprocessing import MinMaxScaler
#Using MinMaxScaler for normalizing data between 0 & 1, x1=(1 to 20) and x2=(50-1000) and
y
#used to predict then minmax scaler is biased to a greater value
normalizer = MinMaxScaler(feature_range=(0,1))
ds_scaled = normalizer.fit_transform (np.array (ds).reshape (-1,1))
len (ds_scaled), len(ds)
#Defining test and train data sizes 70% for training and rest for testing
train_size = int(len(ds_scaled)*0.70)
test_size = len(ds_scaled) - train_size
train_size,test_size
ds_train, ds_test = ds_scaled[0:train_size,:], ds_scaled[train_size:len(ds_scaled),:1]
len(ds_train),len(ds_test)
#creating dataset in time series for LSTM model  single record for x and y
#this fnc takes len(x) records and predicts the next value
#X[100,120,140,160,180] : Y[200]
def create_ds(dataset,step):
    Xtrain, Ytrain = [], []
    for i in range(len(dataset)-step-1):
        a = dataset[i:(i+step), 0]
        Xtrain.append(a)
        Ytrain.append(dataset[i + step, 0])
    return np.array(Xtrain), np.array(Ytrain)
#Taking 100 days price as one record for training
time_stamp = 100
X_train, y_train = create_ds(ds_train,time_stamp)
X_test, y_test = create_ds(ds_test,time_stamp)
X_train.shape,y_train.shape
```

```
X_test.shape, y_test.shape
#Reshaping data to fit into LSTM model
X_train = X_train.reshape(X_train.shape[0],X_train.shape[1] , 1)
X_test = X_test.reshape(X_test.shape[0],X_test.shape[1] , 1)
from keras.models import Sequential
from keras.layers import Dense, LSTM
#creating LSTM model using keras
model = Sequential()
model.add(LSTM(units=50,return_sequences=True,input_shape=(X_train.shape[1],1)))
model.add(LSTM(units=50,return_sequences=True))
model.add(LSTM(units=50))
model.add(Dense(units=1,activation='linear'))
model.summary()
#Training model with adam optimizer and mean squared error loss function
model.compile(loss='mean_squared_error',optimizer='adam')
model.fit(X_train,y_train,validation_data=(X_test,y_test),epochs=100,batch_size=64)
#PLotting loss, it shows that loss has decreased significantly and model trained well
loss = model.history.history['loss']
plt.plot(loss)
#Predicitng on train and test data
train_predict = model.predict(X_train)
test_predict = model.predict(X_test)
#Inverse transform to get actual value
train_predict = normalizer.inverse_transform(train_predict)
test_predict = normalizer.inverse_transform(test_predict)
#Comparing using visuals
plt.plot(normalizer.inverse_transform(ds_scaled))
plt.plot(train_predict)
plt.plot(test_predict)
type(train_predict)
test = np.vstack((train_predict,test_predict))
#Combining the predited data to create uniform data visualization
plt.plot(normalizer.inverse_transform(ds_scaled))
plt.plot(test)
len(ds_test)
#Getting the last 100 days records
fut_inp = ds_test[270:]
fut_inp = fut_inp.reshape(1,-1)
tmp_inp = list(fut_inp)
fut_inp.shape
#Creating list of the last 100 data
tmp_inp = tmp_inp[0].tolist()
#Predicting next 30 days price suing the current data
#It will predict in sliding window manner (algorithm) with stride 1
lst_output=[]
n_steps=100
```

```
i=0
while(i<30):

    if(len(tmp_inp)>100):
        fut_inp = np.array(tmp_inp[1:])
        fut_inp=fut_inp.reshape(1,-1)
        fut_inp = fut_inp.reshape((1, n_steps, 1))
        yhat = model.predict(fut_inp, verbose=0)
        tmp_inp.extend(yhat[0].tolist())
        tmp_inp = tmp_inp[1:]
        lst_output.extend(yhat.tolist())
        i=i+1
    else:
        fut_inp = fut_inp.reshape((1, n_steps,1))
        yhat = model.predict(fut_inp, verbose=0)
        tmp_inp.extend(yhat[0].tolist())
        lst_output.extend(yhat.tolist())
        i=i+1
print(lst_output)
len(ds_scaled)
#Creating a dummy plane to plot graph one after another
plot_new=np.arange(1,101)
plot_pred=np.arange(101,131)
plt.plot(plot_pred, normalizer.inverse_transform(lst_output))
ds_new = ds_scaled.tolist()
len(ds_new)
#Entends helps us to fill the missing value with approx value
ds_new.extend(lst_output)
plt.plot(ds_new[1200:])
#Creating final data for plotting
final_graph = normalizer.inverse_transform(ds_new).tolist()
#Plotting final results with predicted value after 30 Days
plt.plot(final_graph,)
plt.ylabel("Price")
plt.xlabel("Time")
plt.title("{0} prediction of next month open".format(stock_symbol))
plt.axhline(y=final_graph[len(final_graph)-1], color = 'red', linestyle = ':', label = 'NEXT 30D:
{0}'.format(round(float(*final_graph[len(final_graph)-1]),2)))
plt.legend()
```

12. RESULT



Out[60]:

	Date	Open	High	Low	Close	Adj Close	Volume
0	2017-11-13	114.525024	115.650024	111.887527	113.862526	92.435074	13531548
1	2017-11-14	114.150024	115.500031	112.012527	114.012527	92.556847	13911812
2	2017-11-15	115.750031	116.000031	110.262527	112.300026	91.166618	23771194
3	2017-11-16	113.187531	114.600029	111.637527	112.362526	91.217361	19288663
4	2017-11-17	113.200027	114.050026	110.700027	111.187531	90.263474	12884792
...	...	...	...	...	...	...	...
1230	2022-11-04	91.000000	91.349998	88.099998	88.550003	88.550003	13339109
1231	2022-11-07	88.550003	90.800003	88.000000	90.550003	90.550003	13463366
1232	2022-11-09	90.500000	90.550003	88.250000	88.599998	88.599998	12172444
1233	2022-11-10	88.500000	89.199997	87.949997	88.199997	88.199997	6898707
1234	2022-11-11	89.000000	89.349998	88.050003	88.849998	88.849998	8365138

```
Out[9]: pandas.core.frame.DataFrame
```

Out[62]:

	Date	Open	High	Low	Close	Adj Close	Volume
0	2017-11-13	114.525024	115.650024	111.887527	113.862526	92.435074	13531548
1	2017-11-14	114.150024	115.500031	112.012527	114.012527	92.556847	13911812
2	2017-11-15	115.750031	116.000031	110.262527	112.300026	91.166618	23771194
3	2017-11-16	113.187531	114.600029	111.637527	112.362526	91.217361	19288663
4	2017-11-17	113.200027	114.050026	110.700027	111.187531	90.263474	12884792

Out[14]:

	Open
0	114.525024
1	114.150024
2	115.750031
3	113.187531
4	113.200027
...	...
1230	91.000000
1231	88.550003
1232	90.500000
1233	88.500000
1234	89.000000

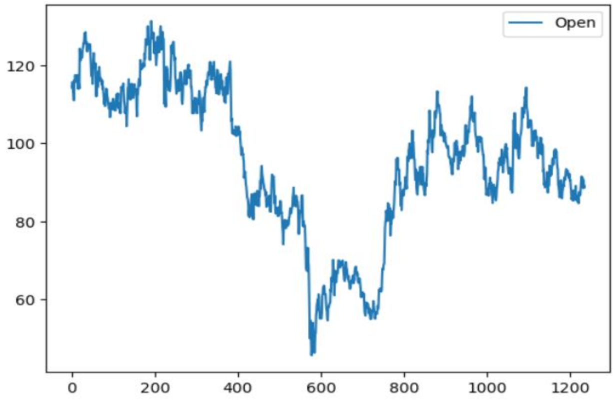
1235 rows × 1 columns

```
Out[15]: pandas.core.frame.DataFrame

Out[11]: 1235
```

Out[12]:

	Date	Open	High	Low	Close	Adj Close	Volume
1230	2022-11-04	91.000000	91.349998	88.099998	88.550003	88.550003	13339109
1231	2022-11-07	88.550003	90.800003	88.000000	90.550003	90.550003	13463366
1232	2022-11-09	90.500000	90.550003	88.250000	88.599998	88.599998	12172444
1233	2022-11-10	88.500000	89.199997	87.949997	88.199997	88.199997	6898707
1234	2022-11-11	89.000000	89.349998	88.050003	88.849998	88.849998	8365138

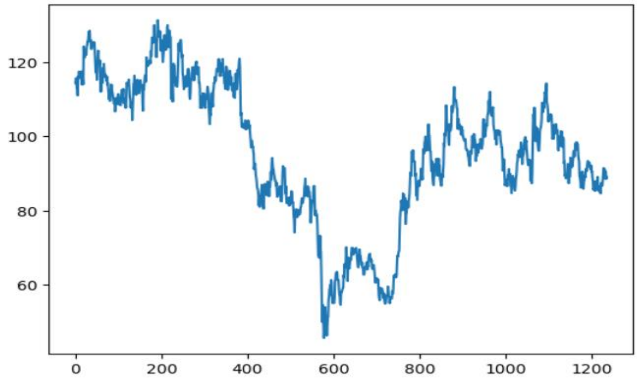
```
Out[16]: <AxesSubplot:>
```



```
Out[19]: array([[114.525024],
               [114.150024],
               [115.750031],
               ...,
               [ 90.5     ],
               [ 88.5     ],
               [ 89.      ]])
```

```
Out[63]: numpy.ndarray
```

```
Out[20]: [<matplotlib.lines.Line2D at 0x255ba2ddb80>]
```

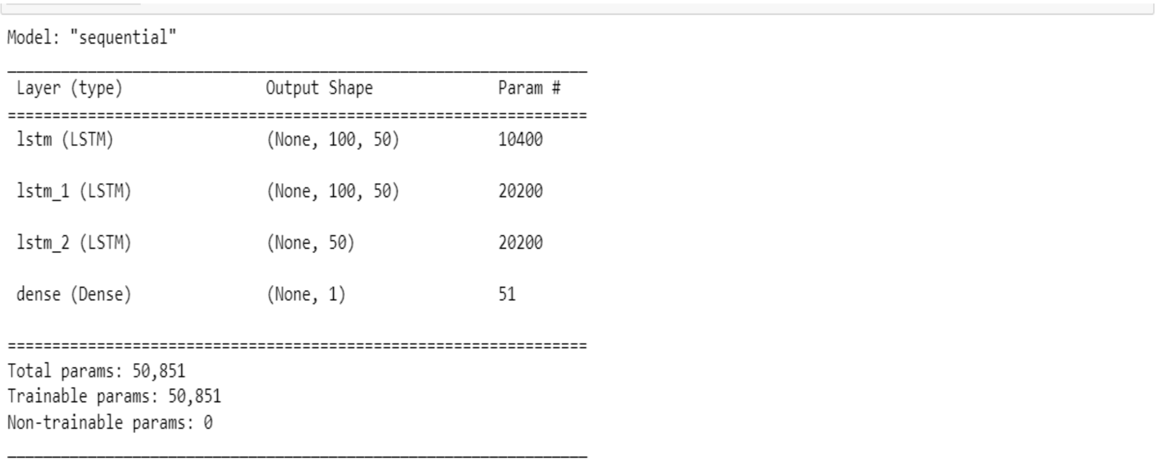


```
Out[23]: (1235, 1235)
```

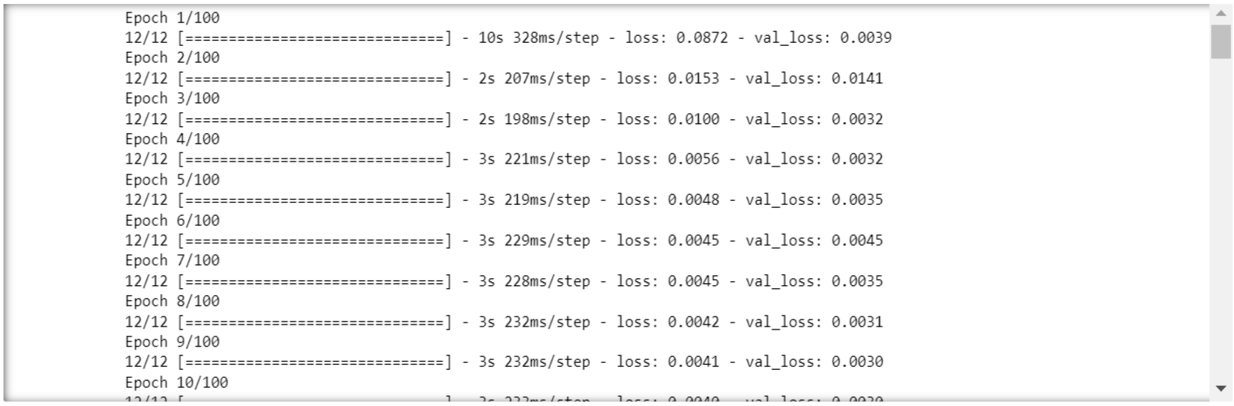
```
Out[25]: (864, 371)
```

```
Out[31]: ((270, 100), (270,))
```

```
Out[30]: ((763, 100), (763,))
```

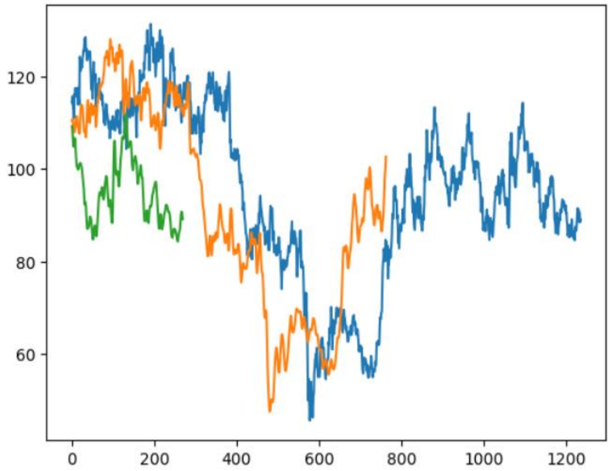


```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 100, 50)           10400

 lstm_1 (LSTM)               (None, 100, 50)           20200

 lstm_2 (LSTM)               (None, 50)                20200

 dense (Dense)               (None, 1)                 51

=================================================================
Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0
_________________________________________________________________
```

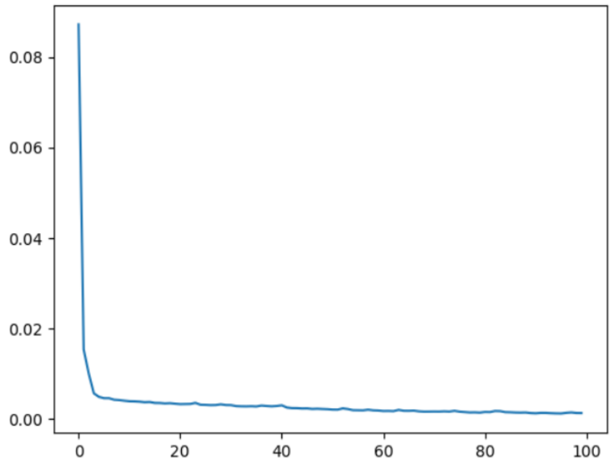


```
Epoch 1/100
12/12 [==============================] - 10s 328ms/step - loss: 0.0872 - val_loss: 0.0039
Epoch 2/100
12/12 [==============================] - 2s 207ms/step - loss: 0.0153 - val_loss: 0.0141
Epoch 3/100
12/12 [==============================] - 2s 198ms/step - loss: 0.0100 - val_loss: 0.0032
Epoch 4/100
12/12 [==============================] - 3s 221ms/step - loss: 0.0056 - val_loss: 0.0032
Epoch 5/100
12/12 [==============================] - 3s 219ms/step - loss: 0.0048 - val_loss: 0.0035
Epoch 6/100
12/12 [==============================] - 3s 229ms/step - loss: 0.0045 - val_loss: 0.0045
Epoch 7/100
12/12 [==============================] - 3s 228ms/step - loss: 0.0045 - val_loss: 0.0035
Epoch 8/100
12/12 [==============================] - 3s 232ms/step - loss: 0.0042 - val_loss: 0.0031
Epoch 9/100
12/12 [==============================] - 3s 232ms/step - loss: 0.0041 - val_loss: 0.0030
Epoch 10/100
12/12 [==============================] - 3s 232ms/step - loss: 0.0040 - val_loss: 0.0030
```

```
Out[39]: [<matplotlib.lines.Line2D at 0x255d3d18520>]
```

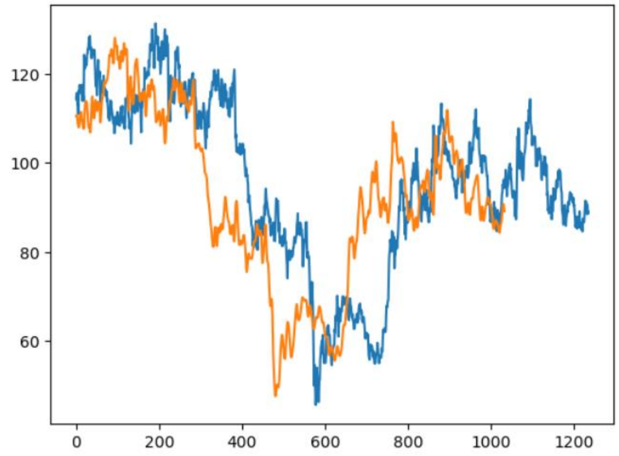


```
Out[36]: [<matplotlib.lines.Line2D at 0x255d29c67f0>]
```



```
Out[40]: numpy.ndarray
```

```
Out[42]: [<matplotlib.lines.Line2D at 0x255d4e77dc0>]
```

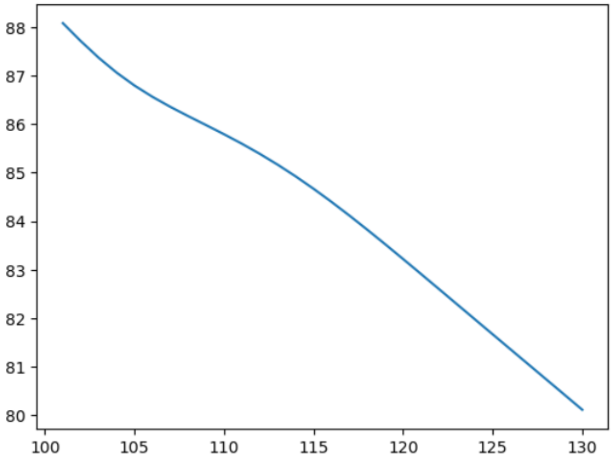


```
Out[43]: 371
```

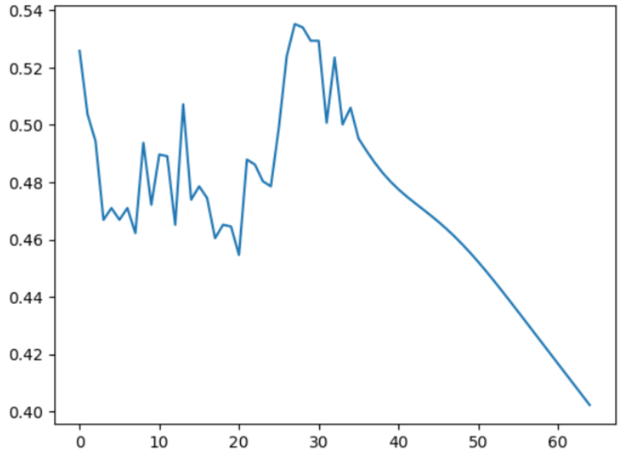
```
Out[47]: (1, 101)
```

```
[[0.49529969692230225], [0.4909862279891968], [0.4869818091392517], [0.48340973258018494], [0.4802994430065155], [0.47759032249
450684], [0.4751768708229065], [0.47294217348098755], [0.4707763195037842], [0.4685860574245453], [0.4662985801696773], [0.463
86343240737915], [0.46125122904777527], [0.4584517776966095], [0.45546960830688477], [0.452321320772171], [0.4490303695201874],
[0.44562405347824097], [0.44212988018989563], [0.43857380747795105], [0.4349776804447174], [0.4313592314720154], [0.42773106694
221497], [0.4241012930870056], [0.420474112033844], [0.41685032844543457], [0.41322842240333557], [0.4096051752567291], [0.4059
7692131996155], [0.4023396074771881]]
```

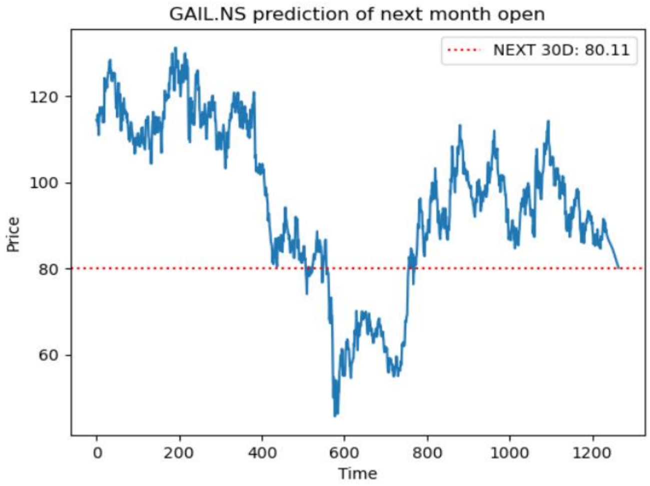
```
Out[54]: [<matplotlib.lines.Line2D at 0x255d9069580>]
```



```
Out[57]: [<matplotlib.lines.Line2D at 0x255d8fdf100>]
```



```
Out[59]: <matplotlib.legend.Legend at 0x255d900ceb0>
```



13. CONCLUSION

Stock price prediction is the area of high interest for equity traders, individual investors, and portfolio managers. However, precise and consistent stock price prediction is a difficult task due to its noisy and non-linear manner.
There are several factors that can impact the prediction such as fundamental market data, macroeconomic data, technical indicators, and others.

Predicting stock market returns is a challenging task due to consistently changing stock values which are dependent on multiple parameters which form complex patterns. The historical dataset available on company's website consists of only few features like high, low, open, close, adjacent close value of stock prices, volume of shares traded etc., which are not sufficient enough.

To obtain higher accuracy in the predicted price value new variables have been created using the existing variables

This study focuses on developing LSTM based models to predict S&P 500 index's closing price by extracting a well-balanced combination of input variables capturing the multiple aspects of the economy and broader markets.

Both single and multilayer LSTM architectures have been implemented and their performances are analyzed by using various evaluation metrics to identify the best model.

The experimental results show that single layer LSTM model can provide a superior fit and high prediction accuracy compared to multilayer LSTM.

The proposed model can be easily customized to apply in other broad market indexes where the data exhibits a similar behavior. Interested stakeholders can use the proposed model to better inform the market situation before making their investment decisions.

We explored LSTM and stock price using LSTM. We then visualized the opening and closing price value after using LSTM.

14. BIBLIOGRAPHY

[1] Usmani, S. H. Adil, K. Raza and S. S. A. Ali, "Stock market prediction using machine learning techniques," 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), Kuala Lumpur, 2016, pp. 322-327.

[2] K. Raza, "Prediction of Stock Market performance by using machine learning techniques," 2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT), Karachi, 2017, pp. 1-1.

[3] H. Gunduz, Z. Cataltepe and Y. Yaslan, "Stock market direction prediction using deep neural networks," 2017 25th Signal Processing and Communications Applications Conference (SIU), Antalya, 2017, pp. 1-4.

[4] C K Gomathy and V Geetha. Article: A Real Time Analysis of Service based using Mobile Phone Controlled Vehicle using DTMF for Accident Prevention. International Journal of Computer Applications 138(2):11-13, March 2016. Published by Foundation of Computer Science (FCS), NY, USA,ISSN No: 0975-8887

[5] C K Gomathy and V Geetha. Article: Evaluation on Ethernet based Passive Optical Network Service Enhancement through Splitting of Architecture. International Journal of Computer Applications 138(2):14-17, March 2016. Published by Foundation of Computer Science (FCS), NY, USA, ISSN No: 0975-8887

[6] Dr.C.K.Gomathy , K. Bindhu sravya , P. Swetha , S.Chandrika Article: A Location Based Value Prediction for Quality of Web Service, Published by International Journal of Advanced Engineering Research and Science (IJAERS), Vol-3, Issue-4 , April- 2016] ISSN: 2349-6495

[7]DR.C.K.Gomathy , V.Geetha , S.Madhumitha , S.Sangeetha , R.Vishnupriya Article: A Secure With Efficient Data Transaction In Cloud Service, Published by International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 5 Issue 4, March 2016, ISSN: 2278 – 1323.

[8]C.K.Gomathy.(2010),"Cloud Computing: Business Management for Effective Service Oriented Architecture" International Journal of Power Control Signal and Computation (IJPCSC), Volume 1, Issue IV, Oct - Dec 2010, P.No:22-27, ISSN: 0976-268X