STOCK MARKET PREDICTION

Submitted in partial fulfillment of the requirements

of the degree of

T. E. Computer Engineering

By

Menezes Nithin Mark 56

182066

Mishra Subhashkumar 61

182071

Aditya Suthar 62

Guide:

Monali korde

Assistant Professor

Department of Computer Engineering

St. Francis Institute of Technology

(Engineering College)

University of Mumbai 2021-2022

CERTIFICATE

This is to certify that the project entitled **"Stock Market Prediction"**is a bonafide work of **"Nithin Menezes" (58) and "Subhashkumar Mishra"(61) , Aditrya suthar** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of T.E. in Computer Engineering

Monali korde
Guide

Dr. Kavita Sonawane
Head of Department

Project Report Approval for B.E.

This project report entitled ***Stock Market Prediction*** by ***Nithin Menezes, Subhashkumar , Aditya suthar*** is approved for the degree of ***T.E. in Computer Engineering.***

Examiners

Monali korde

Date: 08/04/2022

Place:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. we understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Menezes Nithin Mark

—----------------

Mishra Subhashkumar

Aditya Suthar

Date: 08/04/2022

Abstract

In Stock Market Prediction, the aim is to predict the future values of the financial stocks of a company. As per efficient market theory when all information related to a company and stock market event share instantly available to all stakeholders then the effects of those event share ready embed themselves in the stock price. So it is said that only the historical spot price carries the impact of all the other market events and can be employed to predict its future movement 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 and deploy it on cloud using stramlit. Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science

learning to predict stock values . Factors considered are open, close low, high and volume. The prediction of a stock market direction may serve as early recommendation system for short term investors and as early financial distress warning system for long term shareholders. This paper aims to build a deep learning short term memory model to achieve better prediction of stock market trends using correlated STI. This model is implemented in python using powerful deep learning tensor flow. We achieved mean prediction accuracy over number of stock datasets which is much higher than benchmark approaches. The prediction accuracy indicates that the proposed model with combination of technical analysis can provide often better results than the model build with benchmark ML algorithms. The stock data is usually non stationary and attributes are non correlative to each other. Several traditional stock technical indicators may incorrectly predict the stock market trends.

Contents

List of Figures

List of Abbreviations

Sr No.	Abbreviation	Expanded form
1	API	Application Programming Interface
2	GUI	Graphical User Interface
3	MVO	Mean Variation Optimization
4	DFD	Data Flow Diagram
5	CNN	Convolutional Neural Network
6	RMSE	Root Mean Square Error
7	LSTM	Long Short term Memory

Chapter 1

Introduction

Stock market prediction is the act of trying to determine the future value of a company stock or other financial instrument traded on an exchange. Stock market is the important part of economy of the company and plays a vital role in the growth of the industry and commerce of the country that eventually affects the economy of the company. the stock market is the primary source for any company to raise funds for business expansions. Stock market prediction will reveal the market patterns and predict the time to purchase stocks.

1.1 Description

The stock market is a vast array of investors and traders who buy and sell stock pushing the price up and 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 money in companies whose perceived value is expected to rise. The main functionality of our project is that it will help us to predict the actual and closing price of any stock along with the actual and predicted graph.The values could be consisting of high expected stock,low,open and closing stocks columns which will help us to predict the stock price earlier.With the help of the graph it will be easier for the stock viewers optimizing his/her stock in order to receive maximum profit. This project considers the stock data of various US based companies over a time span of many years. It takes into account attributes and the changes in stock prices measured over a one day span. Due to the sheer volume of money involved and number of transactions that takes every minute it will take time to be loaded. Thus by applying certain algorithms, we can build an user interface with reduced risks and increased returns.

1.2 Problem Formulation

Everyone wants to be rich in his life with low efforts and great advantages.Similarly we want to look into our future with inner most desire as we do not want to take risks or we want to decrease risk factor.Stock market is a place where selling and purchasing can provide future aims of life.An accurate prediction of future prices may lead to a higher yield of profit for investors through stock investments.The project target is to create web application that analyses previous stock data of companies and implement these values in data mining algorithm to determine the value that particular stock will have in near future with suitable accuracy.

1.3 Motivation

Stock Prediction is the process of selecting the best stocks available , out of the set to consider buying selling or keep hold of the stocks according to some objective. The objective typically maximizes factors such as expected return, and minimizes costs like financial risk.

Basic idea is for a given level of risk, we want to make sure that we are getting as much return as possible.

This method analyses historical data and attempts to approximate future values of a time series as a linear combination of these historical data.Online and batch learning algorithms differ in the way in which they operate. In an online algorithm, it is possible to stop the optimization process in the middle of a learning run and still train an effective model. This is particularly useful for very large data sets (such as stock price datasets) when the convergence can be measured and learning can be quit early. The stochastic learning paradigm is ideally used for online learning because the model is trained on every data point — each parameter update only uses a single randomly chosen data point, and while oscillations may be observed, the weights eventually converge to the same optimal value. The literature provides strong evidence that stock price values can be predicted from past price data. Principal component analysis (PCA) identifies a small number of principle components that explain most of the variation in a data set. This method is often used for dimensionality reduction and analysis of the data.Projecting the noisy observation onto a principle subspace results in a wellconditioned problem.Economics and stock prices are mainly reliant upon subjective perceptions about the stock market. It is nearimpossible 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.

Stock market prediction aims to determine the future movement of the stock value of a financial exchange. The accurate prediction of share price movement will lead to more profit investors can make. Predicting how the stock market will move is one of the most challenging issues due to many factors that involved in the stock prediction, such as interest rates, politics, and economic growth that make the stock market volatile and very hard to predict accurately.

1.4 Proposed Solution

In the proposed system we try to find the accurate value of the next day closing value that helps the investors to invest or sell their shares. Long Short Term Memory (LSTM) is an artificial neural network in the field of deep learning. LSTM is an advance Neural network with having a memory cell that stores a small amount of data for further references. LSTM has feedback links that make it a "general-purpose computer". LSTM can also process an entire series of data not only single value like image. Because of the dropout process which takes place in the LSTM algorithm, it is comparatively faster than SVM and Back propagation. LSTM algorithm is more suitable in predicting the future stock price than the SVM and Back propagation algorithm because of removing the undesired data. The time and memory consumption are also reduced when compared to the exciting system due to the dropout process.LSTM algorithm is more proper in handling non-linear data.

we are deploying our model on stremlit. Streamlit Cloud is a workspace **Streamlit** is an open-source app framework for Machine Learning and Data Science

The working of the LSTM learning algorithm is explained in the below steps:

Input:Historical stock price data.

Output:Prediction for stock prices based on the stock price variation 1. Start

2. Stock Data is taken and stored in a numpy array of 3 dimensions.

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 layers 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 in 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 providing test data as input to the network

9. End

Given below is a brief summary of the various terminologies relating to our proposed stock prediction system:

1. Training set : subsection of the original data that is used to train the neural network model for predicting the output values
2. Test set : part of the original data that is used to make predictions of the output value,

which are then compared with the actual values to evaluate the performance of the model.

3. Validation set : portion of the original data that is used to tune the parameters of the neural network model.
4. Activation function: in a neural network, the activation function of a node defines the output of that node as a weighted sum of inputs.
5. Batch size : number of samples that must be processed by the model before updating the weights of the parameters.
6. Epoch : a complete pass through the given dataset by the training algorithm.
7. Dropout: a technique where randomly selected neurons are ignored during training i.e., they are "dropped out" randomly. Thus, their contribution to the activation of downstream neurons is temporally removed on the forward pass, and any weight updates are not applied to the neuron on the backward pass.

8. Loss function : a function, defined on a data point, prediction and label, that measures a penalty such as square loss which is mathematically explained as f.

9. Cost function: a sum of loss functions over the training set. An example is the Mean Squared Error (MSE), which is mathematically explained as follows:

10. Root Mean Square Error (RMSE): measure of the difference between values predicted by a model and the values actually observed. It is calculated by taking the summation of the squares of the differences between the predicted value and actual value, and dividing it by the number of samples. It is mathematically expressed as follows: In general, smaller the RMSE value, greater the accuracy of the prediction made

1.5 Scope of the Project

This project focuses on maximizing profits by stock prediction. It will benefit the business sector greatly and it will help the users to make wise investment choices. To the field applied like finance, stock market, Small/large businesses companies with or without chains etc profits can be maximized depending on their attributes. The goal of stock prediction is to help organizations deliver maximum business value. Processes such as prioritization and managing resource capacity help us to "make the best or most effective use of human resources". In this way, organizations can increase business value delivery. This project focuses on maximizing profits by stock prediction. It will benefit the business sector greatly and it will help the users to make wise investment choices. To the field applied like finance, stock market, Small/large businesses companies with or without chains etc profits can be maximized depending on their attributes. The goal of stock prediction is to help organizations deliver maximum business value. Processes such as prioritization and managing resource capacity help us to "make the best or most effective use of human resources". In this way, organizations can increase business value delivery.

Chapter 2 Review of Literature

Using Neural Networks to Forecast Stock Market Prices, Ramon Lawrence.

This paper is a survey on the application of neural networks in forecasting stock market prices. With their ability to discover patterns in nonlinear and chaotic systems, neural networks offer the ability to predict market directions more accurately than current techniques. Common market analysis techniques such as technical analysis, fundamental analysis, and regression are discussed and compared with neural network performance. Also, the Efficient Market Hypothesis (EMH) is presented and contrasted with chaos theory and neural networks. Finally, future directions for applying neural networks to the financial markets are discussed [1].

Stock Market Prediction Using Hybrid Approach, Vivek Rajput,

The objective of this paper is to construct a model to predict stock value movement using the opinion mining and clustering method to predict National Stock Exchange (NSE). It used domain specific approach to predict the stocks from each domain and taken some stock with maximum capitalization. Topics and related opinion of shareholders are automatically extracted from the writings in a message board by utilizing our proposed strategy alongside isolating clusters of comparable sort of stocks from others using clustering algorithms. Proposed methodology will give two output set i.e. one from sentiment analysis and another from clustering based prediction with respect to some specialized parameters of stock exchange. By examining both the results an efficient prediction is produced. In this paper stocks with maximum capitalization within all the important sectors are taken into consideration for empirical analysis [2].

Hybrid ARIMA-BPNN Model for Time Series Prediction of the Chinese Stock Market,

Stock price prediction is a challenging task owing to the complexity patterns behind time series. Autoregressive integrated moving average (ARIMA) model and back propagation neural network (BPNN) model are popular linear and nonlinear models for time series forecasting respectively. The integration of two models can effectively capture the linear and nonlinear patterns hidden in a time series and improve forecast accuracy. In this paper, a new hybrid ARIMA-BPNN model containing technical indicators is proposed to forecast four individual stocks consisting of both main board market and growth enterprise market in software and information services sector [3].

Deep Learning for Stock Market Prediction Using Technical Indicators and Financial News Articles, Manuel R. Vargas, Carlos E. M. dos Anjos, Gustavo L G. Bichara, Alexandre G. Evsukoff. This work uses deep learning models for daily directional movements prediction of a stock price using financial news titles and technical indicators as input. A comparison is made between two different sets of technical indicators, set 1: Stochastic (%K), Stochastic (%D), Momentum, Rate of change, Williams (%R), Accumulation/Distribution (A/D) oscillator and Disparity 5; set 2: Exponential Moving Average, Moving Average Convergence-Divergence, Relative Strength Index, On Balance Volume and Bollinger Bands. Deep learning methods can detect and analyze complex patterns and interactions in the data allowing a more precise trading process. Experiments has shown that Convolutional Neural Network (CNN) can be better than Recurrent Neural Networks (RNN) on catching semantic from texts and RNN is better on catching the context information and modeling complex temporal characteristics for stock market forecasting. So, there are two models compared in this paper: a hybrid model composed by a CNN for the financial news and a Long Short-Term Memory (LSTM) for technical indicators, named as SI-RCNN; and a LSTM network only for technical indicators, named as I- RNN. The output of each model is used as input for a trading agent that buys stocks on the current day and sells the next day when the model predicts that the price is going up, otherwise the agent sells stocks on the current day and buys the next day. The proposed method shows a major role of financial news in stabilizing the results and almost no improvement when comparing different sets of technical indicators [4].

Financial Indices Modelling and Trading utilizing deep learning techniques, Marios Mourelatos, Thomas Amorgianiotis, Christos Alexakos, Spiridon Likothanassis.

Prediction and modelling of the financial indices is a very challenging and demanding problem because its dynamic, noisy and multivariate nature. Modern approaches have also to challenge the fact that they are dependencies between different global financial indices. All this complexity in combination with the large volume of historic financial data raised the need for advanced machine learning solutions to the problem. This article proposes a Deep Learning approachutilizing Long Short-Term Memory (LSTM) Networks for the modelling and trading of financial indices [5].

Hybrid Deep Learning Models for Stock Prediction, Mohammad Asiful Hossain, Rezaul Karim, Ruppa Thulasiram, Neil D B. Bruce, Yang Wang.

Stock market prediction has always caught the attention of many analysts and researchers. Popular theories suggest that stock markets are essentially a random walk and it is a fools game to try and predict them. Predicting stock prices is a challenging problem in itself because of the number of variables which are involved. This paper reviews all these points

The stock returns is an area of study wherein many research scholars have shown immense interest for past several years. A brief review of literature will help in understanding the relevance of the content analysis in the area of stock returns. The researches in social sciences or in the field of economics depend in one way or the other on careful reading of written materials and the research work done by many research scholars on similar subjects. Considering this fact, the importance of content analysis becomes very significant. Barelson (1952) defined content analysis as a technique of research that is systematic representation of the matter of communication. According to Stone (1964), the content analysis is a methodology or procedure which can be used to access particular information based on the past references. The definition of content analysis requires that the inference be derived
from the counts of frequency to place a number of standard methods on the borderline of acceptability (Leites & Poo, 1942). The various areas to which the technique of content analysis can be applied is based on theinvolved using both the qualitative as well as quantitative measures for analyzing the literature relating to stock returns. The important determinants or factors of stock returns are analyzed first qualitatively using the abstracts, introduction, literature review, methodology, analysis and conclusions of the selected 368 research papers. Further analysis has been performed using frequency, counts and percentages to find out the other important aspects like appearance in journals, number of authors, and contribution of authors country-wise and appearance of authors in the select research papers.

Chapter 3 System Analysis

3.1 Functional Requirements

The various functional requirements of the system can be summarized as:

- A home page that is user friendly and descriptive.
- Client can add any foreign stock symbol.
- Client has to insert start date.
- Client has to insert end date and date to predict the close price.
- Client can view the high, low, open, close, volume and adjacent Close records recorded ●

Client can view the close price graph which will give a idea of how that market is flowing. ●

Client can view the graph of predicted values as well.

- Client can compare of how accurate the model has predicted close price upto a year information.

- System can display the predicted close price.
- System can display the close price the client has displayed date of.

3.2 Non-Functional Requirements

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to another. They are also called non-behavioral requirements. • The processing of each request should be done within 10 seconds.

• The site takes a few seconds more to predict close price depending on how large the client has entered the input.

• The system should provide better accuracy and optimized results.

• Easy to use and user friendly.

3.3 **Specific Requirements**

3.3.1. User interfaces.
The new system shall provide a very intuitive and simple interface to the client as well as investor or the visitors, so that the user can easily navigate through software. The client can
Check the stock's adjacent close, volume etc recorded. It can compare the close and predicted price of the date entered by the user. The user can check the flow of the market with the help of close price graph as well as predicted price graph.

3.3.2. Hardware interfaces.

Hardware

The hardware environment consists of the following:

CPU : Intel Pentium IV 600MHz or above

a)

Mother Board : Intel 810 or above

Hard disk space : 20GB or more

Display : Color Monitor

Memory : 128 MB RAM

Other Devices : Keyboard, mouse.

Server side

The web application will be hosted on a web server which is listening on the web standard port, port 80. b) Client side

Monitor screen – the software shall display information to the user via the monitor screen Mouse – the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus.

Keyboard – the software shall interact with the keystrokes of the keyboard. The keyboard will input data into the active area of the database.

3.3.3. Software interfaces.

Technically the system will run on any OS having Web Browser.

Development Tools:

Front End: Steamlit

Back End: Python Connection

Operating System: Windows 10

Web server :

The actual program that will perform the operations is written in Python. All data will be stored in a database.

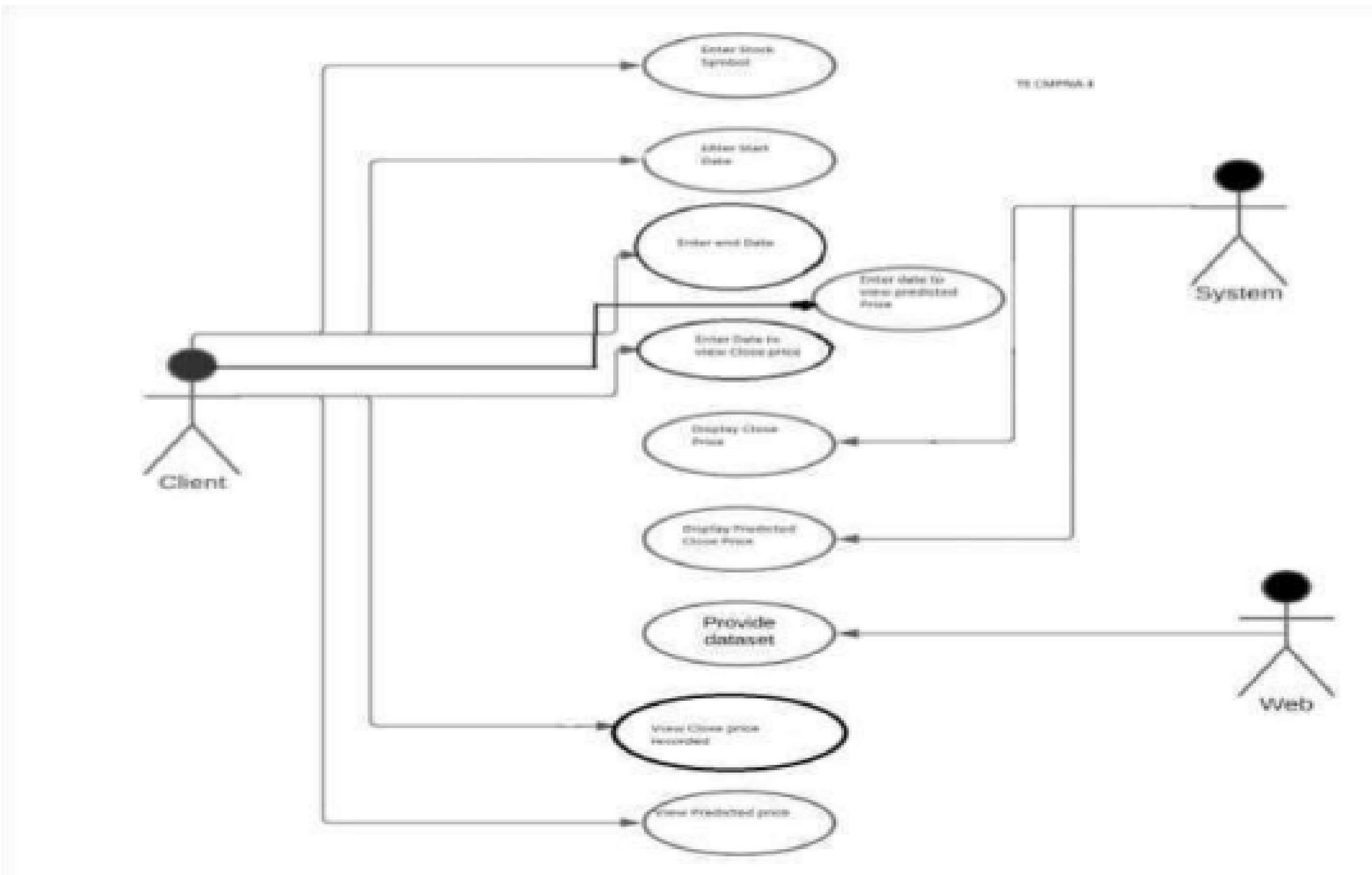
3.3.4. Communications interfaces

The HTPP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

3.4 Use-Case Diagrams and Description **<u>UseCase</u> <u>diagram:</u>**

Fig 3.1 Use case diagram for stock market prediction

<u>Use-Case Description</u>:



In this Use Case diagram, we have three actors Client, Web and System. The web provides us with the dataset of the companies and stock details. The Client can Enter Stock symbol, Enter start date, Enter end date, View predicted of the date entered in the End date bar and View close Price The system can Display Close price and Display predicted price so that the client can compare.

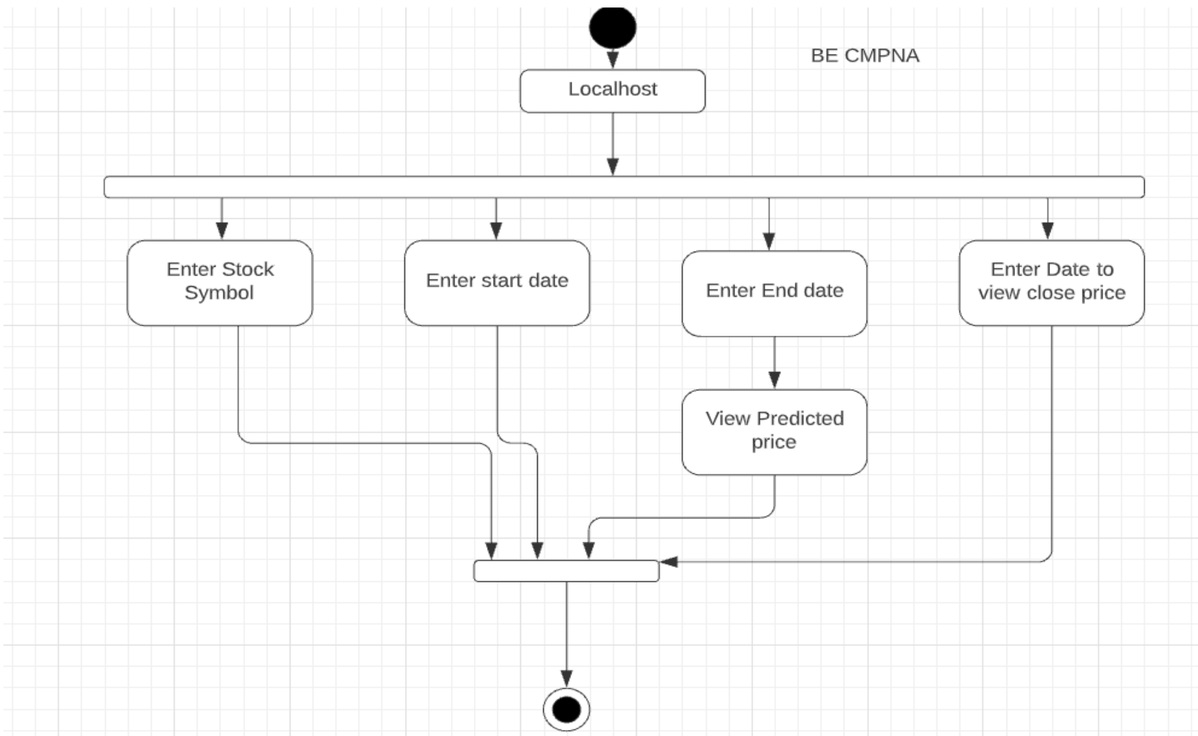
Chapter 4 Analysis Modeling

4.1Activity Diagram

Activity diagram is flow of functions without trigger mechanism, state machine is consisting of triggered states.

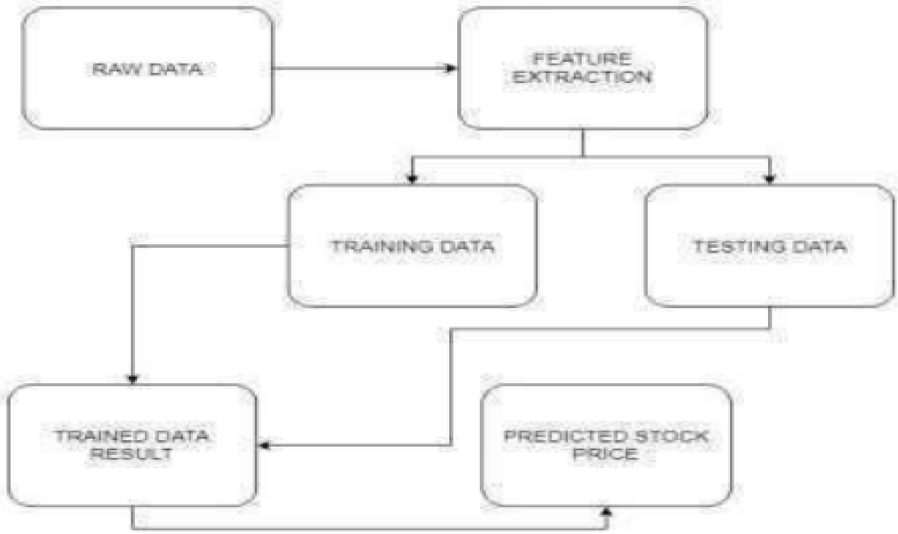
Following Activity diagram created for use cases like Enter Stock Symbol, enter start date, enter end date, view predicted price under End date and enter date to view close price for Stock market prediction.

Fig 4.1 Activity Diagram for stock market prediction



4.2 Functional Modelling

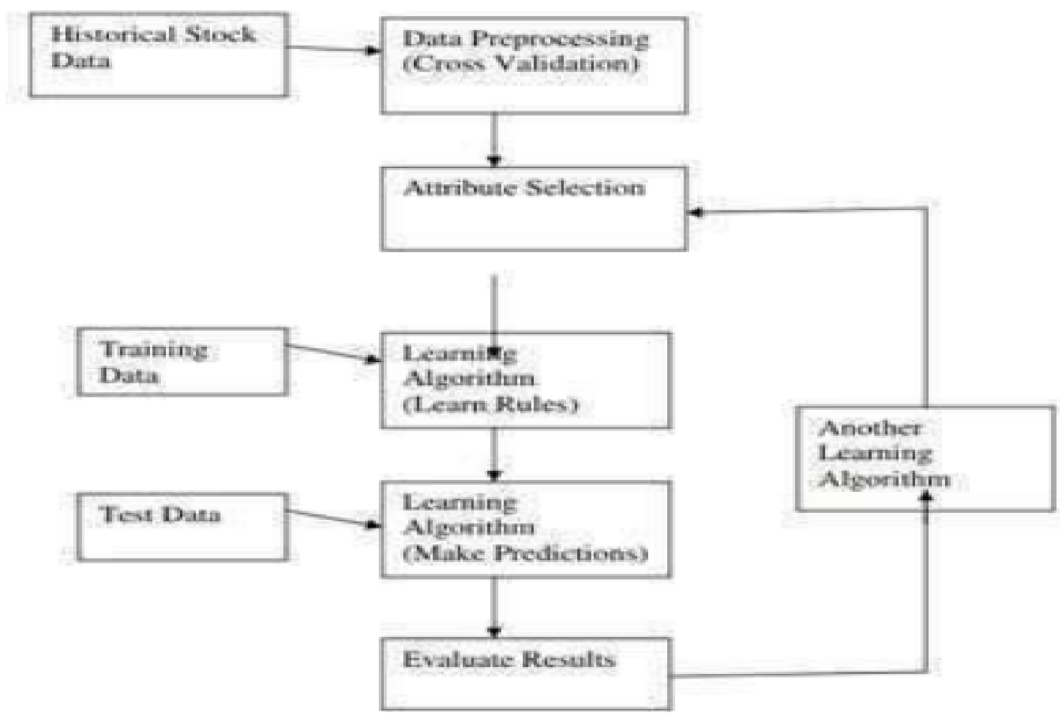
Fig 4.2 Data Flow diagram for Stock Market prediction



Chapter 5 Design

5.1 Architectural Design :

Fig 5.1 Block diagram for stock Market prediction



Description:

I. Data Collection Data collection is a very basic module and the initial step towards the project. It generally deals with the collection of the right dataset. The dataset that is to be used in the market prediction has to be used to be filtered based on various aspects. Data collection also complements to enhance the dataset by adding more data that are external. Our data mainly consists of the previous year stock prices. Initially, we will be analysing the dataset and according to the accuracy, we will be using the model with the data to analyse the predictions accurately.

II.Pre Processing Data pre-processing is a part of data mining, which involves transforming raw data into a more coherent format. Raw data is usually, inconsistent or incomplete and usually contains many errors. The data preprocessing involves checking out for missing values, looking for categorical values, splitting the data-set into training and test set and finally do a feature scaling to limit the range of variables so that they can be compared on common environs.

III. Training the Machine Training the machine is similar to feeding the data to the algorithm to touch up the test data.

IV. The training sets are used to tune and fit the models. The test sets are untouched, as a model should not be judged based on unseen data. The training of the model includes cross-validation where we get a well-grounded approximate performance of the model using the training data.

5.2 User Interface Design:

Weblink:

Fig:5.2.1 Overview History of entered Stock

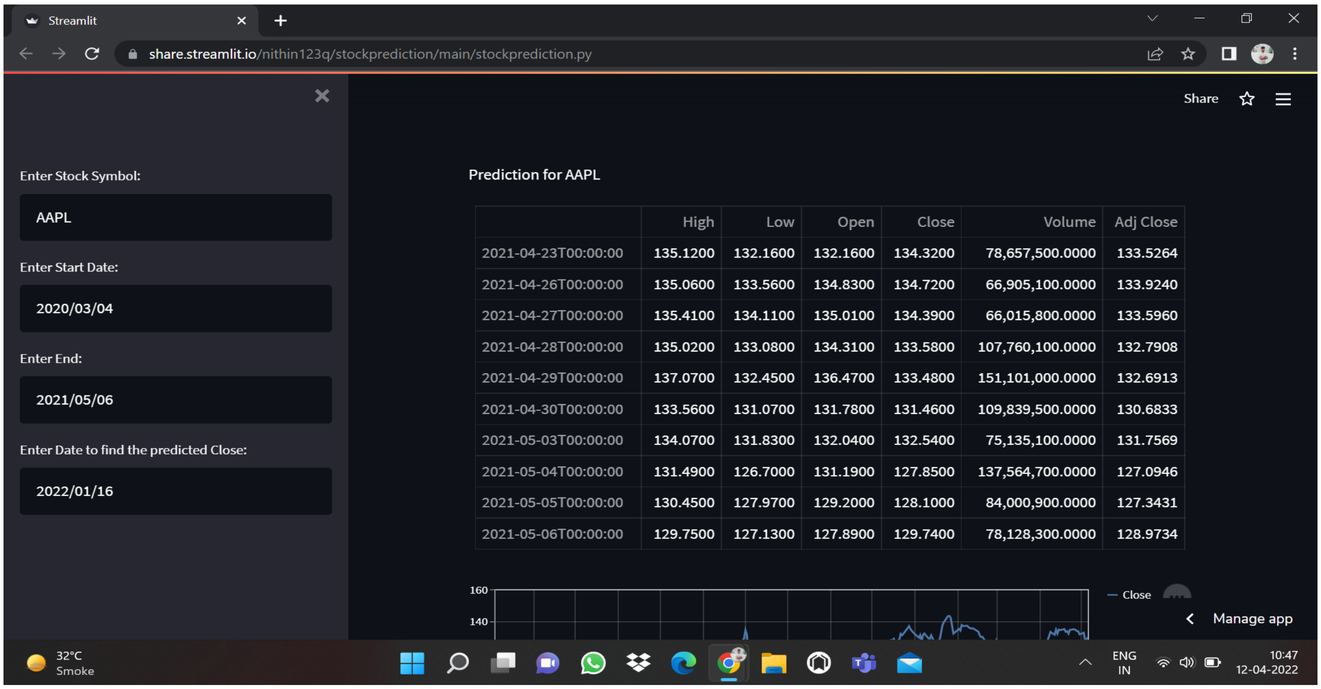
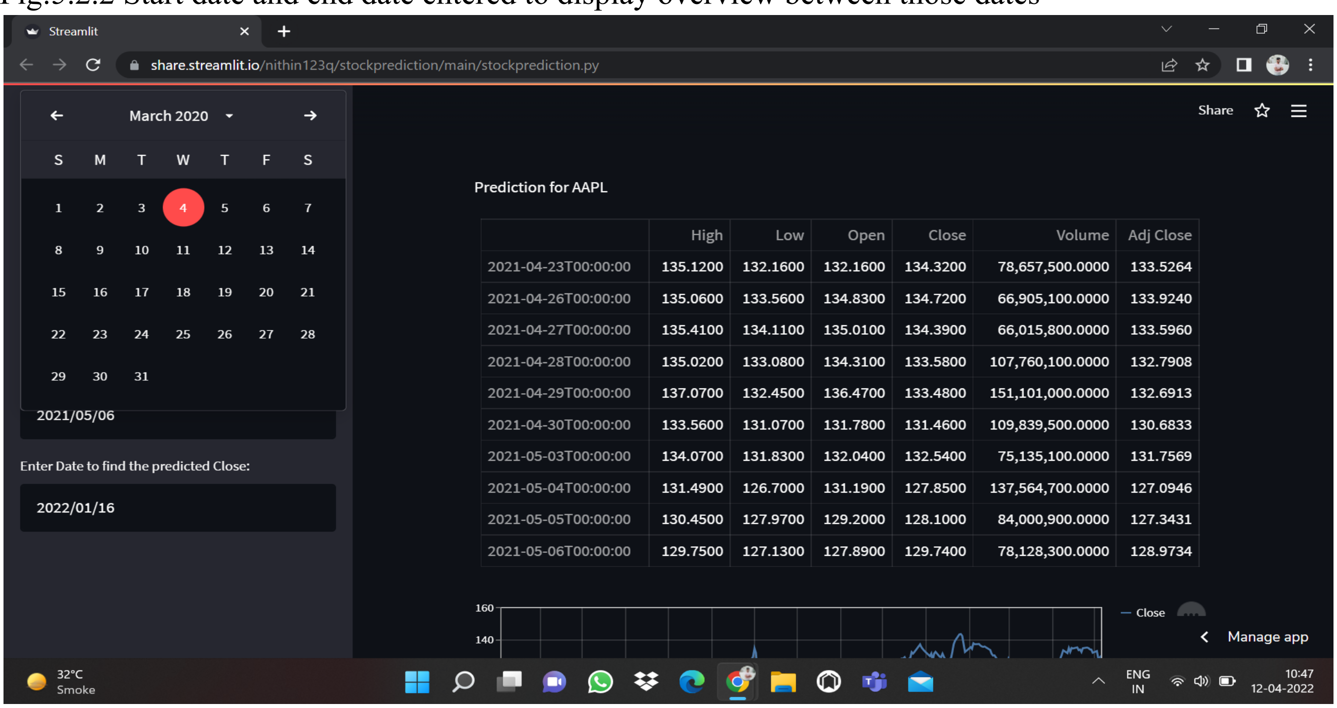


Fig:5.2.2 Start date and end date entered to display overview between those dates



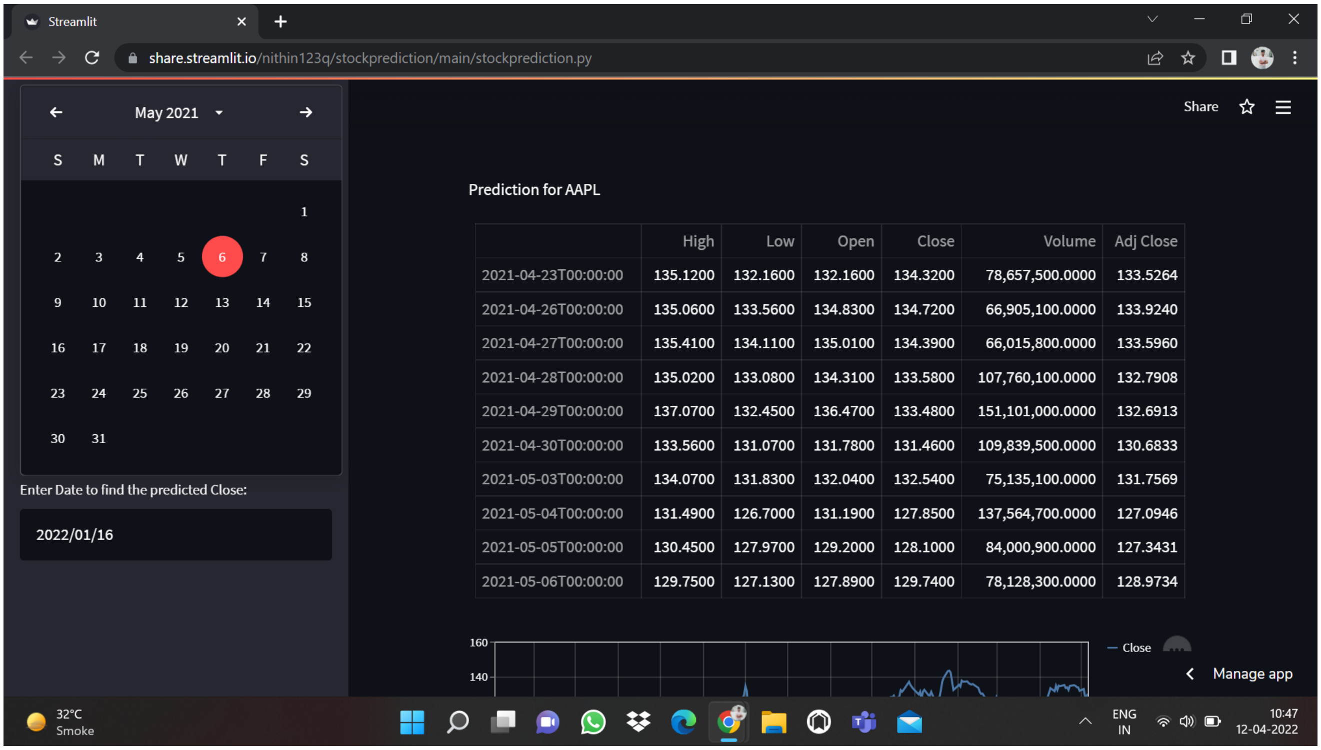


Fig:5.2.3 Close Price Graph is displayed.

Fig:5.2.4: The predicted Close graph is shown

Fig:5.2.5 The difference between Close and Prediction is shown

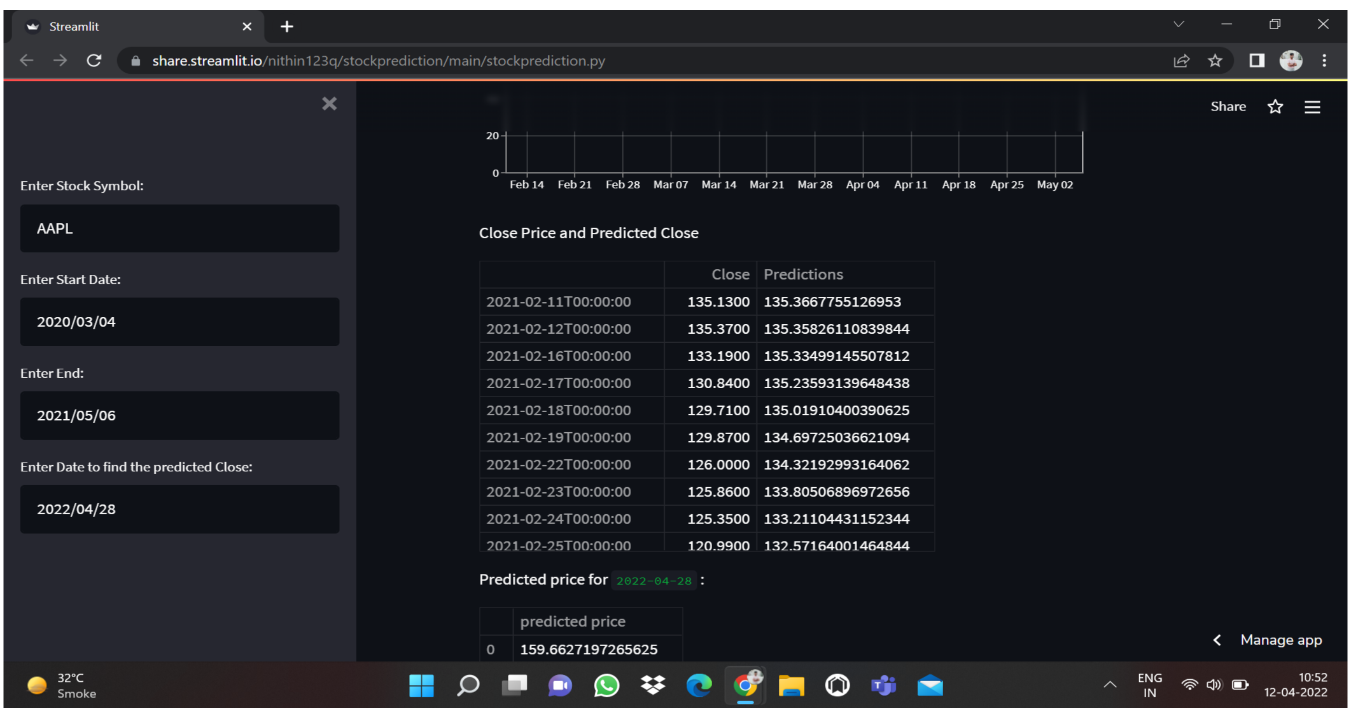
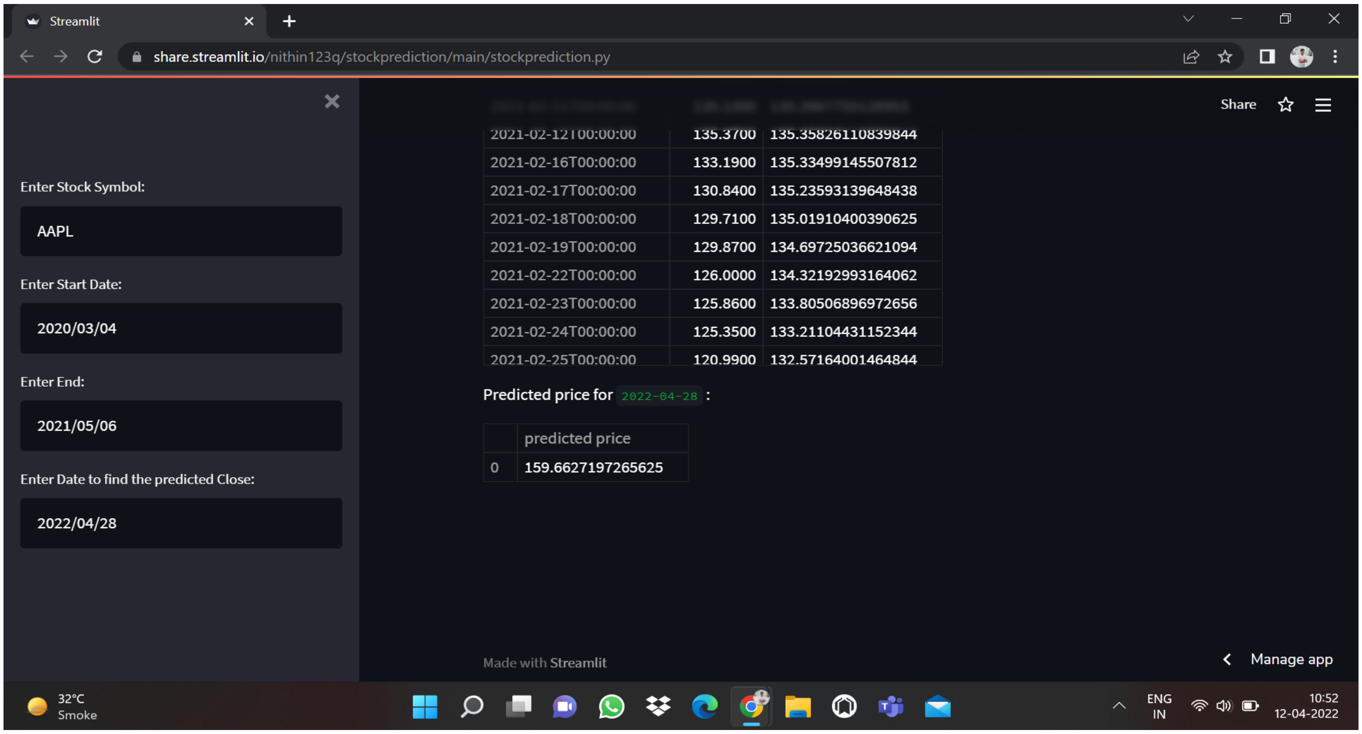


Fig: 5.5.6 The user can enter date to view predicted price as well as close price



Chapter 6 Implementation

6.1 Algorithms/Methods used

In this project we have used LSTM i.e long short term memory model for our stock prediction.

Long short term memory model is an artificial neural network architecture usd in the field of deep learning and unlike standard feet for neural network

Long short term memory has feedback connections

It can not only process single data but also entire sequence of data.

Lstm model stores past information and forgets informations that is not important

Fig 6.1.1 LSTM model set

```
[ ] #Build the LSTM model
    model = Sequential()
    model.add(LSTM(50, return_sequences=True, input_shape= (x_train.shape[1], 1)))
    model.add(LSTM(50, return_sequences= False))
    model.add(Dense(25))
    model.add(Dense(1))
```

Algorithm for LSTM model

Input:Historical stock price data.

Output:Prediction for stock prices based on the stock price variation

1. Start

a. Stock Data is taken and stored in a numpy array of 3 dimensions.

Where

N is number of training sequences

W is sequence length

F is the number of features of each sequence

b. 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 layers and a single layer with a linear activation function.

c. Train the constructed network on the data

d. Use the output of the last layer as prediction in the next time step

e. Repeat steps 4 and 5 until optimal convergence is reached

f. Obtain predictions byproviding test data as input to the network

g. Evaluate accuracy by providing test data as input to the network h. End

<u>Implementation of RMSE model</u>

This model takes the difference for each observed and predicted value.you can swap the order of substractions because the next step is always square of the difference.This is because the square of the negative value will always be a positive value.

Fig 6.1.2 RMSE model set

```
#Get the root mean squarred error (RMSE)
rmse = np.sqrt( np.mean( predictions - y_test )**2 )
rmse
```

6.2 Working of the project

```
#Import the libraries import streamlit as st
import datetime as dt
import math
import pandas_datareader as web
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import Dense, LSTM
import matplotlib.pyplot as plt
import streamlit as st

#To find closing price of particular day
def pred(symbol):
    current_d=st.sidebar.date_input("Enter Date to view close price:")
    st.write('Close price of the entered date:')
    ds = symbol
    apple_quote2 = web.DataReader(ds, data_source='yahoo', start=current_d, end=current_d)
    st.write(apple_quote2['Close'])

#prediction Function
def prediction(symbol, start, end):
    ds = symbol
    start_d = start
    end_d = end
    df = web.DataReader(ds, data_source='yahoo', start=start_d, end = end_d)
    #Show teh data
    tail_s = df.tail(10)
    st.table(tail_s)
    #shape
    #Visualize the closing price history
    chart_data = pd.DataFrame(
    df, columns=['Close'])
    st.line_chart(chart_data)

    #Create a new dataframe with only the 'Close column
    data = df.filter(['Close'])
    #Convert the dataframe to a numpy array
    dataset = data.values
    #Get the number of rows to train the model on
    training_data_len = math.ceil( len(dataset) * .8 )
    #Scale the data
    scaler = MinMaxScaler(feature_range=(0,1))
    scaled_data = scaler.fit_transform(dataset)
    #Create the training data set
```

```
# #Create the scaled training data set
train_data = scaled_data[0:training_data_len , :]
#Split the data into x_train and y_train data sets
x_train = []
y_train = []
for i in range(60, len(train_data)):
    x_train.append(train_data[i-60:i, 0])
    y_train.append(train_data[i,0])
#Convert the x_train and y_train to numpy arrays
x_train, y_train = np.array(x_train), np.array(y_train)
#Reshape the data
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
x_train.shape
#Build the LSTM model
model = Sequential()
model.add(LSTM(50, return_sequences=True, input_shape= (x_train.shape[1], 1)))
model.add(LSTM(50, return_sequences= False))
model.add(Dense(25))
model.add(Dense(1))
#Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')

#Train the model
model.fit(x_train, y_train, batch_size=1, epochs=1)
#Create the testing dataset
#Create the new array containing scaled values from index 1802 to 2003
test_data = scaled_data[training_data_len - 60: , :]
#Create the data sets x_test and y_test
x_test = []
y_test = dataset[training_data_len:, :]
for i in range(60, len(test_data)):
    x_test.append(test_data[i-60:i, 0])
    #Convert the data to a numpy array
x_test = np.array(x_test)
#Reshape the data
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1 ))
#Get the models prdicted price values
predictions = model.predict(x_test)
predictions = scaler.inverse_transform(predictions)
#Get the root mean squarred error (RMSE)
rmse = np.sqrt( np.mean( predictions - y_test )**2 )
#Plot the data
train = data[:training_data_len]
valid = data[training_data_len:]
valid['Predictions'] = predictions
#Visualize the data plt.figure(figsize=(16,8))
```

```
    plt.title('Model')
    plt.xlabel('Date', fontsize=18)
    plt.ylabel('Close Price USD ($)', fontsize=18)
    plt.plot(train['Close'])
    plt.plot(valid[['Close', 'Predictions']])
    plt.legend(['Train','Val','Predictions'], loc= 'lower right')
    plt.show()
    predict_chart_data = pd.DataFrame(valid, columns=['Close', 'Predictions'])
    st.line_chart(predict_chart_data)

    #Show the valid and predicted prics
    st.write('Close Price and Predicted Close ')
    st.write(valid)
    #Get the quote to find the predicted CLOSE price
    p_end = st.sidebar.date_input("Enter Date to find the predicted Close:", dt.date(2022, 1,16))
    apple_quote = web.DataReader(symbol, data_source='yahoo',start=start_d, end=p_end)
    #Create a new dataframe
    new_df = apple_quote.filter(['Close'])
    #Get teh last 60 daysclosing price values and convert the dataframe to an array
    last_60_days = new_df[-60:].values
    #Scale the data to be values between 0 and 1
    last_60_days_scaled = scaler.transform(last_60_days)
    #Create an empty list
    X_test = []
    #Append teh past 60 days
    X_test.append(last_60_days_scaled)
    #Convert the X_test data set to a numpy array
    X_test = np.array(X_test)
    #Reshape the data
    X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
    #Get the predicted scaled price
    pred_price = model.predict(X_test)
    #undo the scaling
    pred_price = scaler.inverse_transform(pred_price)
    labels = ['predicted price']
    pred_price = pd.DataFrame(pred_price, columns=labels)
    st.write("Predicted price for",p_end,":",pred_price)

    #working of streamlit code
start_again = 0
end_again = 0
symbol = st.sidebar.text_input("Enter Stock Symbol:", value="AAPL")
start = st.sidebar.date_input("Enter Start Date:", dt.date(2020, 3, 4))
end = st.sidebar.date_input("Enter End:", dt.date(2021, 5, 6))

#Main screen
```

```
if symbol == '':
    st.write("Select Stock")
else:
    st.write("Prediction for ", symbol)
#Function for Prediction
if start == start_again and end == end_again:
    print('repeated')
else:
    start_again = start
    end_again = end
prediction(symbol, start, end)
# Function to get current value pred(symbol)
```

Chapter 7 Conclusion

- In this project, we tried to develop a prediction model for the stock market based on the technical analysis using LSTM stock market data.
- LSTM model are very powerful in sequence prediction problems because they are able to store past information.
- This could guide the future investors in the stock market to make profitable investment decisions whether to buy or sell or hold a share.
- Stock markets are hard to monitor and require plenty of context when try to interpret the movement and predict prices.
- They are able to keep track of context specific temporal dependencies between stock prices for a longer period of time while performing predictions.

References

Rather A.M., Agarwal A., Sastry V.N. Recurrent neural network and a hybrid model for prediction of stock returns Expert Systems with Applications.

Zhang G., Patuwo B.E., Hu M.Y. Forecasting with artificial neural networks: The state of the art International journal of forecasting

https://www.analyticsvidhya.com/blog/2020/10/create-interactive-dashboards-with-streamlit-andpython/
https://www.youtube.com/watch?v=B2iAodr0fOo

https://docs.streamlit.io/en/stable/api.html?highlight=altair_chart#display-data

Acknowledgements

We express our deep sense of gratitude to our project guide Mrs. Rupesh Mishra for encouraging us and guiding us throughout this project. We were able to successfully complete this project with the help of her deep insights into the subject and constant help.

We are very much thankful to Dr. .Kavita Sonawane, HOD of the COMPS department at St. Francis Institute of Technology for providing us with the opportunity of undertaking this project which has led to us learning so much in the domain of Cloud computing.

Last but not the least we would like to thank all our peers who greatly contributed to the completion of this project with their constant support and help.