A

Major Project

On

STOCK MARKET PREDICTION USING MACHINE LEARNING

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

BY

J.NAVANEETH (187R1A05E1)
K.SHIVADEEP (187R1A05E4)
J.PAVAN (187R1A05E9)

Under the Guidance of

R.DILEEP KUMAR

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)
Recognized Under Section 2(f) & 12(B) of the UGC Act.1956,
Kandlakoya (V), Medchal Road, Hyderabad-501401.
2018-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**STOCK MARKET PREDICTION USING MACHINE LEARNING**" being submitted by **J.NAVANEETH (187R1A05E1), K.SHIVADEEP (187R1A05E4), J.PAVAN(187R1A05E9)** in partial fulfillment of the requirements for the award of the degree of B. Tech in Computer Science and Engineering of the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him under our guidance and supervision during the year 2021-2022.

R. Dileep Kumar
Assistant Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted on viva voce Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **R.DILEEP KUMAR** Assistant Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) Coordinators: **Mr. J. Narasimha Rao, Dr. T.S.Mastan Rao , Mr A.Uday Kiran , Mr.A.Kiran Kumar , Mrs G.Latha** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to the Head of the Department **Dr. K. Srujan Raju** for providing excellent infrastructure and a nice atmosphere for completing this project successfully.

We are obliged to our Director **Dr. A. Raji Reddy** for being cooperative throughout the course of this project. We would like to express our sincere gratitude to our Chairman Sri. **Ch. Gopal Reddy** for his encouragement throughout the course of this project

The guidance and support received from all the members of **CMR TECHNICAL CAMPUS** who contributed and who are contributing to this project, was vital for the success of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family members for their constant encouragement without which this assignment would not be possible. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project.

J.NAVANEETH (187R1A05E1)
K.SHIVADEEP (187R1A05E4)
J. PAVAN (187R1A05E9)

ABSTRACT

Stock market is place where people buy and sell shares of publicly listed companies. Every buyer and seller try to predict the stock market price movements to get maximum profits and minimum losses. Using cutting edge technology such as AI can improve prediction stock price. The stock market is characterized by extreme fluctuations, non-linearity, and shifts in internal and external environmental variables. Artificial intelligence (AI) techniques can detect such non-linearity, resulting in much-improved forecast results.

The application of Artificial Intelligence (AI) to financial investment is a research area that has attracted extensive research attention since the 1990s, when there was an accelerated technological development and popularization of the personal computer. Since then, countless approaches have been proposed to deal with the problem of price prediction in the stock market. This paper presents a systematic review of the literature on Artificial Intelligence applied to investments in the stock market based on a sample of 2326 papers from the Scopus website between 1995 and 2019. This culminating experience project used artificial intelligence (AI) technology to forecast and analyse the stock market and construct complex nonlinear relationships between the input data and the output data. These papers were divided into four categories: portfolio optimization, stock market prediction using AI, financial sentiment analysis, and combinations involving two or more approaches. For each category, the initial introductory research to its state-of-the-art applications is described. In addition, an overview of the review leads to the conclusion that this research area is gaining continuous attention and the literature is becoming increasingly specific and thorough.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

Researchers have been studying different methods to effectively predict the stock market price. 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. One such method is to use machine learning algorithms for forecasting. This project's objective is to improve the quality of output of stock market predicted by using stock value.

1.2 PROJECT PURPOSE

The future price of a stock is the main motivation behind the stock price prediction. In various cases like business and industry, environmental science, finance and economics motivation can be useful. The future value of the company's stock can be determining

1.3 PROJECT FEATURES

In this proposed system, we focus on predicting the stock values using machine learning algorithms like Random Forest and Support Vector Machines. We proposed the system "Stock market price prediction" we have predicted the stock market price using the random forest algorithm. we were able to train the machine from the various data points from the past to make a future prediction. We took data from the previous year stocks to train the model. We majorly used two machine-learning libraries to solve the problem. The first one was numpy, which was used to clean and manipulate the data, and getting it into a form ready for analysis. The other was scikit, which was used for real analysis and prediction. The dataframe features were date and the closing price for a particular day. We used all these features to train the machine on random forest model and predicted the object variable, which is the price for a given day.

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified.

2.1 PROBLEM DEFINITION

Stock market prediction is basically defined as trying to determine the stock value and offer a robust idea for the people to know and predict the market and the stock prices. It is generally presented using the quarterly financial ratio using the dataset. Thus, relying on a single dataset may not be sufficient for the prediction and can give a result which is inaccurate. Hence, we are contemplating towards the study of machine learning with various datasets integration to predict the market and the stock trends. The problem with estimating the stock price will remain a problem if a better stock market prediction algorithm is not proposed. Predicting how the stock market will perform is quite difficult. The movement in the stock market is usually determined by the sentiments of thousands of investors. Stock market prediction, calls for an ability to predict the effect of recent events on the investors. These events can be political events like a statement by a political leader, a piece of news on scam etc. It can also be an international event like sharp movements in currencies and commodity etc. All these events affect the corporate earnings, which in turn affects the sentiment of investors. It is beyond the scope of almost all investors to correctly and consistently predict these hyperparameters. All these factors make stock price prediction very difficult. Once the right data is collected, it then can be used to train a machine and to generate a predictive result.

2.2 EXISTING METHODS

2.2.1 Stock Market Prediction Using Machine Learning

In the finance world stock trading is one of the most important activities. Stock market prediction is an act of trying to determine the future value of a stock other financial instrument traded on a financial exchange. This paper explains the prediction of a stock using Machine Learning. The technical and fundamental or the time series analysis is used by the most of the stockbrokers while making the stock predictions. The programming language is used to predict the stock market using machine learning is Python. In this paper we propose a Machine Learning (ML) approach that will be trained from the available stocks data and gain intelligence and then uses the acquired knowledge for an accurate prediction. In this context this study uses a machine learning technique called Support Vector Machine (SVM) to predict stock prices for the large and small capitalizations and in the three different markets, employing prices with both daily and up-to-the-minute frequencies.

2.2.2 Forecasting the Stock Market Index Using Artificial Intelligence Techniques

The research work done by Lufuno Ronald Marwala A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering. The weak form of Efficient Market hypothesis (EMH) states that it is impossible to forecast the future price of an asset based on the information contained in the historical prices of an asset. This means that the market behaves as a random walk and as a result makes forecasting impossible. Furthermore, financial forecasting is a difficult task due to the intrinsic complexity of the financial system. The objective of this work was to use artificial intelligence (AI) techniques to model and predict the future price of a stock market index. Three artificial intelligence techniques, namely, neural networks (NN), support vector machines and neuro-fuzzy systems are implemented in forecasting the future price of a stock market index based on its historical price information. Artificial intelligence techniques have the ability to take into consideration financial system complexities and they are used as financial time series forecasting tools.

2.2.3 Indian stock market prediction using artificial neural networks on tick data

The research work done by Dharmaraja Selvamuthu, Vineet Kumar and Abhishek Mishra Department of Mathematics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi 110016, India. A stock market is a platform for trading of a company's stocks and derivatives at an agreed price. Supply and demand of shares drive the stock market. In any country stock market is one of the most emerging sectors. Nowadays, many people are indirectly or directly related to this sector. Therefore, it becomes essential to know about market trends. Thus, with the development of the stock 5 market, people are interested in forecasting stock price. But, due to dynamic nature and liable to quick changes in stock price, prediction of the stock price becomes a challenging task. Stock m Prior work has proposed effective methods to learn event representations that can capture syntactic and semantic information over text corpus, demonstrating their effectiveness for downstream tasks such as script event prediction. On the other hand, events extracted from raw texts lacks of common-sense knowledge, such as the intents and emotions of the event participants, which are useful for distinguishing event pairs when there are only subtle differences in their surface realizations. To address this issue, this paper proposes to leverage external common-sense knowledge about the intent and sentiment of the event.

2.2.4 The Stock Market and Investment

The research work done by Manh Ha Duong Boriss Siliverstovs. Investigating the relation between equity prices and aggregate investment in major European countries including France, Germany, Italy, the Netherlands and the United Kingdom. Increasing integration of European financial markets is likely to result in even stronger correlation between equity prices in different European countries. This process can also lead to convergence in economic development across European countries if developments in stock markets influence real economic components, such as investment and consumption. Indeed, our vector autoregressive models suggest that the positive correlation between changes equity prices and investment is, in general, significant. Hence, 6 monetary authorities should monitor reactions of share prices to monetary policy and their effects on the business cycle.

2.2.5 LIMITATIONS OF EXISTING SYSTEM

- The existing system fails when there are rare outcomes or predictors, as the algorithm is based on bootstrap sampling.
- The previous results indicate that the stock price is unpredictable when the traditional classifier is used.
- The existence system reported highly predictive values, by selecting an appropriate time period for their experiment to obtain highly predictive scores.
- The existing system does not perform well when there is a change in the operating environment.
- It doesn‟t focus on external events in the environment, like news events or social media.
- It exploits only one data source, thus highly biased.

2.3 PROPOSED SYSTEM

In this proposed system, we focus on predicting the stock values using machine learning algorithms like Random Forest and Support Vector Machines. We proposed the system "Stock market price prediction" we have predicted the stock market price using the random forest algorithm. In this proposed system, we were able to train the machine from the various data points from the past to make a future prediction. We took data from the previous year stocks to train the model. We majorly used two machine-learning libraries to solve the problem. The first one was numpy, which was used to clean and manipulate the data, and getting it into a form ready for analysis. The other was scikit, which was used for real analysis and prediction. The data set we used was from the previous years stock markets collected from the public database available online, 80 % of data was used to train the machine and the rest 20 % to test the data. The basic approach of the supervised learning model is to learn the patterns and relationships in the data from the training set and then reproduce them for the test data. We used the python pandas library for data processing which combined different datasets into a data frame. The tuned up dataframe allowed us to prepare the data for feature extraction. The dataframe features were date and the closing price for a particular day. We used all these features to train the machine on random forest model and predicted the object variable, which is the price for a given day.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- The investors are more confident to invest on stocks according to the predictions.
- The using of different algorithms according to the events to give the accurate prediction.
- Multiple algorithms will be applied at the moment of event.

2.3.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the user.

Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give the user the best quality of life possible. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend proposed system. Also all the resources are already available, it give indication of the system economically development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

- Processor : Intel Dual Core@ CPU 2.90GHz
- RAM : 1GB and above.
- Space on Hard Disk : 4GB

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements.

- Operating System : Windows 8
- Backend : Python 3.7
- Database : SQLite
- IDE : Google Colab

3.ARCHITECTURE

3.ARCHITECTURE

3.1 PROJECT ARCHITECTURE

A structure chart (SC) in software engineering and organizational theory is a chart which shows the breakdown of a system to its lowest manageable levels. They are used in structured programming to arrange program modules into a tree. Each module is represented by a box, which contains the module's name.

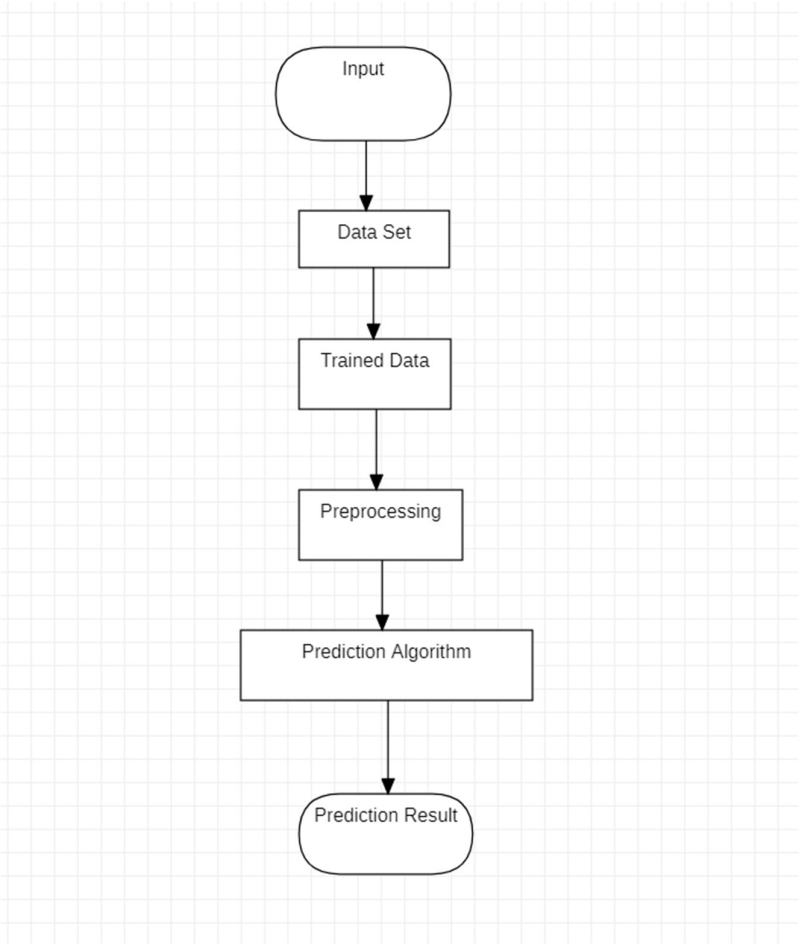


Fig 3.1: Training and Prediction

3.2 UML Diagrams

A UML diagram is a partial graphical representation (view) of a model of a system under design, implementation, or already in existence. UML diagram contains graphical elements (symbols) – UML nodes connected with edges (also known as paths or flows) – that represent elements in the UML model of the designed system. The UML model of the system might also contain other documentation such as use cases written as templated texts.

The kind of the diagram is defined by the primary graphical symbols shown on the diagram. For example, a diagram where the primary symbols in the contents area are classes is class diagram. A diagram which shows use cases and actors is use case diagram. A sequence diagram shows sequence of message exchanges between lifelines.

UML specification does not preclude mixing of different kinds of diagrams, e.g. to combine structural and behavioral elements to show a state machine nested inside a use case. Consequently, the boundaries between the various kinds of diagrams are not strictly enforced. At the same time, some UML Tools do restrict set of available graphical elements which could be used when working on specific type of diagram.

UML specification defines two major kinds of UML diagram: structure diagrams and behavior diagrams.

Structure diagrams show the static structure of the system and its parts on different abstraction and implementation levels and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.

Behavior diagrams show the dynamic behavior of the objects in a system, which can be described as a series of changes to the system over time.

3.2.1 Use Case Diagram

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

• Scenarios in which system or application interacts with people, organizations, or external systems.

• Goals that your system or application helps those entities (known as actors) achieve.

• The scope of system.

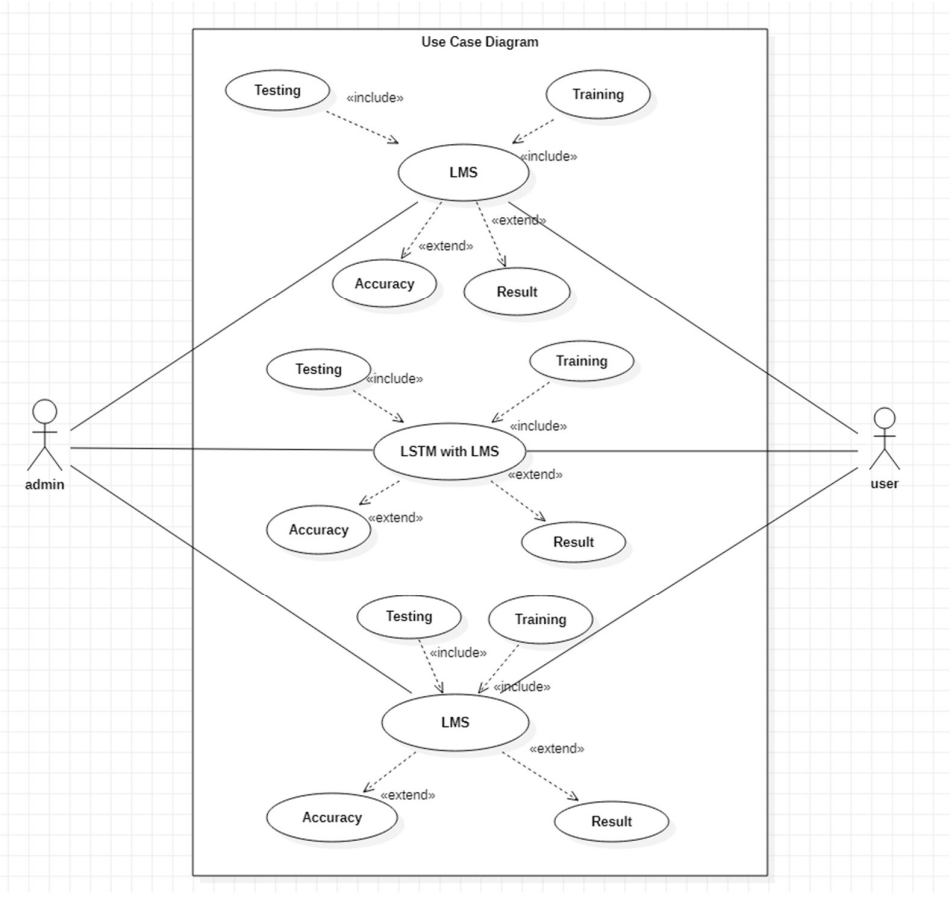


Fig 3.2.1 : Using LMS, LSTM and LSTM with LMS in the system

3.2.2 Sequence Diagram

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Sequence diagrams can be useful references for businesses and other organizations. Try drawing a sequence diagram to:

Represent the details of a UML use case.

- Model the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.

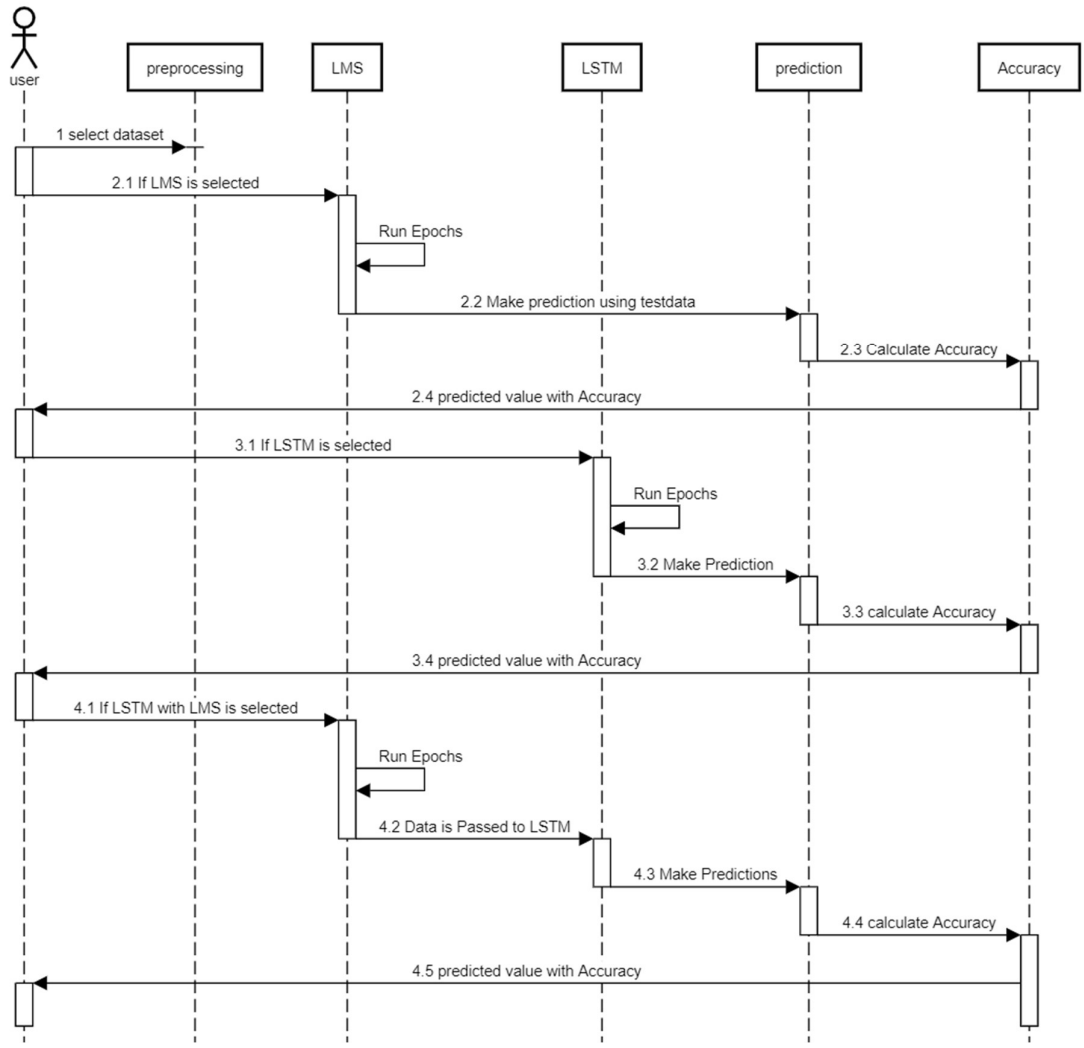


fig 3.2.2 : Execution based on model selection

3.2.3 Activity Diagram

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

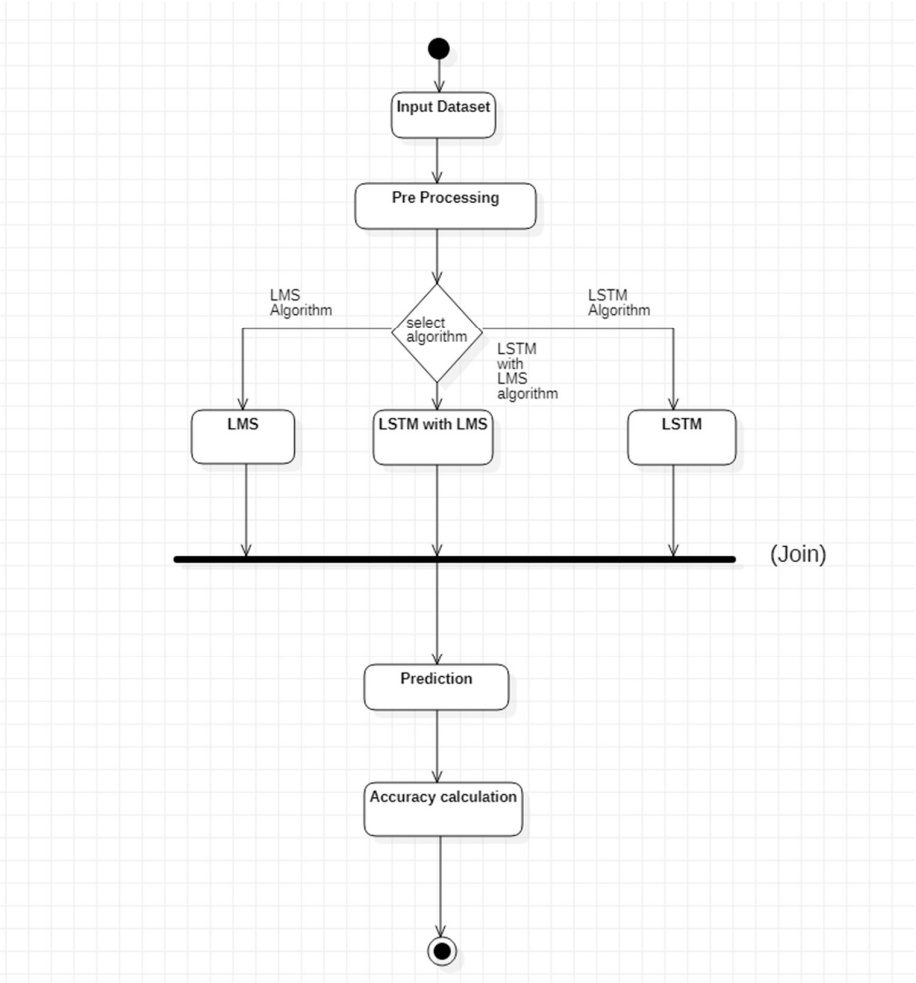


Fig 3.2.3 : Execution based on algorithm selection

4.IMPLEMENTATION

4. IMPLEMENATATION

```
import pandas as pd

df = pd.read_csv('/content/drive/MyDrive/STOCK_DataSet/MSFT.csv')

df

df = df[['Date', 'Close']]

df
df['Date']

import datetime

def str_to_datetime(s):
  split = s.split('-')
  year, month, day = int(split[0]), int(split[1]), int(split[2])
  return datetime.datetime(year=year, month=month, day=day)

datetime_object = str_to_datetime('1986-03-19')
datetime_object

df

df['Date'] = df['Date'].apply(str_to_datetime)
df['Date']
df.index = df.pop('Date')
df
import matplotlib.pyplot as plt
plt.plot(df.index, df['Close'])
import numpy as np

 def df_to_windowed_df(dataframe, first_date_str, last_date_str, n=3):
  first_date = str_to_datetime(first_date_str)
  last_date  = str_to_datetime(last_date_str)
  target_date = first_date
```

```
dates = []
X, Y = [], []

last_time = False
while True:
  df_subset = dataframe.loc[:target_date].tail(n+1)

  if len(df_subset) != n+1:
    print(f'Error: Window of size {n} is too large for date {target_date}')
    return

  values = df_subset['Close'].to_numpy()
  x, y = values[:-1], values[-1]

  dates.append(target_date)
  X.append(x)
  Y.append(y)
  next_week = dataframe.loc[target_date:target_date+datetime.timedelta(days=7)]
  next_datetime_str = str(next_week.head(2).tail(1).index.values[0])
  next_date_str = next_datetime_str.split('T')[0]
  year_month_day = next_date_str.split('-')
  year, month, day = year_month_day
  next_date = datetime.datetime(day=int(day), month=int(month), year=int(year))

  if last_time:
    break
  target_date = next_date
  if target_date == last_date:
    last_time = True
ret_df = pd.DataFrame({})
ret_df['Target Date'] = dates
X = np.array(X)
for i in range(0, n):
  X[:, i]
  ret_df[f'Target-{n-i}'] = X[:, i]
ret_df['Target'] = Y
```

```
  return ret_df

# Start day second time around: '2021-03-25'
windowed_df = df_to_windowed_df(df,
                                '2021-03-25',
                                '2022-03-23',
                                n=3)
windowed_df

def windowed_df_to_date_X_y(windowed_dataframe):
  df_as_np = windowed_dataframe.to_numpy()

  dates = df_as_np[:, 0]

  middle_matrix = df_as_np[:, 1:-1]
  X = middle_matrix.reshape((len(dates), middle_matrix.shape[1], 1))

  Y = df_as_np[:, -1]

  return dates, X.astype(np.float32), Y.astype(np.float32)

dates, X, y = windowed_df_to_date_X_y(windowed_df)
dates.shape, X.shape, y.shape
q_80 = int(len(dates) * .8)
q_90 = int(len(dates) * .9)
dates_train, X_train, y_train = dates[:q_80], X[:q_80], y[:q_80]
dates_val, X_val, y_val = dates[q_80:q_90], X[q_80:q_90], y[q_80:q_90]
dates_test, X_test, y_test = dates[q_90:], X[q_90:], y[q_90:]

plt.plot(dates_train, y_train)
plt.plot(dates_val, y_val)
plt.plot(dates_test, y_test)
plt.legend(['Train', 'Validation', 'Test'])
from tensorflow.keras.models import Sequential
from tensorflow.keras.optimizers import Adam
from tensorflow.keras import layers
```

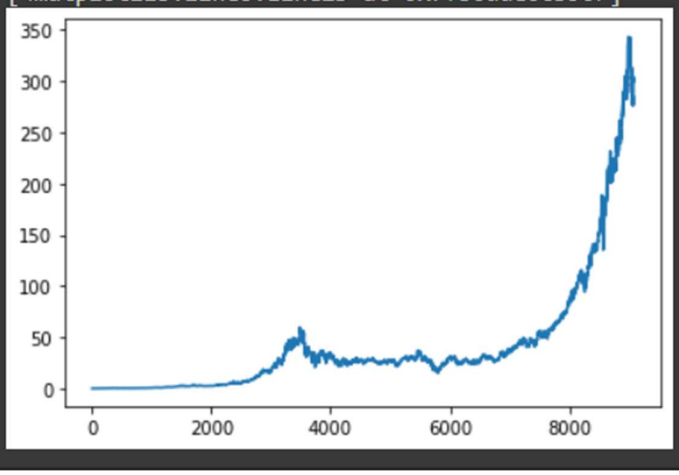
```
model = Sequential([layers.Input((3, 1)),
                    layers.LSTM(64),
                    layers.Dense(32, activation='relu'),
                    layers.Dense(32, activation='relu'),
                    layers.Dense(1)])

model.compile(loss='mse',
              optimizer=Adam(learning_rate=0.001),
              metrics=['mean_absolute_error'])

model.fit(X_train, y_train, validation_data=(X_val, y_val), epochs=100)

train_predictions = model.predict(X_train).flatten()

plt.plot(dates_train, train_predictions)
plt.plot(dates_train, y_train)
plt.legend(['Training Predictions', 'Training Observations'])

val_predictions = model.predict(X_val).flatten()

plt.plot(dates_val, val_predictions)
plt.plot(dates_val, y_val)
plt.legend(['Validation Predictions', 'Validation Observations'])
test_predictions = model.predict(X_test).flatten()
plt.plot(dates_test, test_predictions)
plt.plot(dates_test, y_test)
plt.legend(['Testing Predictions', 'Testing Observations'])

plt.plot(dates_train, train_predictions)
plt.plot(dates_train, y_train)
plt.plot(dates_val, val_predictions)
plt.plot(dates_val, y_val)
plt.plot(dates_test, test_predictions)
plt.plot(dates_test, y_test)
plt.legend(['Training Predictions',
```

```
        'Training Observations',
        'Validation Predictions',
        'Validation Observations',
        'Testing Predictions',
        'Testing Observations'])
from copy import deepcopy
recursive_predictions = []
recursive_dates = np.concatenate([dates_val, dates_test])
for target_date in recursive_dates:
  last_window = deepcopy(X_train[-1])
  next_prediction = model.predict(np.array([last_window])).flatten()
  recursive_predictions.append(next_prediction)
  last_window[-1] = next_prediction
plt.plot(dates_train, train_predictions)
plt.plot(dates_train, y_train)
plt.plot(dates_val, val_predictions)
plt.plot(dates_val, y_val)
plt.plot(dates_test, test_predictions)
plt.plot(dates_test, y_test)
plt.plot(recursive_dates, recursive_predictions)
plt.legend(['Training Predictions',
        'Training Observations',
        'Validation Predictions',
        'Validation Observations',
        'Testing Predictions',
        'Testing Observations'
        'Recursive Predictions']
```

5.SCREENSHOTS

```
import matplotlib.pyplot as plt

plt.plot(df.index, df['Close'])
```

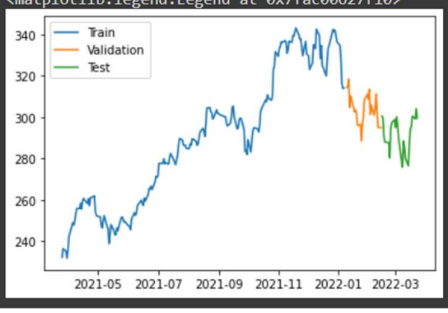
```
[<matplotlib.lines.Line2D at 0x7fb6dde0c390>]
```



Screenshot 5.1: Graph of given data

```
q_80 = int(len(dates) * .8)
q_90 = int(len(dates) * .9)

dates_train, X_train, y_train = dates[:q_80], X[:q_80], y[:q_80]

dates_val, X_val, y_val = dates[q_80:q_90], X[q_80:q_90], y[q_80:q_90]
dates_test, X_test, y_test = dates[q_90:], X[q_90:], y[q_90:]

plt.plot(dates_train, y_train)
plt.plot(dates_val, y_val)
plt.plot(dates_test, y_test)

plt.legend(['Train', 'Validation', 'Test'])
```

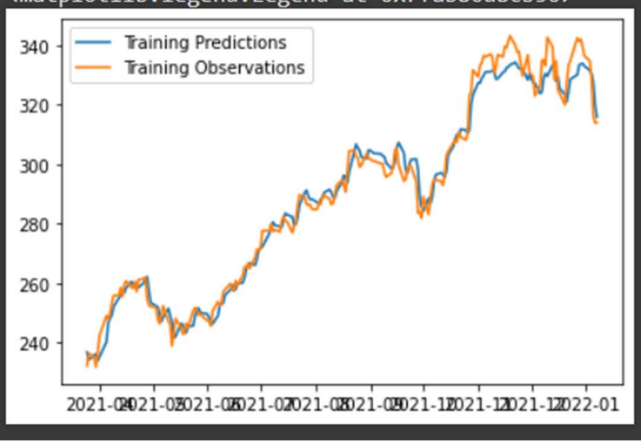
```
<matplotlib.legend.Legend at 0x7fac00627f10>
```



Screenshot 5.2:Training the data

```
train_predictions = model.predict(X_train).flatten()

plt.plot(dates_train, train_predictions)
plt.plot(dates_train, y_train)
plt.legend(['Training Predictions', 'Training Observations'])
```

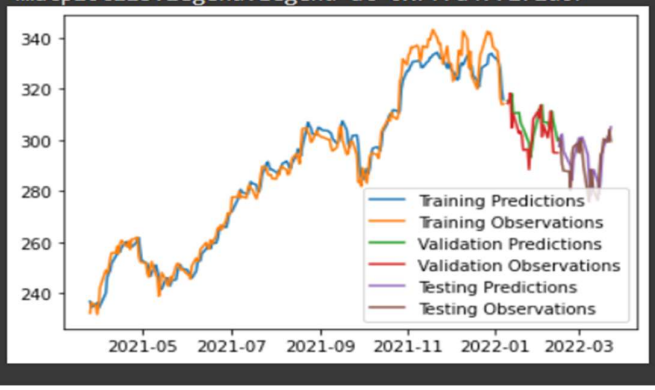
```
<matplotlib.legend.Legend at 0x7fab80d8cb50>
```



Screenshot 5.3: Training prediction

```
plt.plot(dates_val, y_val)
plt.plot(dates_test, test_predictions)
plt.plot(dates_test, y_test)
plt.legend(['Training Predictions',
            'Training Observations',
            'Validation Predictions',
            'Validation Observations',
            'Testing Predictions',
            'Testing Observations'])
```

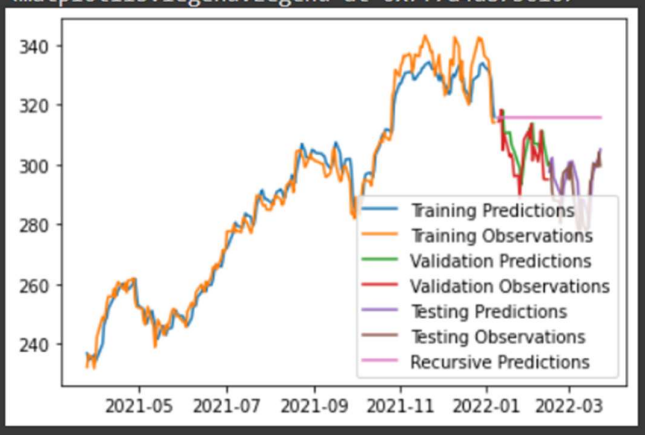
```
<matplotlib.legend.Legend at 0x7f7a477271d0>
```



Screenshot 5.4: Visualization

```
plt.plot(recursive_dates, recursive_predictions)
plt.legend(['Training Predictions',
            'Training Observations',
            'Validation Predictions',
            'Validation Observations',
            'Testing Predictions',
            'Testing Observations',
            'Recursive Predictions'])
```

```
<matplotlib.legend.Legend at 0x7f7a4a87bc10>
```



Screenshot 5.5: Predicted Graph

6.TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. This is a structural testing that relies on knowledge of its construction and is invasive.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate thatalthough the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNXTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, . Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted

7.CONCLUSION

7.CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

- By estimating the parameters of accuracy, it was found that the most appropriate method for stock market prediction is to use multiple algorithms such as RNN, LSM, LSTM and combine their results for predictions, and also considering all the factors affecting the stock market price such as news sentiment, currency, commodity prices and other international stock exchange market data. High accuracy was achieved by doing all this. Considering other factors affecting the stock price improves accuracy of prediction. Trying to predict the stock market only when its more predictable and avoiding when market is more uncertain gives greater profits.

7.2 FUTURE SCOPE

- We want to extend this application for predicting cryptocurrency trading.
- We want to add sentiment analysis for better analysis.

8.BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

[1] Stock Price Prediction Using LSTM on Indian Share Market by Achyut Ghosh, Soumik Bose1, Giridhar Maji, Narayan C. Debnath, Soumya Sen.

[2] S. Selvin, R. Vinayakumar, E. A. Gopalkrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in International Conference on Advances in Computing, Communications and Informatics, 2017.

[3] Murtaza Roondiwala, Harshal Patel, Shraddha Varma, "Predicting Stock Prices Using LSTM" in Undergraduate Engineering Students, Department of InformationTechnology, Mumbai University, 2015.

[4] Xiongwen Pang, Yanqiang Zhou, Pan Wang, Weiwei Lin, "An innovative neural network approach for stock market prediction", 2018

[5] Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan Department of Computer Science and Engineering National Institute of Technology, Hamirpur – 177005, INDIA - Stock Market Prediction Using Machine Learning.

[6] Pranav Bhat Electronics and Telecommunication Department, Maharashtra Institute of Technology, Pune. Savitribai Phule Pune University - A Machine Learning Model for Stock Market Prediction.

[7] Anurag Sinha Department of computer science, Student, Amity University Jharkhand Ranchi, Jharkhand (India), 834001 - Stock Market Prediction Using Machine Learning.

[8] V Kranthi Sai Reddy Student, ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India - Stock Market Prediction Using Machine Learning.

GITHUB LINK: **https://github.com/Navaneethnani/Stockmarket-prediction-using-machine-learning**

9.JOURNAL