A PROJECT REPORT ON

STOCK MARKET PREDICTION ANALYSIS

SUBMITTED TO THE VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY,
PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF TECHNOLOGY (COMPUTER ENGINEERING)

SUBMITTED BY

RISHIKESH SHEDE	**17U208**
PRATIKSHA DHANDE	**17U262**
JEET CHETWANI	**17U578**
SHREYA JAMBHALE	**17U642**

DEPARTMENT OF COMPUTER ENGINEERING

BRACT'S
VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY

SURVEY NO. 3/4, KONDHWA (BUDRUK), PUNE – 411048, MAHARASHTRA (INDIA).

BRACT'S
VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE
2020 -2021

[P-2]

CERTIFICATE

This is to certify that the project report entitles

STOCK MARKET PREDICTION ANALYSIS

Submitted by

RISHIKESH SHEDE	**17U208**
PRATIKSHA DHANDE	**17U262**
JEET CHETWANI	**17U578**
SHREYA JAMBHALE	**17U642**

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Mr. Rajiv Kale** (Industry mentor) & **Prof. Vikas Kolekar** (Faculty mentor) and it is approved for the partial fulfillment of the requirement of VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, for the award of the degree of **Bachelor of Technology** (Computer Engineering).

(12, Sentence case)

(Prof. Vikas Kolekar) **(Dr. S. R. Sakhare)**
Guide, Head,
Department of Computer Engineering Department of Computer Engineering

(Dr. Vivek Deshpande)
Director,
BRACT's Vishwakarma Institute of Information Technology, Pune-48

Place : Pune
Date : 30/05/2021

ACKNOWLEDGEMENT

This project was sponsored by Persistent Solutions. Thanks to Persistent Solutions for providing us details regarding their workflow, daily operations and for explaining how each module that we worked on should work and the extent to which they will be utilized. These helped us significantly in designing and developing the proposed system.

RISHIKESH SHEDE	**17U208**
PRATIKSHA DHANDE	**17U262**
JEET CHETWANI	**17U578**
SHREYA JAMBHALE	**17U642**

ABSTRACT

Broadly, stock market analysis is divided into two parts – Fundamental Analysis and Technical Analysis.

Fundamental Analysis involves analyzing the company's future profitability on the basis of its current business environment and financial performance.

Technical analysis on the other hand, includes reading the charts and using statistical figures to identify trends in the stock market.

We have gone with technical analysis, because with technical analysis you can know when to time your action and better understand demand and supply levels. Another significant advantage of technical analysis is that you can learn about short and long-term trends with ease and make a decision accordingly. For short-term traders most of the things depend on technical analysis rather than fundamental analysis.

[P3]

(Four blank spaces)
TABLE OF CONTENTS (14, bold, uppercase)

CHAPTER	TITLE	PAGE NO.

(12, bold, UPPERCASE)

LIST OF ABBREVIATIONS

1. DFD – Data Flow Diagram
2. JS – JavaScript
3. DB – Database
4. UI – User Interface

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

1.1 OVERVIEW

Our Stock market prediction system aims to provide professional to beginner traders/investors a basic analysis and prediction of the stock the user wants to possibly invest in or just want to enquire about, system will be able to display that particular prediction on a systematic graph and will predict future low and high points too with the help of machine learning and display them on an interactive graphical user interface. Our System will be able to Store this analysis onto a database to reduce future computations related to that particular stock and also utilizing that prior analysis to improve future predictions.

We will also be implementing a separate additional feature for user to inculcate his/her intuition into the prediction and possibly improving the accuracy based on that.

1.2 MOTIVATION

Our Motivation behind this project comes from the following points:
- The opportunity to work on an Industry sponsored Project.
- To understand and learn how projects are developed, executed, and managed in Industries.
- Design, Planning, Developing, and Testing a project according to Industry standards.
- Delivering a self-made, robust software application.

1.3 PROBLEM DEFINITIONS AND OBJECTIVES

Problem Definition: Developing a Stock market prediction system to help users understand trends and technical charts clearly and thereby improve trading efficiency and overall user experience.

Objective: To develop a robust, easy-to-use stock market prediction system wherein users can access the technical charts and trends through easy and simple to understand user interface. User should be able to select a company/index of their choice and see the details and technical charts of the selected stock/index and thereby improve their trading efficiency.

1.4 PROJECT SCOPE AND LIMITATIONS

1.4.1 The scope of the project is as follows:

- Selection/Search the desired company/index.
- If company/ index data not already present then data for same to be downloaded.
- Information/ Data and technical charts to be displayed for the respective company/ index.
- Based on past data and other factors, prediction will be done with a certain level of accuracy.

1.4.2 Limitations:

- A proper internet connection is required for using the system and downloading data of a new stock/index that is not already in the database.
- Cannot possibly predict sudden market crashes like CoVid-19 Market Crash.

1.5 METHODOLOGIES OF PROBLEM SOLVING

Our proposed methodology for each module is given below:

A. **SELECT**: In this module, the user can select the stock/index that he/she wishes to trade out of the multiple stocks/indices. For example, if the user wishes to trade in Google stocks, the user can search for the same in the search bar provided.

B. **INFORMATION DISPLAY:** After the user selects the desired stock/index, they get an option to select from the different type of technical charts available like the line chart, step charts, spline etc. Seeing these the user can notice the various trends, previous ups and downs, predicted direction and various other technicalities that help the user to make the decision of buying/selling.

C. **PREDICTION:** A certain prediction will be made for the stock that has been selected. This prediction will be based on various factors such as past data, current trends, etc. With this prediction and analysis user will be able to make their decision easily without much confusion.

2.   LITERATURE SURVEY

The research methodology used in this study is the learning and study of the working of the stock market. The search was made using keywords such as stock market basics, stock market prediction, stock market index, time-series prediction models, stock market data etc. But, the issues with the given references were not economical & effective.

The information and guidance shared by our industry mentor at all stages of the project about their personal experience and vast knowledge lighted up the survey field due to which we started the implementation of this system using our own idea and logic in it. The most important part of adopting a stock market system is the trust factor and the accuracy of prediction. The result predicted should be in the right direction and as accurate as possible to the final value. The user should have a better understanding of the technicalities which may help the user to make the decision about buying/ selling and they should also have a good overall user experience.

SOFTWARE REQUIREMENTS SPECIFICATIONS

3.1 ASSUMPTIONS AND DEPENDENCIES

Our project was built with an assumption that multiple users can use the system simultaneously. They can also select a company not already in the database for which the data will now have to be downloaded.

The dependencies list the following libraries and tools:

- Django
- Google Colab
- Pandas
- Numpy
- Sklearn
- Matplotlib
- Tkinter

3.4 NON FUNCTIONAL REQUIREMENTS

3.4.1 Performance Requirements

Since the Stock Prediction system is hosted on a cloud server, there are no minimum hardware requirements apart from the hardware resources required for using a Web Browser. Stock Prediction system is compatible with both desktop and mobile platforms.

3.4.2 Safety Requirements

There are different kinds of data stored in the system namely User data, Client data, etc. To ensure their safety we are using several reliable and secure third party software platforms like Firebase and GCP and using Permission based access which will add another layer of safety.

3.4.3 Security Requirements

To ensure a secure system we are using Firebase Authentication system integrated with our Frontend i.e. VueJS and Backend i.e. Django Framework's server side authentication. Session protection is also in place to protect system's URLs and unauthorised access and signing a user out automatically after the session is closed. Different access levels/roles are assigned to the users to restrict access. Reliable hosts are used to protect the system by restricting IP addresses from any external attacks.

3.4.4 Software Quality Attributes

Stock Prediction system provides an easy-to-use system that reduces manual record keeping and document generation processes. It standardizes, streamlines and simplifies the work and dataflow while providing intuitive interface for document generation. It can be used by both experts and typical users. However, users must already have a basic knowledge of the stock market before using it.

Availability: The service is delivered using online cloud servers, hence it is available from anywhere at any time.

Security: The main security concern is for users' accounts hence proper login mechanism should be used. All the APIs are secured by adding permission only for authorized users.

Usability: As the system is easy to handle and navigates in the most expected way with no delays. In that case, the system program reacts accordingly and transverses quickly between its states.

Reliability: Reliability is the probability of failure-free operation of a system over a specified time within a specified environment for a specified purpose. The software should always be reliable.

Maintainability: Software maintainability is defined as the degree to which an application is understood, repaired, or enhanced. Software maintainability is important because it is approximately 75% of the cost related to a project.

Extensibility: It deals with the extensibility of the system, i.e. extending the features and property of the system. The software should always be extendable.

Reusability: Reusability is the use of existing assets in some form within the software product development process. Assets are products and by-products of the software development life cycle and include code, software components, test suites, designs and documentation i.e. it deals with reusing the software component. The software should always have reusability property.

3.5 SYSTEM REQUIREMENTS

3.5.1 Data Storage Requirements

1. .csv file (Dataset)

3.5.2 Software Requirements (Platform Choice)

Platform:

1. Operating System: Windows, Ubuntu (Web Browser)
2. IDE: Google Collab, VS Code
3. Programming Language: Python3, JavaScript
4. Frameworks: Django, Stacked LSTM
5. Data Storage: .csv file

3.6 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED

The Agile-SCRUM has been applied for the Software Development Life Cycle in this project.

The Agile – Scrum Framework
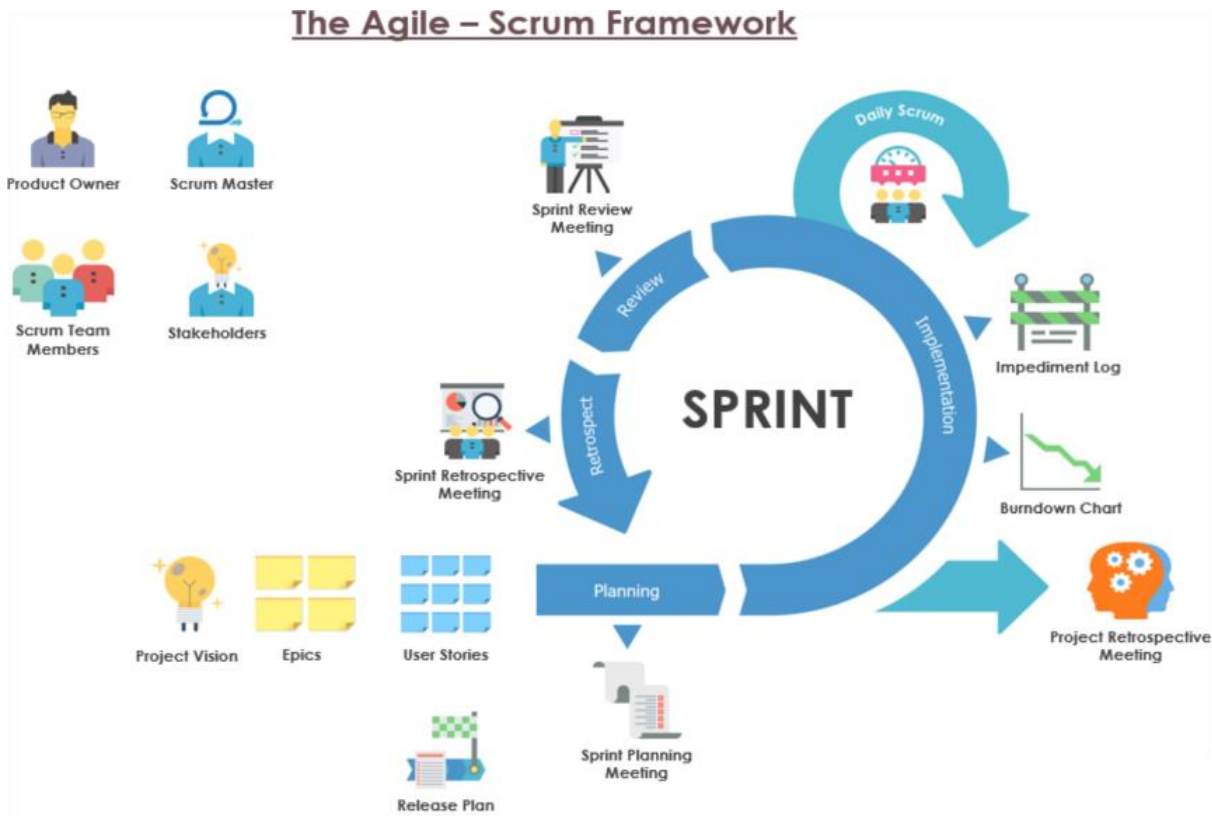
Development Life Cycle in this project

Agile-SCRUM methodology is a practice that promotes continuous iteration of development and testing throughout the software development life cycle of the project.

Every iteration involves cross functional teams working simultaneously on various areas like −

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and
- Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

This system follows MVC software architecture.

MVC is as:

Model: Contains all the data structures and storage related operations.

View: Consists of front end GUI part.

Control: Handles the main execution logic and also manages the communication of View and Model part.

In this project's case the MVC components are as:

Model: Stacked-LSTM

View: VueJS

Control: Django

These components work in the following way:

Stacked-LTSM: A Stacked LSTM architecture can be defined as an LSTM model comprised of multiple LSTM layers. An LSTM layer above provides a sequence output rather than a single value output to the LSTM layer below.

VueJS: This component allows us to create web based pages (GUI Components) for interacting with the user.

Django: Enables us to obtain data from VueJS, perform some operations on it.

API: These are communication mechanisms that allow flow of data from Vue to Django and to Google Colab.

4.2 DATA FLOW DIAGRAM

```
                    ┌─────────┐
                    │  START  │
                    └────┬────┘
                         │
                         ▼
              ┌──────────────────────┐
              │  Download and store  │
              │   the data in CSV    │
              │       format         │
              └──────────┬───────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │   Data Preprocessing │
              └──────────┬───────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │    Model Training    │
              └──────────┬───────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │    Save the model    │
              └──────────┬───────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │  Predict future prices│
              │   using that saved   │
              │        model         │
              └──────────┬───────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │  Plot the graph on UI │
              └──────────┬───────────┘
                         │
                         ▼
                    ┌─────────┐
                    │  STOP   │
                    └─────────┘
```
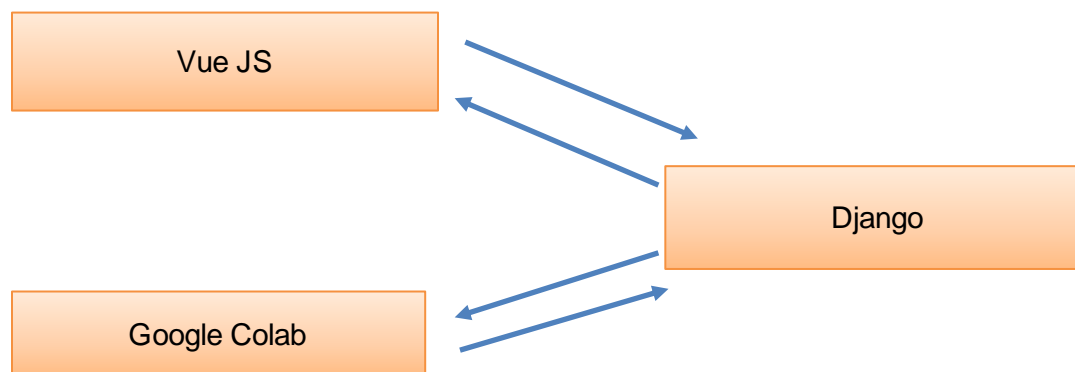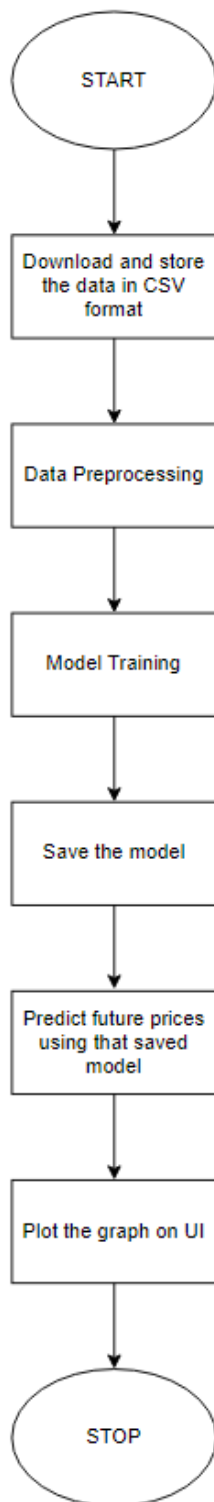
4.3 Stacked Long Short Term Memory (LSTM) Network Based Model

```
Input
  |
  v
LSTM
  |
  v
LSTM
  |
  v
Dense
  |
  v
Output
```
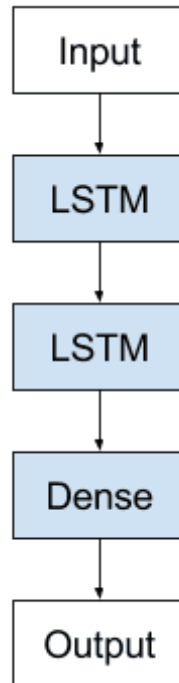
PROJECT PLAN

5.1 RISK MANAGEMENT

5.1.1 Risk Identification

- Any system can be hacked virtually with a lot of expertise, intent and motivation and so can be said for our system developed by students with no industry experience.
- System has been developed and tested in a controlled environment with reasonable understanding of the stock's index workflow for different companies. So in case of unexpected client activities the system might not cope-up their requests or display errors or bugs.
- In case the hosting platform or Database Provider is down the system will not work.
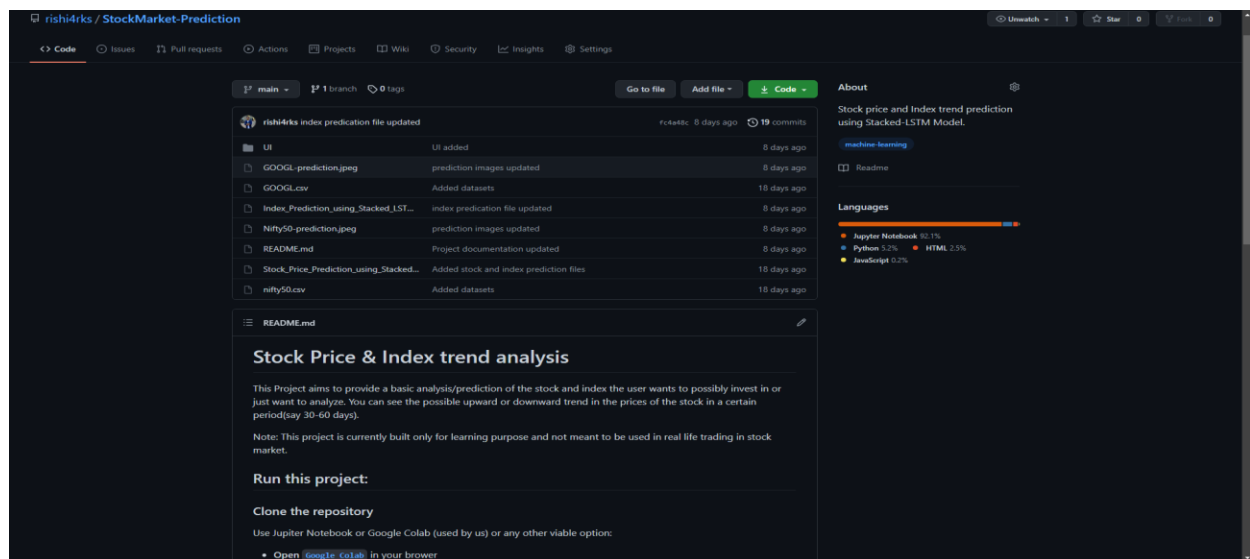
5.1.2 Risk Analysis

Sr. No	Risk Description	Probability	Impact		
			Schedule	Quality	Overall

1	Compromised System Security	Low	Low	High	High
2	Unexpected Requests from Client Side	Medium	Low	High	High
3	Hosting or Database Provider not available	Low	Low	High	High

5.2 PROJECT SCHEDULE

5.2.1 Project Development Repository

5.3 TEAM ORGANISATION

Sr. No.	Name	Role	Work Area
1	Prof. V. K. Kolekar	Mentor	Mentoring
2	Rishikesh Shede	Student	, Testing and Documentation
3	Jeet Chetwani	Student	, Testing and Documentation
4	Pratiksha Dhande	Student	, Testing and Documentation
5	Shreya Jambhale	Student	, Testing and Documentation

PROJECT IMPLEMENTATION

Algorithms:

- The system implements logic for auto increment counters that are stored in the DB. Quotation numbers, Vendor PO numbers, Customer Po numbers and company codes have an auto incremental part at their end. This is done by fetching the stored counter in DB and assigning value counter+1 while updating the same bat in the DB.
- We also have a revision system where one quotation can have multiple revisions or edits. These revisions also have a incremental id stored in their respective storages. The currently active or finalized document is stored in quotations part while revisions are stored in revisions part.
- STM is the advanced version of Recurrent-Neural-Networks (RNN) where the information belonging to previous state persists. These are different from RNNs as they involve long term dependencies and RNNs works on finding the relationship between the recent and the current information. This indicates that the interval of information is relatively smaller than that to LSTM.
- The main purpose behind using this model in stock market prediction is that the predictions depend on large amounts of data and are generally dependent on the long -term history of the market.
- So, LSTM regulates error by giving an aid to the RNNs through retaining information for older stages making the prediction more accurate.
- The original LSTM model is comprised of a single hidden LSTM layer followed by a standard feedforward output layer.
- The Stacked LSTM is an extension to this model that has multiple hidden LSTM layers where each layer contains multiple memory cells.

6.1 TOOLS AND TECHNOLOGIES USED

The Stock Market Prediction system is mainly based on web technologies such as Vue JS and Django Framework that use JavaScript and Python languages respectively. Stacked-LSTM Algorithm is used as the basis of this project and .csv file is used for the database. Finally, the system is deployed on web browser.
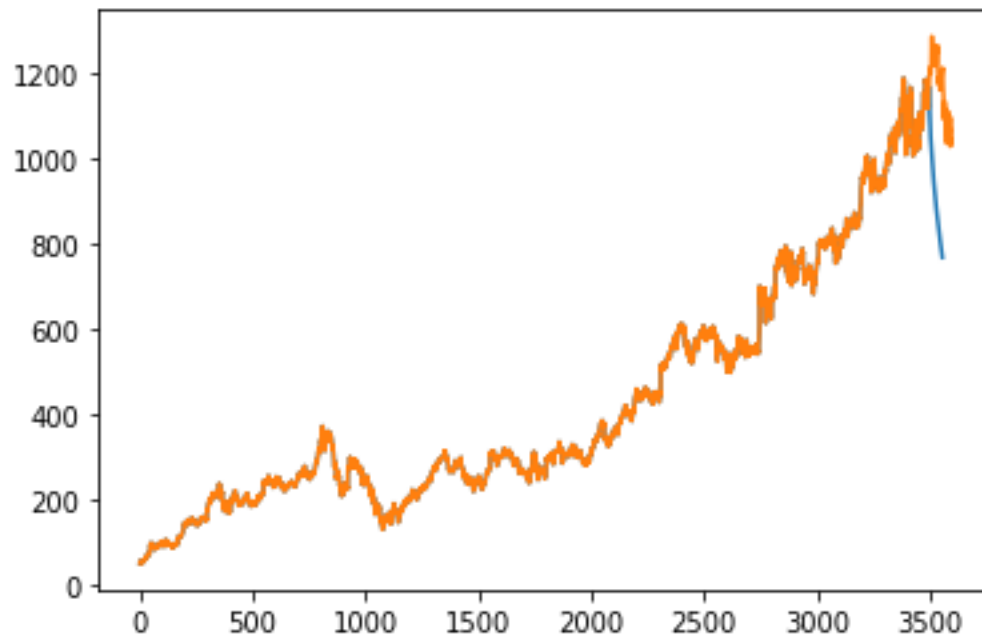
For development purposes we used Google Collab and VS Code IDEs. For our code base and version control management we used Gitlab and its Boards for Agile-SCRUM framework. We have also used Postman to test and verify our APIs.
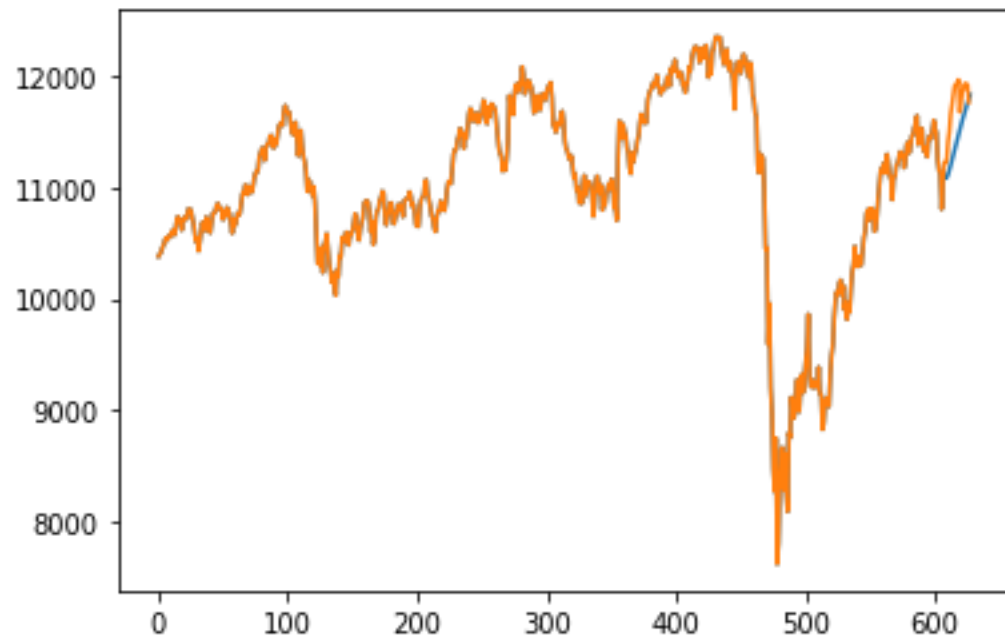
RESULTS

8.1 OUTCOMES

The basic outcomes we get are from the Graphs which can be plotted in two ways as per the user's choice.

8.2 SCREENSHOTS

CHALLENGES WE FACED

1. **Linear Regression Model**
   A. **Introduction**
      1. It is used when we want to predict the value of a variable based on the value of another variable. The variable we want to predict is called the dependent variable.
      2. The simplest form of the regression equation with one dependent and one independent variable is defined by the formula

         $$y = m*x + c$$

         where y is the estimated dependent variable, m is the regression coefficient, or what is commonly called the slope, x is the independent variable and c is a constant. In simple words, y is the output when m, x, and c are used as inputs.

   B. **Usefulness**
      a. It can tell you on average, what has been the past trend, and like mentioned at the start, if the future trend happens to be the same as the past trend then your prediction will be quite accurate using the simple linear regression model which isn't a lot of work
      b. Using the percent error, a range could be deduced and it can be expected the future value could lie within this range.
      c. Once can work out all the cases and become aware of the future values if the trend improves, worsens, or remains the same and so have an idea of the possible ranges.

   C. **Limitations**
      a. The future predicted value is shown as a linear increase which is not true as if you observe the actual values you see a jump up and a fall down in the prices. There is a lot of variability in the actual data but a standard increase in the forecasted values.
      b. The line goes only one way which means if the past has been of an average increase over a period of time then the future will also show the average increase over the next period. In reality, there is always a danger of trend reversal which the linear forecast can never reveal.
      c. We simply assumed that it will be equal to the past which is the standard assumption one makes when using the simple linear regression model.
      d. The values by and of itself differ from the actual by large percentages in excess of 10% which shows it is not such an accurate predictor.

   2. **ARIMA Model**
   A. **Introduction**

a. **ARIMA** is an acronym for "autoregressive integrated moving average." It's a **model used** in statistics and econometrics to measure events that happen over a period of time.

b. The **model** is **used** to understand past data or predict future data in a series.

c. It's used when a metric is recorded in regular intervals, from fractions of a second to daily, weekly or monthly periods.

B. Usefulness

a. ARIMA models are parametric in the sense that they are fitted to and used to predict an individual time series.

b. The future predicted value is not a linear increase like it was using Linear regression. The predicted value can be compared with the actual values.

c. But there is a jump up and a fall down in the prices. There is a lot of variability in the actual data but not in the forecasted values.

C. Limitations

a. The predicted index and the real index have a lot of difference.

b. This model renders front end application for each prediction which is quite time consuming.

3. Stacked LSTM

A. Introduction

a. Multiple hidden LSTM layers can be stacked one on top of another in what is referred to as a Stacked LSTM model.

b. An LSTM layer requires a three-dimensional input and LSTMs by default will produce a two-dimensional output as an interpretation from the end of the sequence.

c. Long Short-Term Memory (LSTM) networks are often used to capture temporal dependency patterns. By stacking multi-layer LSTM networks, it can capture even more complex patterns.

B. Usefulness

a. The main purpose behind using this model in stock market prediction is that the predictions depends on large amounts of data and are generally dependent on the long term history of the marke.

b. So LSTM regulates error by giving an aid to the RNNs through retaining information for older stages making the prediction more accurate.

c. Since stock market involves processing of huge data, the gradients with respect to the weight matrix may become very small and may degrade the learning rate of the system.

d. This corresponds to the problem of Vanishing Gradient. LSTM prevents this from happening.

e. The LSTM consists of a remembering cell, input gate, output gate and a forget gate. The cell remembers the value for long term propagation and the gates regulate them.

CONCLUSIONS

9.1 CONCLUSION

Stock Market Prediction system has been developed keeping in mind the user requirements and security concerns of the organizations. The system has a user-friendly UI built using Vue.js, integrated with Django server using secure REST APIs. Users can conveniently see index value of desired shares. The system provides permission-based access to the features of the portal to ensure data security and integrity. Stock Market Prediction System will thus be a hassle-free mean to manage resources. Stacked-LSTM Technique have been utilized in this paper. The technique has shown an improvement in the accuracy of predictions, thereby yielding positive results. It has led to the conclusion that it is possible to predict stock market with more accuracy and efficiency using machine learning techniques.

9.2 FUTURE WORKS

In the future, the stock market prediction system can be further improved by utilizing a much bigger dataset than the one being utilized currently. This would help to increase the accuracy of our prediction models. Furthermore, other models of Machine Learning could also be studied to check for the accuracy rate resulted by them.

9.3 APPLICATIONS

The applications of the project are:

1. Our application is the act of trying to determine the future price of a company stock.
2. The successful prediction of a stock's future price could yield significant profit.
3. It basically works with the past values of the company and gives the approximate range of the stock price.

REFERENCES

1. https://vuejs.org/
2. https://docs.djangoproject.com/en/3.2/
3. https://www.tensorflow.org/