STOCK PRICE PREDICTION PROGRAM

A PROJECT REPORT

Submitted by

HARDEY PANDYA (166490316073)

SARVIK VAGHASIYA (166490316120)

In fulfilment for the award of the degree

of

DIPLOMA ENGINEERING

in

INFORMATION TECHNOLOGY

Guided by

HARDIK P. JAGAD

Sir Bhavsinhji Polytechnic Institute Bhavnagar

Gujarat Technological University, Ahmedabad

April, 2019

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD

SIR BHAVSINHJI POLYTECHNIC INSTITUTE

BHAVNAGAR

Certificate

This is to certify that Mr. Pandya Hardey Nileshbhai from **Sir Bhavsinhji Polytechnic Institute, Bhavnagar** College having Enrollment No: 166490316073 has completed **Project Report** having title Stock Price Prediction Program in a group consisting of **2** persons under the guidance of the faculty guide Mr. Hardik Jagad.

Institute Guide-UDP

Head of Department

SIR BHAVSINHJI POLYTECHNIC INSTITUTE

BHAVNAGAR

Certificate

This is to certify that Mr./Ms. Vaghasiya Sarvik Ashokbhai from **Sir Bhavsinhji Polytechnic Institute, Bhavnagar** College having Enrollment No: 166490316120 has completed **Project Report** having title Stock Price Prediction Program, individually/ in a group consisting of **2** persons under the guidance of the faculty guide Mr. Hardik Jagad

Institute Guide-UDP   Head of Department

STUDENT INFORMATION SHEET

Name of Student	Pandya	Hardey	Nileshbhai
	Surname	**Name**	**Father’s Name**
Enrollment Number	166490316073		
Contact Numbers	**Mob:** 9265595781	**Land Line: -**	
Email ID	pandyahardey@gmail.com		
College Name	Sir Bhavsinhji Polytechnic Institute, Bhavnagar	**College Code:** 649	
Branch	Information Technology	**Semester :** V	
Student Team	**Name**	**Enrollment Number**	
	Pandya Hardey Nileshbhai	166490316073	
	Vaghasiya Sarvik Ashokbhai	166490316120	
Student Signature			

STUDENT INFORMATION SHEET

Name of Student	Vaghasiya	Sarvik	Ashokbhai
	Surname	**Name**	**Father's Name**
Enrollment Number	166490316120		
Contact Numbers	**Mob:** 7622089518	**Land Line: -**	
Email ID	sarvikvaghasiya611@gmail.com		
College Name	Sir Bhavsinhji Polytechnic Institute, Bhavnagar	**College Code:** 649	
Branch	Information Technology	**Semester :** V	
Student Team	**Name**	**Enrollment Number**	
	Pandya Hardey Nileshbhai	166490316073	
	Vaghasiya Sarvik Ashokbhai	166490316120	
Student Signature			

Acknowledgement

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced our thinking, behaviour, and acts during the course of study.

We express our sincere gratitude to prof. **D.A. Dave,** worthy Principal of the college for providing us an opportunity to implement and test our project at the college.

We are thankful to **Mr. Hardik Jagad, our Project Guide,** for his support, cooperation, and motivation provided to us during the design and analysis of this.

We also extend our sincere appreciation to **Mr. G.M. Pandey, Head of Department – Information Technology** who provided his valuable suggestions and precious time in accomplishing our project report.

Lastly, We would like to thank our friends and collegues with whom we shared our day-to-day experience and received lots of suggestions that improved our quality of work.

Sincerely,

Hardey Pandya.
Sarvik Vaghasiya.

Index

Abstract

Stock Market Prediction is the act of trying to determine the future value of a company stock. The successful prediction of stock's future price will maximize the investor's gains. Investors regularly check the prices, analyse them and according to that they invest in particular share of a company. But it is full of uncertainty. There are no specific rules for estimating stock prices and therefore a lot of approaches has been used with their own pros and cons. We consider the traditional historical time series analysis, Technical Analysis and Fundamental Analysis.

We analysed some research papers and performed some experiments with Data Mining tool WEKA and found that Machine learning model which use Artificial Neural Networks (ANN) and Support Vector Regression (SVR) are most efficient. In the domain of Deep Learning, most popular and widely used are Long-Short Term Memory (LSTM).

Financial News, Business Articles are also taken into consideration to perform Textual Analysis. Here we take data of 100 companies registered in NSE India, which are always under the eye of traders. The project is aimed to aid regular shareholders to invest in right area. We consider the traditional historical time series analysis, Technical Analysis and Sentimental Analysis.

Keywords: Machine Learning, Technical Analysis, Time Series Analysis, Artificial Neural Networks, Support Vector Machine Textual Analysis, LSTM, SVR.

-: List of Figures :-

-: List of Tables :-

-: List of Abbreviations :-

1. ANN – Artificial Neural Network
2. RNN – Recurrent Neural Network
3. EMH – Efficient Market Hypothesis
4. SVM – Support Vector Machine
5. NSE – National Stock Exchange (India)
6. BSE – Bombay Stock Exchange
7. CNN – Convolutional Neural Network
8. DFD – Data Flow Diagram
9. ERD – Entity-Relationship Diagram
10. RSI – Relative Strength Index
11. CCI – Commodity Channel Index
12. SMA – Simple Moving Average
13. MD – Mean Deviation
14. LSTM – Long Short Term Memory

1

Chapter # 1: Introduction

1.1 Project Introduction

1.2 Purpose

1.3 Scope

1. Introduction

1.1 Project Introduction:

Stock Market prediction and analysis 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 country and plays a vital role in growth of the industry and commerce of the country. Both investors and industry are involved in stock market and wants to know whether some stock will rise or fall over certain period of time.

The main motivation behind choosing the project:

- There is large amount of relevant financial data available on the internet which is increasing day by day.
- Large number of Computer Science disciplines including software engineering, databases, distributed systems and machine learning have increased possibility to apply skills.
- The opportunity to expand our knowledge in finance and investing, as we had only little prior exposure to these fields.
- It possesses many theoretical and experimental challenges.

Stock market is very difficult to understand. It is considered too much uncertain to predict due to huge fluctuations in the market. Stock market prediction task is interesting and it divides researchers into two schools: one who believes market can be modelled into some algorithms and subsequently can be predicted and those who believe in EMH.

Most of the trading in the Indian stock market takes place on its two stock exchanges: the BSE and the NSE. The BSE has been in existence since 1875. The NSE, on the other hand, was founded in 1992 and started trading in 1994. However, both exchanges follow the same trading mechanism, trading hours, settlement process, etc. At the last count, the BSE had about 4,700 listed firms, whereas the rival NSE had about 1,200. [1]

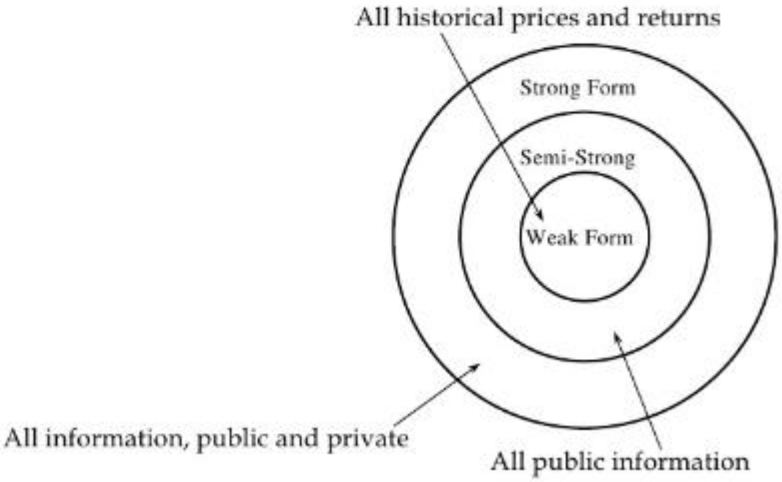
Here is the basic terminology related to prediction of stock market:

Efficient Market Hypothesis (EMH) :-

Weak-form Efficient Market Hypothesis: - The weak form of the hypothesis says that no one can profit from the stock market by looking at trends and patterns within the price of a product itself. It is important to note that this does not rule out profiting from predictions of the price of a product based on data external to the price. We will see examples of prediction based on both in sample and out of sample data, and provide evidence in support of the weak form.

Semi-Strong Efficient Market Hypothesis: - The semi-strong form rules out all methods of prediction, except for insider trading. This means that if we are only to use public domain information in our prediction attempt, the semi-strong form says that we will be unsuccessful.

Strong form Efficient Market Hypothesis: - The strong form says that no one can profit from predicting the market, not even insider traders.


All historical prices and returns
Strong Form
Semi-Strong
Weak Form
All information, public and private
All public information

(Efficient Market Hypothesis)

In order to clarify the goal of the project, following are the dominant schools of thought on investing must first be introduced.

Time Series analysis: A time series is a sequence of numerical data points in successive order. In investing, a time series tracks the movement of the chosen data points, such as a security’s price, over a specified period of time with data points recorded at regular intervals. There is no minimum or maximum amount of time that must be included, allowing the data to be gathered in a way that provides the information being sought by the investor or analyst examining the activity.

Fundamental analysis: - This approach is to analyze fundamental attributes in order to identify promising companies. This includes characteristics such as financial results, company’s assets, liabilities, and stock and growth forecasts. It’s very important to understand that this type of analysis is not static; newly released financial information, corporate announcements and other news can influence the fundamental outlook of a company. Fundamental analysis requires expertise in a particular sector and is often conducted by professional analysts. Their recommended investments are regularly published and updated.

Technical analysis :- In contrast to fundamental analysis, technical analysis does not try to gain deep insight into a company's business. It assumes the available public information does not offer a competitive trading advantage. Instead, it focuses on studying a company's historical share price and on identifying patterns in the chart. The intention is to recognize trends in advance and to capitalize on them.

Within the technical analysis community there exist several schools with different techniques, but they all have in common that they use price and volume history. A basic thought is that it takes time before the market reacts upon new information and that pattern often occurs in price behavior which makes forecasting possible.

There are several factors that explain why technical analysis works:

1. Most speculators on the market act upon fundamental analysis, so that kind of facts influence stock prices strongly. But all operators do not get this information at the same time. When there is positive news of a company, those acting immediately can buy shares for a lower price than those getting the news later.
2. Large investors such as mutual funds and banks are often not placing their whole block orders at the same time when they are buying larger quantities of securities because this

would risk triggering an unnecessary high price advance. Instead, the orders are spread over a period that can last several weeks. The resulting increased purchase pressure may result in a steady advancing trend under the period the purchases continue.

3. It is more psychological stressing to go against the trend than to follow it. People are herding animals and like to do as others are doing. This is why a rising stock price is a signal in itself that the price will advance even more. Of course, one has to be careful with stocks that have been rocketing, because they will often recoil.

Comparative Study of Prediction Techniques :-

Criteria	**Technical Analysis**	**Fundamental Analysis**	**Traditional Time Series Analysis**
Data Used	Price, volume, highest, lowest prices.	Growth, dividend payment, sales level, interest rates, tax rates etc.	Historical data
Learning methods	Extraction of trading rules from charts	Simple trading rules extraction	Regression analysis on attributes is used
Type of Tools	Charts are used	Trading rules	RNN, ANN, Linear Regression, etc.
Implementation	Daily basis prediction	Long –term basis prediction	Long –term basis prediction

(Comparative Study of Prediction Techniques)

1.2 Purpose

The purpose of this project are as follows:

- To identify factors affecting share market.
- To generate the pattern from large datasets of NSE stock market for prediction.
- To predict the future value of share price.
- Perform traditional methods of Technical Analysis and Time Series Analysis.
- Predict and inform the user about the status of the company as accurately as possible through Textual Analysis.

Company Selection:

This program will predict the stocks of following companies registered in NSE. This companies are more or less popular among stock market dealer’s community.

- ABB India Ltd.
- ACC Ltd.
- Adani Ports and Special Economic Zone Ltd.
- Aditya Birla Capital Ltd.
- Ambuja Cements Ltd.
- Ashok Leyland Ltd.
- Asian Paints Ltd.
- Aurobindo Pharma Ltd.
- Avenue Supermarts Ltd.
- Axis Bank Ltd.
- Bajaj Auto Ltd.
- Bajaj Finance Ltd.
- Bajaj Finserv Ltd.
- Bandhan Bank Ltd.
- Bank of Baroda
- Bharat Electronics Ltd.
- Bharat Heavy Electricals Ltd.

- Bharat Petroleum Corporation Ltd.
- Bharti Airtel Ltd.
- Bharti Infratel Ltd.
- Biocon Ltd.
- Bosch Ltd.
- Britannia Industries Ltd.
- Cadila Healthcare Ltd.
- Cipla Ltd.
- Coal India Ltd.
- Colgate Palmolive (India) Ltd.
- Container Corporation of India Ltd.
- DLF Ltd.
- Dabur India Ltd.
- Dr. Reddy's Laboratories Ltd.
- Eicher Motors Ltd.
- GAIL (India) Ltd.
- General Insurance Corporation of India
- Godrej Consumer Products Ltd.
- Grasim Industries Ltd.
- HCL Technologies Ltd.
- HDFC Bank Ltd.
- HDFC Life Insurance Company Ltd.
- Havells India Ltd.
- Hero MotoCorp Ltd.
- Hindalco Industries Ltd.
- Hindustan Petroleum Corporation Ltd.
- Hindustan Unilever Ltd.
- Hindustan Zinc Ltd.
- Housing Development Finance Corporation Ltd.
- I T C Ltd.

- ICICI Bank Ltd.
- ICICI Lombard General Insurance Company Ltd.
- ICICI Prudential Life Insurance Company Ltd.
- Indiabulls Housing Finance Ltd.
- Indian Oil Corporation Ltd.
- IndusInd Bank Ltd.
- Infosys Ltd.
- InterGlobe Aviation Ltd.
- JSW Steel Ltd.
- Kotak Mahindra Bank Ltd.
- L&T Finance Holdings Ltd.
- LIC Housing Finance Ltd.
- Larsen & Toubro Ltd.
- Lupin Ltd.
- MRF Ltd.
- Mahindra & Mahindra Ltd.
- Marico Ltd.
- Maruti Suzuki India Ltd.
- Motherson Sumi Systems Ltd.
- NHPC Ltd.
- NMDC Ltd.
- NTPC Ltd.
- Oil & Natural Gas Corporation Ltd.
- Oil India Ltd.
- Oracle Financial Services Software Ltd.
- Petronet LNG Ltd.
- Pidilite Industries Ltd.
- Piramal Enterprises Ltd.
- Power Grid Corporation of India Ltd.
- Procter & Gamble Hygiene & Health Care Ltd.

- Reliance Industries Ltd.
- SBI Life Insurance Company Ltd.
- Shree Cement Ltd.
- Shriram Transport Finance Co. Ltd.
- Siemens Ltd.
- State Bank of India
- Steel Authority of India Ltd.
- Sun Pharmaceutical Industries Ltd.
- Sun TV Network Ltd.
- Tata Consultancy Services Ltd.
- Tata Motors Ltd DVR
- Tata Motors Ltd.
- Tata Steel Ltd.
- Tech Mahindra Ltd.
- The New India Assurance Company Ltd.
- Titan Company Ltd.
- UPL Ltd.
- UltraTech Cement Ltd.
- United Spirits Ltd.
- Vedanta Ltd.
- Vodafone Idea Ltd.
- Wipro Ltd.
- Yes Bank Ltd.
- Zee Entertainment Enterprises Ltd.

1.3 Scope

The project will be useful for investors to invest in stock market based on the various factors. The project target is to create a program that analyses important parameters which affect share prices and implement these parameters in Machine Learning Algorithm to determine the value that particular share will have in near future as accurately as possible. These predicted and analysed data can be observed by anyone and can know the financial status of companies and their comparisons.

Predicted and analysed data is also useful to Companies themselves. Company and industry can use it to stretch their limitations and enhance their stock value. It can be very useful to even researchers, stock brokers, market makers, government and general people.

The main feature of this project is to generate an approximate forecasting output and create a general idea of future values based on the previous data by generating a pattern. The scope of this project does not exceed than a generalized suggestion tool.

The objective of the system is to give an approximate idea of where the stock market might be headed. It does not give a long-term forecasting of a stock value. There are way too many reasons to acknowledge for the long-term output of a current stock.

2

Chapter # 2: System Requirement Analysis

2.1 Current System Study
2.2 Weakness of Current System
2.3 Problem Identification / Definition
2.4 Requirement of New System
2.5 Feasibility Study
2.5.1 Technical
2.5.2 Economical
2.5.3 Operational
2.5.4 Schedule (Time Line Chart)
2.6 Development model (Software Process Model)
2.7 Requirement Validation
2.8 Tools and Technology / Minimum Hardware and Software Requirements
2.9 System Architecture
2.10 Data Flow Diagram
2.11 Use –case Diagram
2.12 Activity Diagram

2. System Requirement Analysis

2.1 Current System Study:

There are various kinds of systems available for stock market forecasting. But each system differs in their approaches.

The approaches used to forecast future directions of share market prices are historically splitted into two main categories: those that rely on technical analysis, and those that rely on fundamental analysis.

The machine learning approach to the latter problem has been declined in several forms, especially during recent years. As an example, good results have been obtained using linear classifiers as the logistic regression one, which has been used to predict the Indian Stock market. More complicated techniques such as Support Vector Machine (SVM), was the best choice for prediction before the rise of neural networks.

Many other systems currently use different set of algorithms including Hidden Markov Models, Bayesian Networks, Multi-Layered Perceptrons (MLP), etc. But in general, most of the time efficient prediction is observed to be done by the use of SVM and several ANNs, rather than other models. It is Important to note the fact that for every company, it may be possible that any different model may be more suitable. It’s not necessary that one model used for one company is suitable for other company. (Check References: In every research paper cited here, one may easily observe this issue)

Currently the systems use RNN, CNN, SVM to predict stock prices, with the advent in computational power of the computers. Computers with mediocre computational powers cannot perform heavy algorithms which involve complex neural networks.

The main disadvantage of all current systems is that once you define the model of prediction, you cannot change it. It may be result into heavy loss in the accuracy of the system. So, every current system tends not to follow EMH.

2.2 Weakness of Current System:

Current systems for stock price prediction are generally proprietary software and generally it costs very high to its consumers. At the same time there are very handful amount of systems available which are cost effective and efficient to perform stock market analysis. Moreover, many current systems do not involve textual analysis. Sometimes stock market is more about the emotions of business rather than just doing analysis of historical data. In such scenarios, textual analysis is helpful.

We also introduce some indices which give information of the weight of time series analysis, fundamental analysis and technical analysis on the final future value of the share.

2.3 Project Definition / Problem Identification :

Forecasting of stock market is gaining more attention as the profitability of investors in the stock market mainly depends on the predictability. If the direction of the market is successfully predicted the investors can yield enough profits out of market using prediction.

Stock price prediction is rather a hazardous operation. A good analyst is therefore not the one who is always right, but someone who is at average, someone who has higher efficiency than his colleagues.

Sometimes overall impression of the company is more helpful in predicting its stock price, therefore textual analysis of relevant newspaper articles, financial news and impressions have also been into consideration.

In the last few years, it has become clear that ANN have become part of this class of analysts. ANNs are programs that are based on the geometry of the human brain. We analysed some research papers and performed some experiments with Data Mining tool WEKA and found that Machine learning model which use Combination of Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are most efficient and they are also fast and it can also be able to run in mediocre hardware specifications.

Investing in a good stock but at a bad time can have disastrous result, while investing in a stock at the right time can bear profits. Financial investors of today are facing this problem of trading as they do not properly understand as to which stocks to buy or which stocks to sell in order to get optimal result. So, the purposed project will reduce the problem with suitable accuracy faced in such real time scenarios.

2.4 Requirement of New System:

It is difficult to find existing systems that are both performance efficient and have high accuracy. Existing systems are also costly. By proposing new system which is open source and give sufficient amount of accuracy to its users will likely to be helpful to take them their decisions.

Day by day, on the internet, availability of financial data, use of social networking, Online opinions and comments, E-Newspaper, Financial articles are increasing at a very high rate. Any new system can make full use of it. By using sentimental analysis and textual analysis it is possible to give probable overall status of the company and sometimes stock market ups and downs are more about status of company rather than just mining the historical data.

2.5 Feasibility Study:

2.5.1 Technical Feasibility:

Simply put, stock market cannot be accurately predicted. The future, like any complex problem, has far too many variables to be predicted. When there are more buyers than sellers, the price increases. When there are more sellers than buyers, the price decreases. It has more to do with emotion than logic. Because emotion is unpredictable, stock market movements will be unpredictable. It’s futile to try to predict where markets are going. They are designed to be unpredictable. To deal with such case we have introduced textual analysis which involve analysis of business articles, news, etc. to give the user certain idea of the status of the company.

2.5.2 Economic Feasibility:

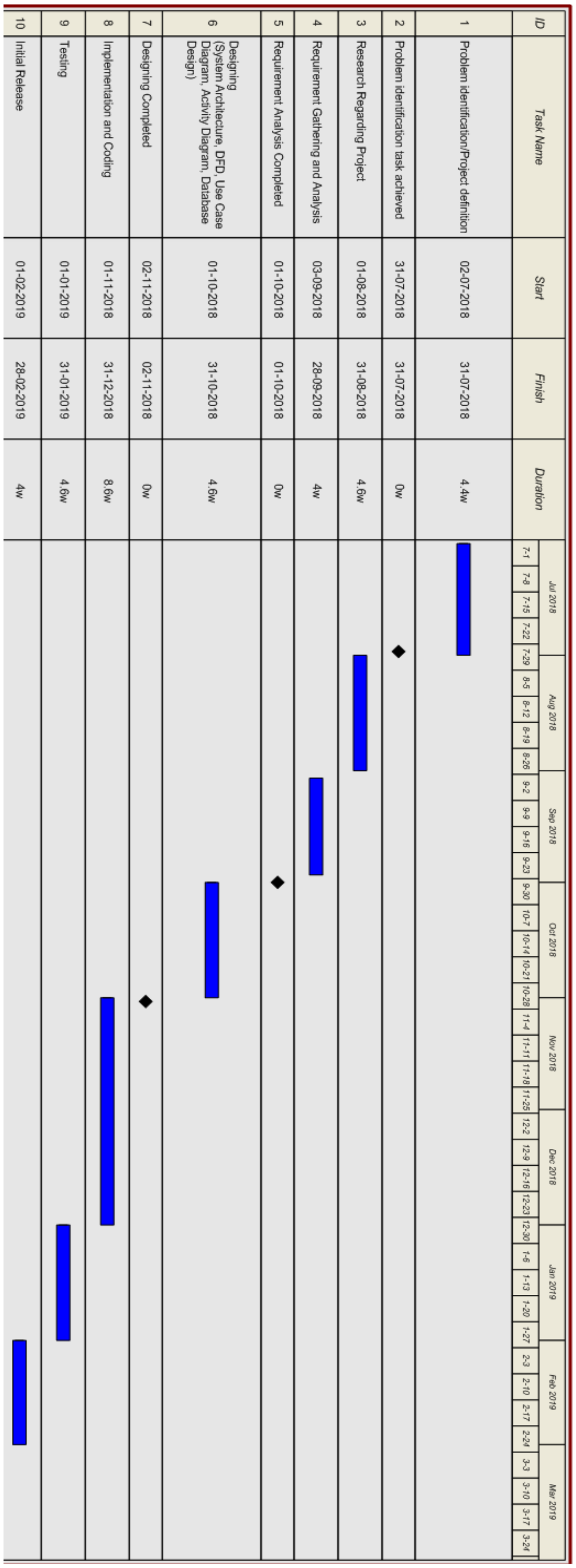
The proposed system is extremely economic feasible. As mentioned above that day by day, availability of financial data is increasing on the internet. Internet is also available with negligible cost today. The system uses internet to fetch the data and process according to certain algorithms which can run on normal hardware specifications. Hardware requirement is also not very fancy.

2.5.3 Operational Feasibility:

The proposed system will not always produce accurate results since it does not account for human behaviours. Factors like change in company’s leadership, internal matters, strikes, protests, natural disasters, terrorist attack, change in authority, political affairs, etc. cannot be considered for relating it to change in Stock market by machine at any circumstances.

The objective of the system is to give an approximate idea of where the stock market might be headed. It does not give a long-term forecasting of a stock value. There are way too many reasons to acknowledge for the long-term output of a current stock. Many things and parameters may affect it on the way due to which long term forecasting is just not feasible.

2.5.4 Time-Line Chart:



ID	Task Name	Start	Finish	Duration
1	Problem identification/Project definition	02-07-2018	31-07-2018	4.4w
2	Problem identification task achieved	31-07-2018	31-07-2018	0w
3	Research Regarding Project	01-08-2018	31-08-2018	4.6w
4	Requirement Gathering and Analysis	03-09-2018	28-09-2018	4w
5	Requirement Analysis Completed	01-10-2018	01-10-2018	0w
6	Designing (System Architecture, DFD, Use Case Diagram, Activity Diagram, Database Design)	01-10-2018	31-10-2018	4.6w
7	Designing Completed	02-11-2018	02-11-2018	0w
8	Implementation and Coding	01-11-2018	31-12-2018	8.6w
9	Testing	01-01-2019	31-01-2019	4.6w
10	Initial Release	01-02-2019	28-02-2019	4w

(Time line chart)

2.6 Software Development model (Software Process Model):

This project will follow the incremental model of software development. The project is decided to follow incremental model due to following reasons:

- Less cost and time will be required to develop the core product.
- This is a smaller scale system.
- Testing each increment is likely to be easier than testing the entire system.
- The feedback providing at each iteration is useful for determining the final requirement of system.

The iteration in the system will be based on the accuracy of the prediction which the system generates. Proper choice of number of hidden layer inputs in ANN can be very important factor in generating the output with higher accuracy. Plus, slight changes in the algorithms will also play an important role with each iteration of the system.

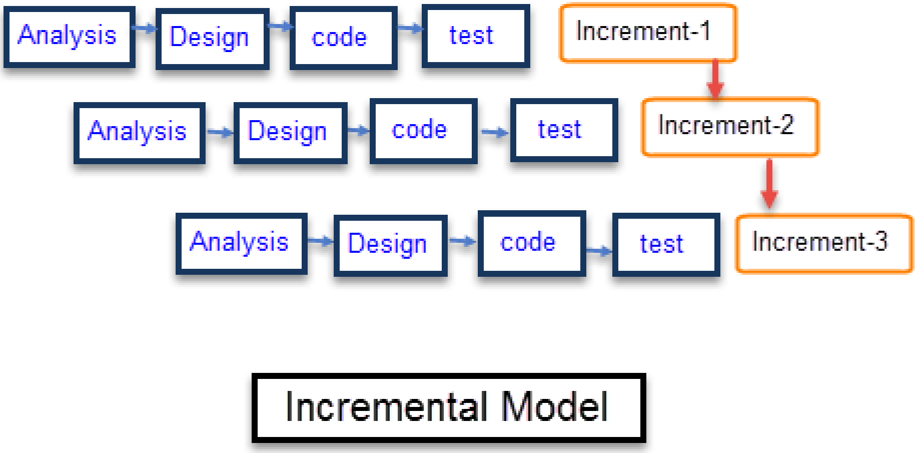
The incremental model combines elements of linear sequential model with the iterative philosophy of prototyping. Each linear sequence produces a deliverable “increment” of the software. We call first increment as the core product. In core product, basic requirements are added but some unknown supplementary features will remain undelivered. This core product will be used by customer to evolute the system and next increment is planned to develop.

During first requirement analysis phase, we will communicate with customers to specify as many requirements as possible. The first version of product with minimal and essential feature will be launched. Based on the feedback and experience of that version, list of additional features will be added. This process is repeated following the delivery of each increment, until the complete product is produced.

It may also possible that there is some change in the requirements of the customers over the time and therefore new design and implementation for this program would have to be prepared. Incremental model is quite flexible in this scenario.

There is not huge risk involved in this system. Moreover, the system architecture is also not that much complex. So, there is no need for spiral model of software development.

After some crucial increments, the final product will be deployed.



(Software Process Model)

2.7 Requirement Validation:

It’s a process of ensuring the specified requirements meet the customer needs. It’s concerned with finding problems with the requirements.

These problems can lead to extensive rework costs when these they are discovered in the later stages, or after the system is in service.

The cost of fixing a requirements problem by making a system change is usually much greater than repairing design or code errors. Because a change to the requirements usually means the design and implementation must also be changed, and re-tested.

So, this phase will validate the requirements which are gathered and an assurance with the customers that their requirements are the same as prescribed earlier.

After the extensive analysis of the problems in the system, we are familiarized with the requirement that the current system needs. The requirement that the system needs is categorized into the functional requirements and non-functional requirements. These requirements are listed below:

Functional Requirements

Functional requirements are the functions or features that must be included in any system to satisfy the business needs and be acceptable to the users. Based on this, the functional requirements of the system must require are as follows:

- The system should be able to generate an approximate share price.
- The system collects the accurate data from internet using various library and tools as prescribed in section 2.7 in consistent manner and regularly.
- The system will also count technical indicators and judge according to those indicators.
- The system will perform Textual analysis using online business articles and news.

Non-Functional Requirements

Non-functional requirement is a description of features, characteristics and attributes of the system as well as any constraints that may limit the boundaries of the proposed system. The non-functional requirements are essentially based on the performance, information, economy, control and security efficiency and services. These non-functional requirements are as follows:

- The system should provide better accuracy.
- The system should have simple interface for users to use.
- To perform efficiently in short amount of time.

The system assessment on the stocks from India’s Bombay Stock Exchange is carried out. For given day’s open index, day’s high, day’s low, volume and adjacent values along with the stock news textual data, our forecaster will forecast the final index price for given trading date.

2.8 Tools and Technology / Minimum Hardware and Software Requirements:

Minimum Hardware and Software Requirements:

- Operating System: Windows 7 or later versions.
- RAM: 2 GB or above.
- Processor: Intel or AMD.

Tools and Technology used:

- Python 3
- PyCharm IDE
- Python nsepy library
- NumPy
- Python pandas library
- Python pyplot library
- Python tkinter library
- Scikit-learn
- BeautifulSoup
- PAGE: PAGE is a cross-platform drag-and-drop GUI generator
- Tensorflow
- Keras

2.9 System Architecture:

Project is overall based upon the historical data which is going to be collected from nseindia.com using nsepy library of python. (See 2.8) After fetching the desired data they have been used for the predictions of related results. We are collecting the stock information day-wise.

The collected historical data will be stored in .csv format. Which will be used further as training data to predict the future stock price. We have taken regression approach, rather than classification approach.

The learning algorithms used here are LSTM, SVR and ARIMA. In fact, all these algorithms predict marginally different prices and ultimately the average of these three values is calculated, which is a better approximation of the future stock price.

Stock quotes are normalized by scaling its units so they occur in small range of 0.0. to 1.0. This has two advantages: first, the performance of algorithm speeds up.

Second, it is necessary condition for using some tools mentioned in previous section, like Keras, which runs on the top of Tensorflow.

The user is supposed to decide themselves whether to buy or sell the particular stock. Our system will only compute or indicate very useful factors which plays the major role in movement of stock price and it will help traders to take key decisions.

Sentimental Analysis:

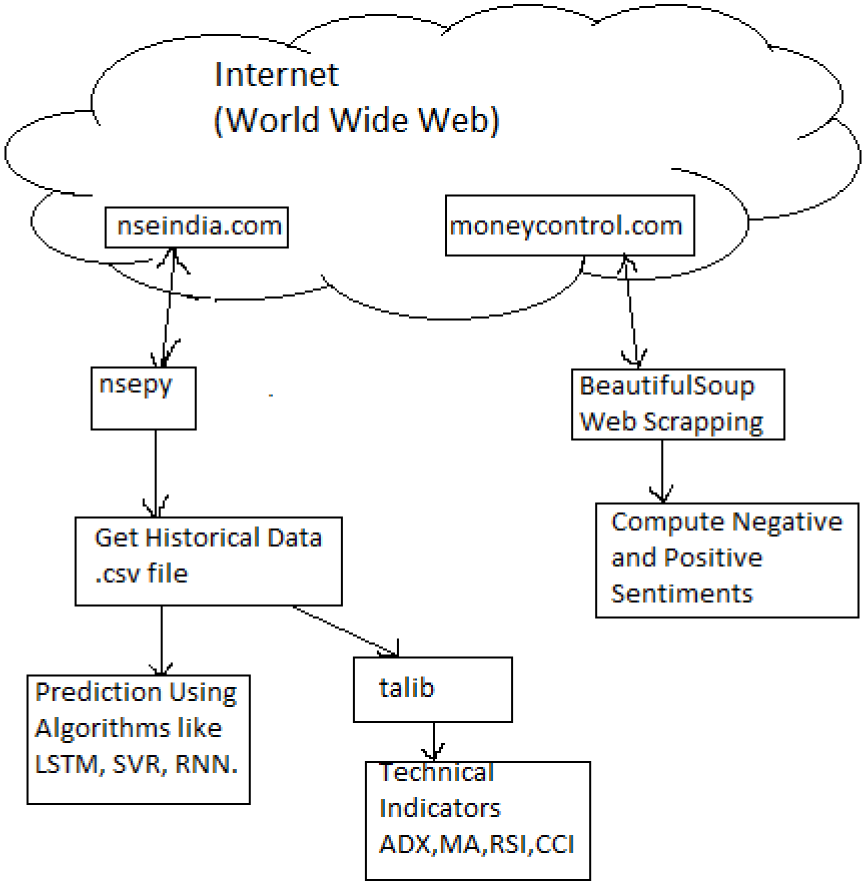
From moneycontrol.com website, we collect news articles and perform following key operations:

I. Stop Words Removal

II. Stemming

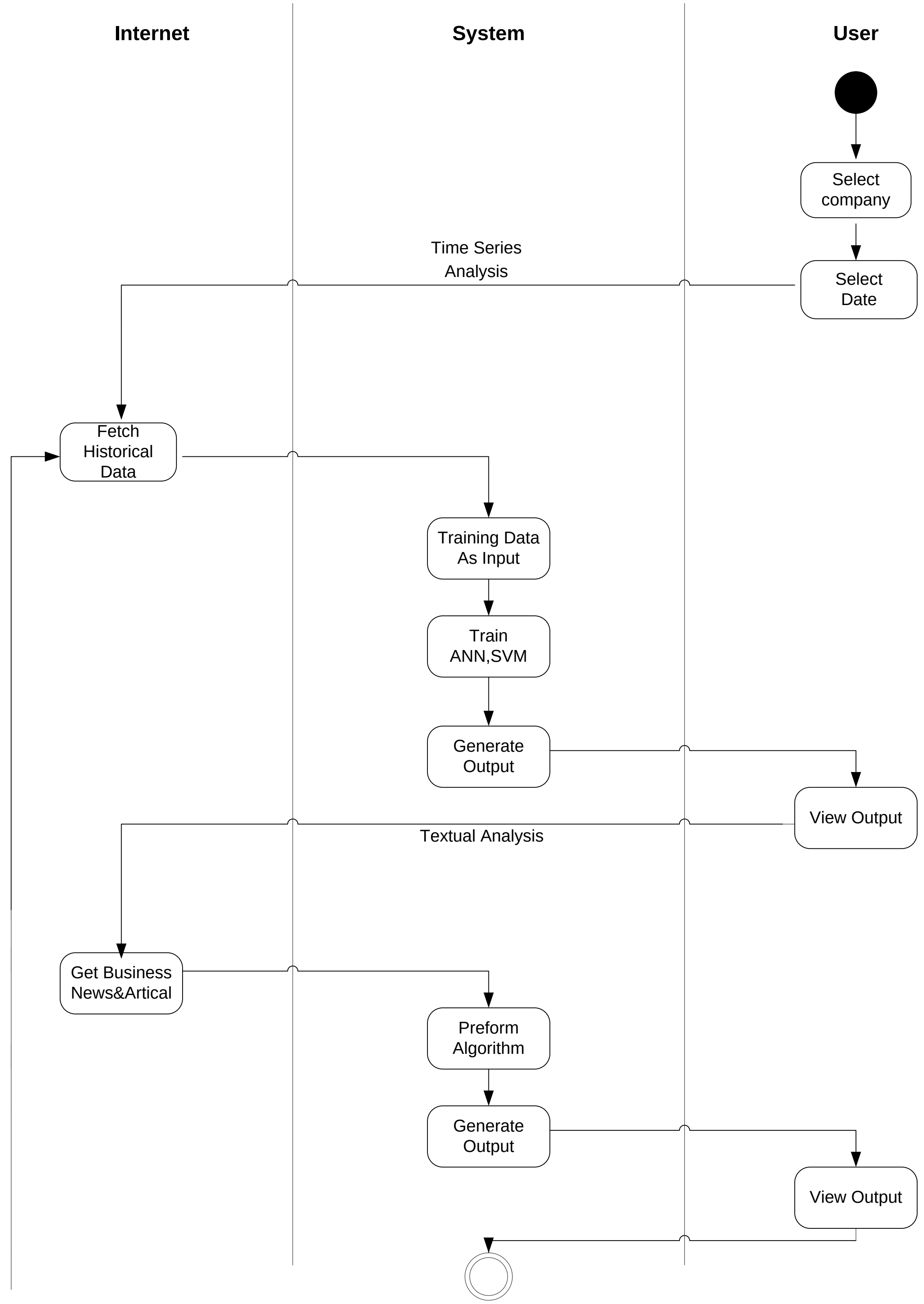
III. Key Phrases Extraction.

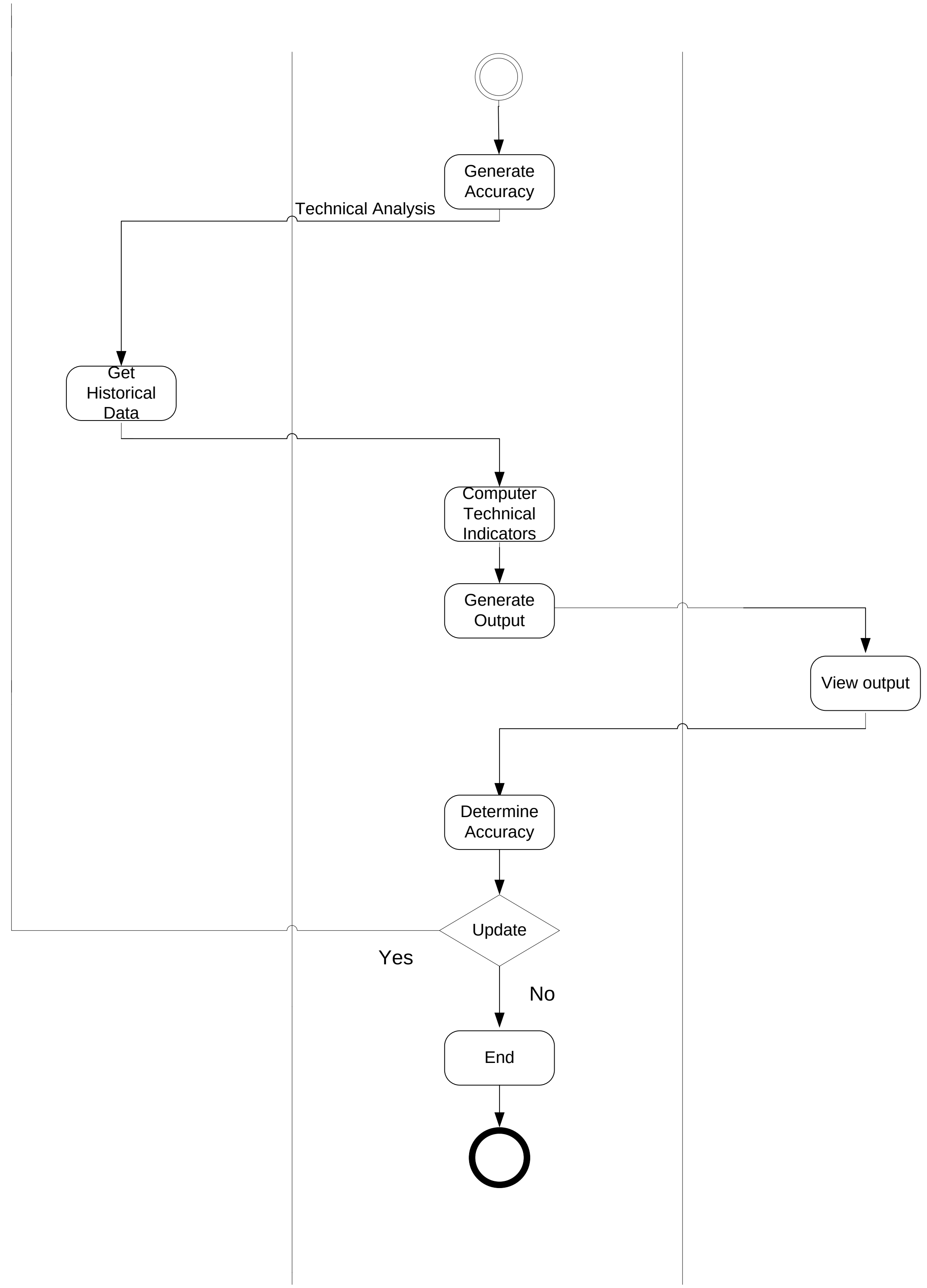
We used bag-of-words approach in sentimental analysis. In bag-of-words approach, we prepared dictionary of positive sentimental words and negative sentimental words. Then we compared the extracted words from news article page with the words from dictionary and according to that we have calculated the positive sentiments and negative sentiments.


Internet
(World Wide Web)
nseindia.com
moneycontrol.com
nsepy
BeautifulSoup Web Scrapping
Get Historical Data .csv file
Compute Negative and Positive Sentiments
Prediction Using Algorithms like LSTM, SVR, RNN.
talib
Technical Indicators ADX,MA,RSI,CCI

(System Architecture)

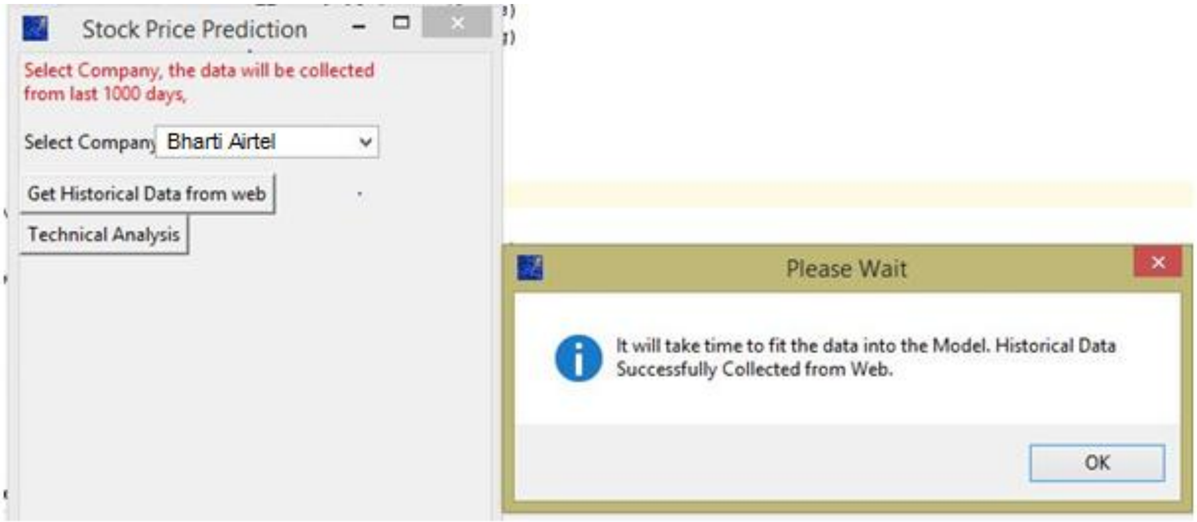
2.10 Activity Diagram:





(Activity Diagram)

3.2 System GUI:

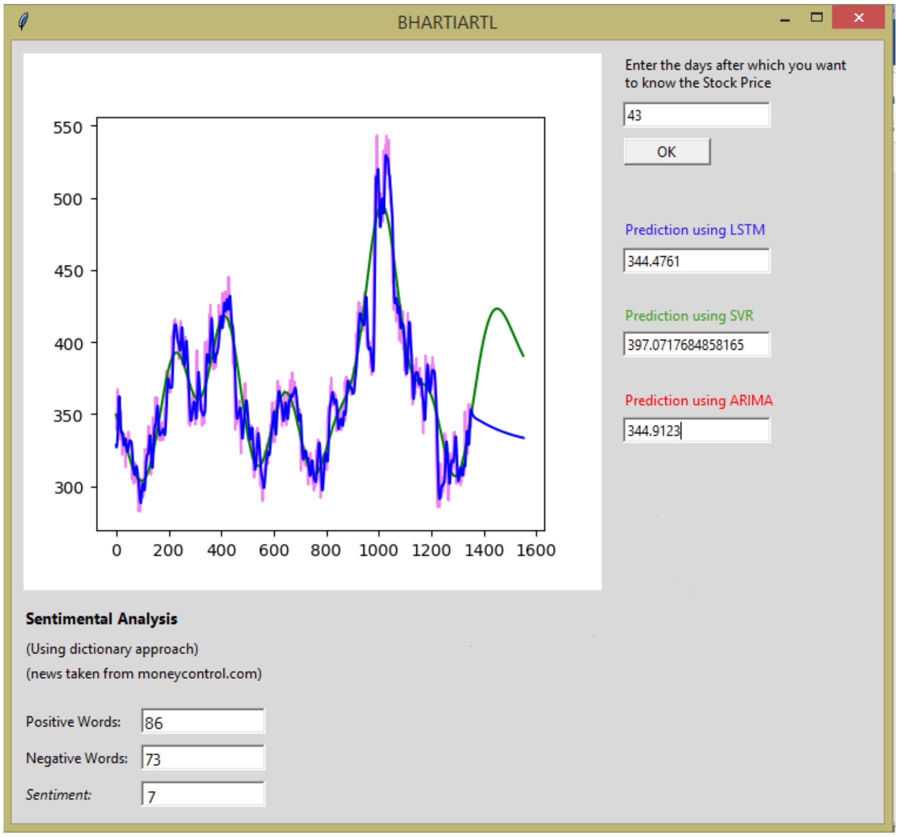


Select Company, the data will be collected from last 1000 days,

Select Company: Bharti Airtel

It will take time to fit the data into the Model. Historical Data Successfully Collected from Web.

(System GUI)



Enter the days after which you want to know the Stock Price: 43

Prediction using LSTM: 344.4761

Prediction using SVR: 397.0717684858165

Prediction using ARIMA: 344.9123

Sentimental Analysis

(Using dictionary approach)

(news taken from moneycontrol.com)

Positive Words: 86

Negative Words: 73

Sentiment: 7

(System GUI)

3

Chapter # 3: System Development

3.1 Coding Standards

3.2 Implementation Environment

3.3 Tools Explanation

3.1 Coding Standards:

- The project uses python programming language.
- We used several already available APIs, listed in section 2.7.
- Project doesn’t follow Object-oriented Programming approach.
- Coding has been done separately for different functionalities of the system. This way modularity is achieved. So, there are different python modules for calculating technical indicators, sentimental analysis and machine learning model.
- We use thread-safe version of tkinter.

3.2 Implementation Environment:

- The program is implemented in PyCharm IDE in python programming language.
- It also requires internet connection to connect with websites like moneycontrol.com and nseindia.com
- A laptop, with minimum of 1 GB RAM is required and Windows operating system is desired. However, the project was successfully tested on Ubuntu also.

3.3 Tools explanation :

- Python 3: Python is an interpreted high-level programming language for general-purpose programming.
- PyCharm IDE: PyCharm is an integrated development environment used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains. Here it is mainly used for debugging purpose. This Project contains several hundred lines of code.
- Python nsepy library: NSEpy is a library to extract historical and realtime data from NSE’s website. This Library aims to keep the API very simple. Python is a great tool for data analysis along with the scipy stack and the main objective of NSEpy is to provide analysis ready data-series for use with scipy stack. NSEpy can seamlessly integrate with Technical Analysis library (Acronymed TA-Lib, includes 200 indicators like MACD, RSI). This library would serve as a basic building block for

automatic/semi-automatic algorithm trading systems or backtesting systems for Indian markets.

- NumPy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. NumPy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
- Python pandas library: Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.
- Python pyplot library: matplotlib.pyplot is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc. In matplotlib.pyplot various states are preserved across function calls, so that it keeps track of things like the current figure and plotting area, and the plotting functions are directed to the current axes (please note that "axes" here and in most places in the documentation refers to the axespart of a figure and not the strict mathematical term for more than one axis).
- Python tkinter library: Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit. It is Not thread safe version.
- Python mttkinter library: It is thread safe version of tkinter. For projects which uses several threads and processes such as this, mttkinter is very useful tool.
- Scikit-learn: It is a free software machine learning library for the Python programming language. It features various classification, regression, clustering algorithms including support vector machines, random forests, gradient boosting, k-means and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

- BeautifulSoup: Beautiful Soup is a Python package for parsing HTML and XML documents (including having malformed markup, i.e. non-closed tags, so named after tag soup). It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping.
- PAGE: PAGE is a cross-platform drag-and-drop GUI generator, bearing a resemblance to Visual Basic. It allows one to easily create GUI windows containing a selection of Tk and ttk widgets. Required are Tcl/Tk 8.6 and Python 2.7+. PAGE is not an end-all, be-all tool, but rather one that attempts to ease the burden on the Python programmer. It is aimed at the user who will put up with a less than completely general GUI capability in order to get an easily generated GUI. A helper and learning tool, it does not build an entire application but rather is aimed at building a single GUI class and the boiler plate code in Python necessary for getting the GUI on the screen.
- Tensorflow: TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks.

4

Chapter # 4: Implementation Details

4.1 Learning Algorithms

4.2 Technical Analysis Indicators

4.3 Sentimental Analysis

4.1 Learning Algorithms:

Support Vector Machines(SVM) and Support Vector Regression(SVR):

In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting).

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. When data are not labeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The support vector clustering algorithm created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.

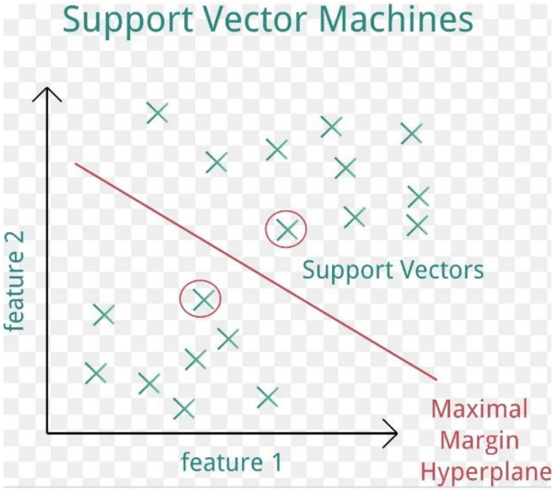
Applications:

SVMs can be used to solve various real world problems:

- SVMs are helpful in text and hypertext categorization as their application can significantly reduce the need for labeled training instances in both the standard inductive and transudative settings.
- Classification of images can also be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of relevance feedback. This is also true of image

segmentation systems, including those using a modified version SVM that uses the privileged approach as suggested by Vapnik.

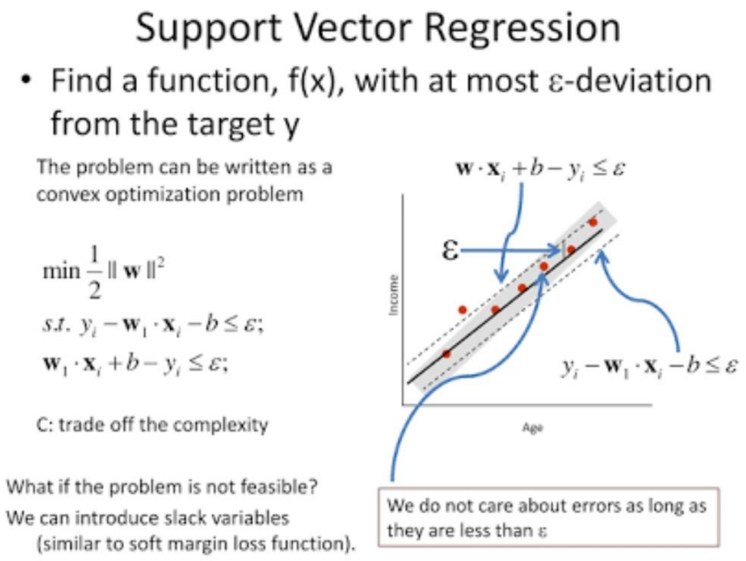
- Hand-written characters can be recognized using SVM.
- The SVM algorithm has been widely applied in the biological and other sciences. They have been used to classify proteins with up to 90% of the compounds classified correctly. Permutation tests based on SVM weights have been suggested as a mechanism for interpretation of SVM models. Support vector machine weights have also been used to interpret SVM models in the past. Postdocs interpretation of support vector machine models in order to identify features used by the model to make predictions is a relatively new area of research with special significance in the biological sciences.



SVM

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

When data are not labeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The support vector clustering algorithm created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.



Support Vector Regression

- Find a function, f(x), with at most ε-deviation from the target y

The problem can be written as a convex optimization problem

$$\min \frac{1}{2} \| \mathbf{w} \|^2$$

$$s.t.\ y_i - \mathbf{w}_1 \cdot \mathbf{x}_i - b \le \varepsilon;$$

$$\mathbf{w}_1 \cdot \mathbf{x}_i + b - y_i \le \varepsilon;$$

C: trade off the complexity

What if the problem is not feasible?
We can introduce slack variables
(similar to soft margin loss function).

We do not care about errors as long as they are less than ε

$\mathbf{w} \cdot \mathbf{x}_i + b - y_i \le \varepsilon$
ε
Income
$y_i - \mathbf{w}_1 \cdot \mathbf{x}_i - b \le \varepsilon$
Age

Support Vector Regression (SVR):

As explained in the idea of SVM is to constructs a hyperplane or set of hyperplanes in a high or infinite dimensional space, which can be used for classification. In the regression problem the same margin concept used in SVM is used. The goal of solving a regression problem is to construct a hyperplane that is as close to as many of the data points as possible. Choosing a hyperplane with small norm is considered as a main objective, while simultaneously minimizing the sum of the distances from the data points to the hyperplane. In case the of solving regression problems using SVM, SVM became known as the support vector regression (SVR) where the aim is to find a function *f* with parameters *w* and *b* by minimizing the following regression risk:

$$R(f) = \frac{1}{2}(x, w) + C \sum_{i=1}^{N} l\left(f(x_i), y_i\right)$$

C in Equation is a trade-off term, the margin in SVM is the first term which is used in measuring VC-dimension

$$f\,(x, w, b) = (w, \varphi\,(x)) + b,$$

In the Eq. $\varphi\,(x) : x \rightarrow \Omega$ is kernel function, mapping x into in the high dimensional space,

SVR and as proposed by. The −*ε* insensitive loss function is used as follows:

$$l\,(y, f\,(x)) = \begin{cases} 0, & \text{if} |y - f\,(x) < \varepsilon \\ |y - f\,(x)| - \varepsilon, & \text{Otherwise} \end{cases}$$

Equation of constrained minimization problem is equivalent to previous minimization.

$$Min \quad y\left(w, b, \zeta^* = \frac{1}{2}(w, w) + C\sum_{i=1}^{N}(\zeta_i + \zeta_{i*})\right)$$

Subject to:

$$y_i - ((w, \phi(x_i) + b)) \leq \varepsilon + \zeta_i,$$

$$((w, \phi(x_i)) + b) - y_i \leq \varepsilon + \zeta_{i*},$$

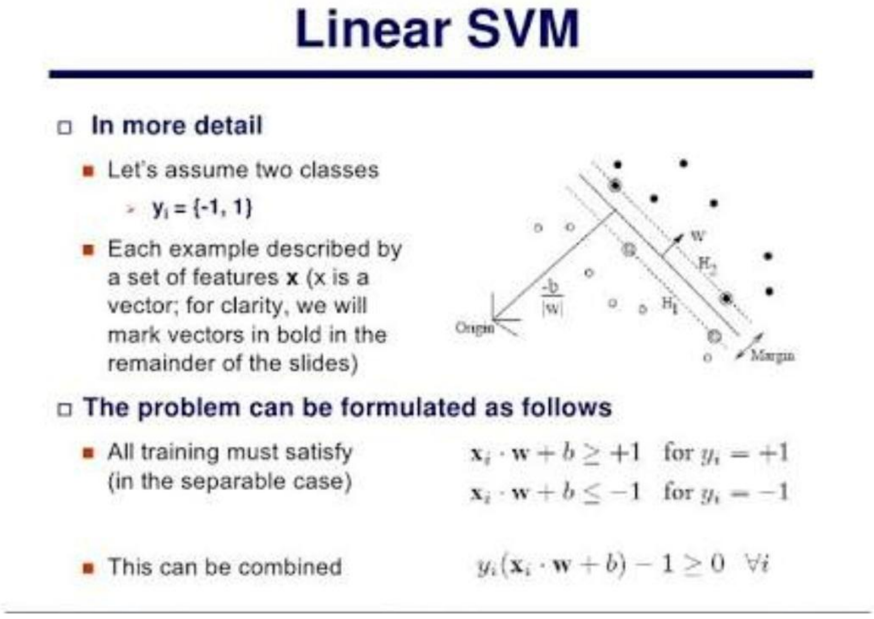
$$\zeta_i^* \geq 0$$

In sample *(x_i , y_i)* the ζ_i and $\zeta_i{}^*$ measure the up error and down error. Maximizing the dual function or in other words constructing the dual problem of this optimization problem (primal problem) by large method is a standard method to solve the above minimization problem. There are four common kernel functions; among these, this chapter will be utilizing the radial basis function (RBF). The RBF kernel function is the most widely applied in SVR. Equation is defining the kernel RBF, where the width of the RBF is denoted by σ. Furthermore, it is suggested that the value of σ must be between 0.1 and 0.5 in order for SVR model to achieve the best performance. In this chapter σ value is determined as 0.1.

$$K(x_i, x_i) = exp\left(\frac{-||x_i - x_j||^2}{2\sigma^2}\right)$$

SVR Linear:

We are reading the dataset and putting them into an array for further operations to be performed. We are interested in the closing price and the dates. So we read those values accordingly. After we have our desired array we can start our operations. Then we are getting our desired values. We are then plotting a graph of it with the help of Matplotlib.

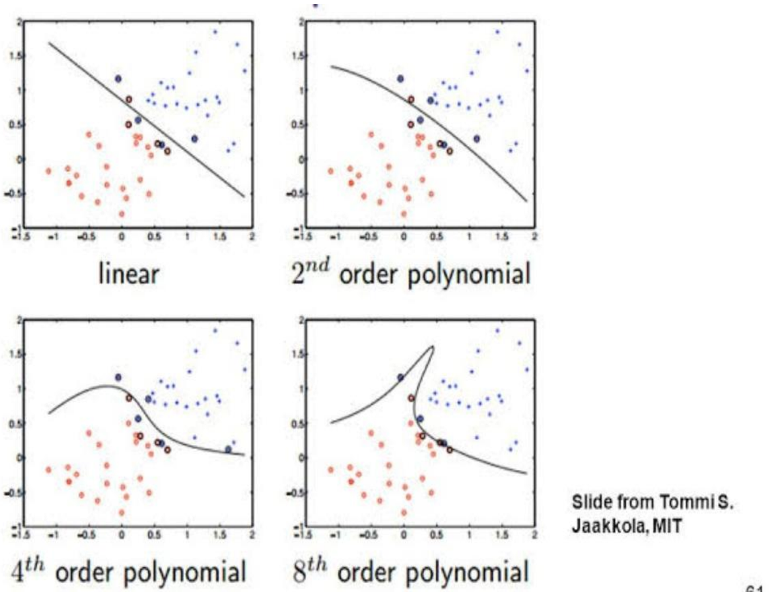


Linear SVM

- In more detail
  - Let's assume two classes
    - $y_i = \{-1, 1\}$
  - Each example described by a set of features **x** (x is a vector; for clarity, we will mark vectors in bold in the remainder of the slides)
- The problem can be formulated as follows
  - All training must satisfy (in the separable case) $\mathbf{x}_i \cdot \mathbf{w} + b \geq +1 \quad \text{for } y_i = +1$
 $\mathbf{x}_i \cdot \mathbf{w} + b \leq -1 \quad \text{for } y_i = -1$
  - This can be combined $y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0 \quad \forall i$

SVR Polynomial:

We are reading the dataset and putting them into an array for further operations to be performed. We are interested in the closing price and the dates. So we read those values accordingly. After we have our desired array we can start our operations. We are putting our values through SVR Polynomial Classifier. Then we are getting our desired values. We are then plotting a graph of it with the help of Matplotlib.



linear

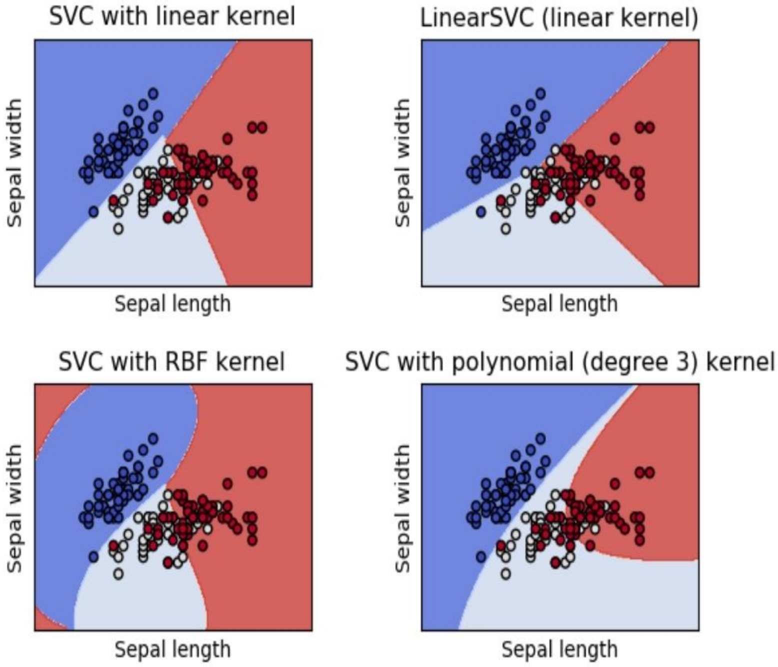
2^{nd} order polynomial

4^{th} order polynomial

8^{th} order polynomial

Slide from Tommi S. Jaakkola, MIT

SVR RBF:



We are reading the dataset and putting them into an array for further operations to be performed. We are interested in the closing price and the dates. So we read those values accordingly. The RBF kernel function is the most widely applied in SVR**.** After we have our desired array we can start our operations. We are putting our values through SVR RBF. Then we are getting our desired values. We are then plotting a graph of it with the help of Matplotlib.

To build the SVR and SVM model, the method which was addressed in the methodology section and in the framework prediction model section to determine the parameters of both model and the factor is applied. There are no general rules for choosing those parameters. Thus, this research adopted the most common approach to search for the best C and γ values.

The criteria of choosing the optimal C and γ parameters is by trying pairs of C and γ and the best combination of these parameters which can generate the minimum mean square error MSE is chosen to set up the prediction model. A set of exponentially growing sequences of C and γ are used and the best parameter combination results are: for the prediction next day closing price is *(C* = 200*, γ* = 0.001*)*, which gives the minimum mean square error MSE in the training data set and is therefore the best to set up the prediction model. For the next day closing price prediction, the best combination parameters are *(C* = 200*, γ* = 0.001*)*, which are the best combination parameters which give the minimum MSE in training data set.

Code-Segment:

```
###################################
## Support Vector Regression Model
###################################
x_train = np.arange(1, length, 1)
x_train = x_train.reshape(-1, 1)
y = data['Close'].values
clf = sklearn.svm.SVR(gamma=5e-5, C=200, epsilon=0.001, tol=0.001)
clf.fit(x_train, training_set)
svr_prediction = np.arange(1, length + 200, 1)
svr_prediction = svr_prediction.reshape(-1, 1)
global final_svr_prediction
final_svr_prediction = clf.predict(svr_prediction)
plt.plot(final_svr_prediction, color='green', label="Predicted Stock Price SVR")
y = y.reshape(-1, 1)
```

Autoregressive Model (AR):

The autoregressive model specifies that the output variable depends linearly on its own previous values and on a stochastic term (an imperfectly predictable term); thus the model is in the form of a stochastic difference equation. The notation *AR*(*p*) indicates an autoregressive model of order *p*. It is defined as:

$$X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_t$$

Where ϕ_1, ..., ϕ_i are the parameters of the model, *c* is constant, and ε*t* is white noise.

Moving Average Model (MA):

The moving-average model specifies that the output variable depends linearly on the current and various past values of a stochastic (imperfectly predictable) term. A moving average term in a time series model is a past error (multiplied by a coefficient).

$$X_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q}$$

where μ is mean of the series, $\theta_1, \dots, \theta_q$ are the parameters of the model and the ε_t, ε_{t-1}, ..., ε_{t-q} are white noise error terms.

Autoregressive Integrated Moving Average (ARIMA):

In this model, non-stationary data can be made stationary by differencing the series, Y t. The general model for Y t is written as,

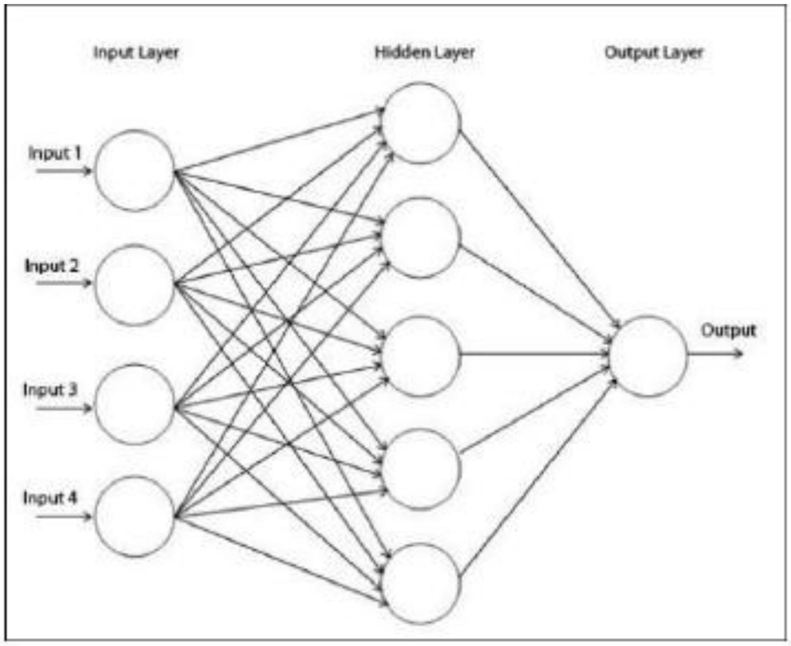
$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} \dots \phi_p Y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots \theta_q \epsilon_{t-q}$$

Where, Y_t is the differenced time series value, ϕ and θ are unknown parameters and ϵ are independent identically distributed error terms with zero mean. Here, Y t is expressed in terms of its past values and the current and past values of error terms. The ARIMA model combines three basic methods:

- Autoregressive (AR) – In auto-regressive model the values of a given time series data are regressed on their own lagged values, which is indicated by the “p” value in the
` model.
- Differencing (I-for Integrated) – This involves differencing the time series data to remove the trend and convert a non-stationary time series to a stationary one. This is indicated by the “d” value in the model. If d = 1, it looks at the difference between two time series entries, if d = 2 it looks at the differences of the differences obtained at d =1, and so forth.
- Moving Average (MA) – The moving average nature of the model is represented by the “q” value which is the number of lagged values of the error term.

Artificial Neural Network:

An (artificial) neural network is a network of simple elements called neurons, which receive input, change their internal state (activation) according to that input, and produce output depending on the input and activation. The network forms by connecting the output of certain neurons to the input of other neurons forming a directed, weighted graph. The weights as well as the functions that compute the activation can be modified by a process called learning which is governed by a learning rule.



ANN

Long Short Term Memory (LSTM):

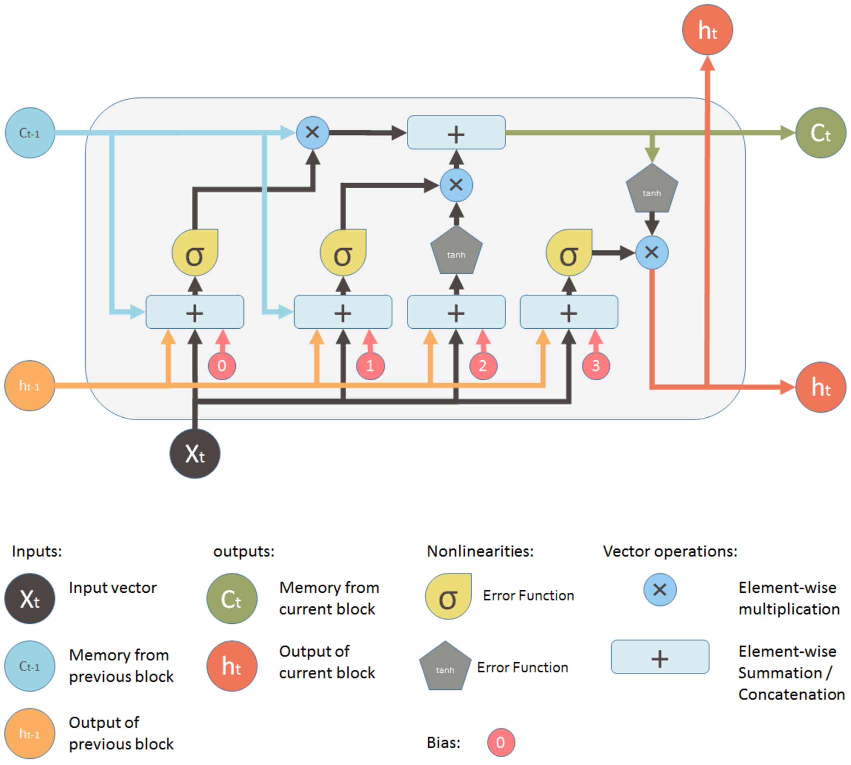
LSTM is form of Recurrent Neural Network (RNN) which is capable of holding long term dependencies. LSTM can remember the information for long period of time. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate.

- Cell: It is used to remember the values over arbitrary time intervals.
- Input Gate: It decides which information to keep in the cell.
- Output Gate: It is used to decide which part of cell state should be given as an output.
- Forget Gate: It is used to decide which information to throw away from the cell.

Code Snippet:

```
####################################################
   ##LSTM Model, 60 timesteps, 1 regression output##
####################################################
   sc = MinMaxScaler(feature_range=(0, 1))
```

```
training_set_scaled = sc.fit_transform(training_set)
x_train = []
y_train = []
length = len(training_set)
for i in range(60, length - 2):
    x_train.append(training_set_scaled[i - 60:i, 0])
    y_train.append(training_set_scaled[i, 0])
x_train, y_train = np.array(x_train), np.array(y_train)
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
regressor = Sequential()
regressor.add(LSTM(50, return_sequences=True, input_shape=(x_train.shape[1], 1)))
regressor.add(Dropout(0.2))
regressor.add(LSTM(50, return_sequences=True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(50, return_sequences=True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(50))
regressor.add(Dropout(0.2))
regressor.add(Dense(units=1))
s = Adam(lr=0.001)
regressor.compile(optimizer=s, loss='mean_squared_error')
regressor.fit(x_train, y_train, epochs=10, batch_size=32)
# Part 3 - Making the predictions and visualising the results
x_test = training_set_scaled[0:60]
x_test = x_test.tolist()
x_test1 = []
x_test1.append(x_test[0:60])
i = 0
global final_predicted_stock_price_lstm
final_predicted_stock_price_lstm = []
while i <= length + 200:
        x_test1 = np.array(x_test1)
        x_test1 = np.reshape(x_test1, (x_test1.shape[0], x_test1.shape[1], 1))
        predicted_stock_price = regressor.predict(x_test1)
        x_test1 = x_test1.tolist()
        x_temptest = x_test1[0]
        if i <= length - 2:
            x_temptest.append(training_set_scaled[i])
        else:
            x_temptest.append(predicted_stock_price[0].tolist())
        del x_test1[0][0]
        final_predicted_stock_price_lstm.append(sc.inverse_transform(predicted_stock_price))
        i = i + 1
    final_predicted_stock_price_lstm = np.array(final_predicted_stock_price_lstm)
    final_predicted_stock_price_lstm = final_predicted_stock_price_lstm.reshape(-1, 1)
    plt.plot(final_predicted_stock_price_lstm, color='blue', label='Predicted Stock Price
LSTM')
```



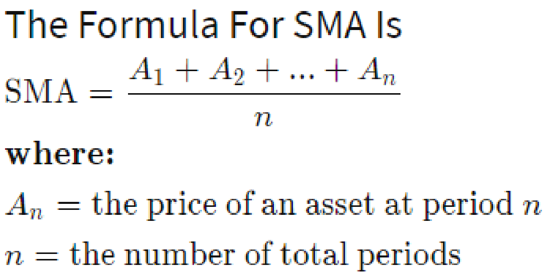
LSTM

4.2 Technical Analysis:

Simple Moving Average:

A simple moving average (SMA) is an arithmetic moving average calculated by adding recent closing prices and then dividing that by the number of time periods in the calculation average. A simple, or arithmetic, moving average that is calculated by adding the closing price of the security for a number of time periods and then dividing this total by that same number of periods. Short-term averages respond quickly to changes in the price of the underlying, while long-term averages are slow to react.

- The SMA is a technical indicator for determining if an asset price will continue or reverse a bull or bear trend.
- The SMA is calculated as the arithmetic average of an asset's price over some period.
- The SMA can be enhanced as an exponential moving average (EMA) that more heavily weights recent price action.



The Formula For SMA Is

$$\text{SMA} = \frac{A_1 + A_2 + \ldots + A_n}{n}$$

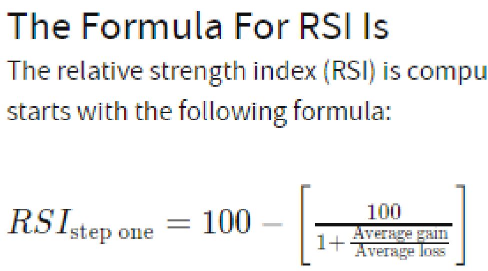
where:

A_n = the price of an asset at period n

n = the number of total periods

Relative Strength Index (RSI):

The relative strength index (RSI) is a momentum indicator that measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset. The RSI is displayed as an oscillator (a line graph that moves between two extremes) and can have a reading from 0 to 100.



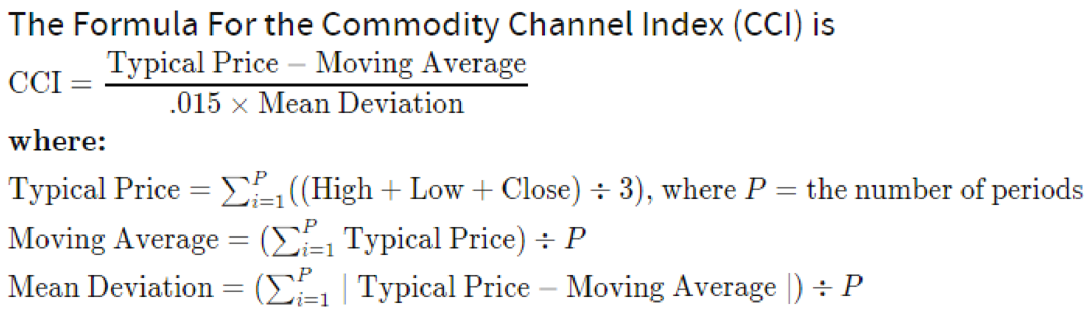
The Formula For RSI Is

The relative strength index (RSI) is computed with a two-part calculation that starts with the following formula:

$$RSI_{\text{step one}} = 100 - \left[\frac{100}{1+\frac{\text{Average gain}}{\text{Average loss}}}\right]$$

Commodity Channel Index :

- The CCI measures the difference between the current price and the historical average price.
- When the CCI is above zero it indicates the price is above the historic average. When CCI is below zero, the price is below the historic average.
- High readings of 100 or above, for example, indicate the price is well above the historic average and the trend has been strong to the upside.
- Low readings below -100, for example, indicate the price is well below the historic average and the trend has been strong to the downside.
- Going from negative or near-zero readings to +100 can be used as a signal to watch for an emerging uptrend.
- Going from positive or near-zero readings to -100 may indicate an emerging downtrend.
- CCI is an unbounded indicator meaning it can go higher or lower indefinitely. For this reason, overbought and oversold levels are typically determined for each individual asset by looking at historical extreme CCI levels where the price reversed from.



The Formula For the Commodity Channel Index (CCI) is

$$\text{CCI} = \frac{\text{Typical Price} - \text{Moving Average}}{.015 \times \text{Mean Deviation}}$$

where:

$$\text{Typical Price} = \textstyle\sum_{i=1}^{P}((\text{High} + \text{Low} + \text{Close}) \div 3), \text{ where } P = \text{the number of periods}$$

$$\text{Moving Average} = (\textstyle\sum_{i=1}^{P} \text{Typical Price}) \div P$$

$$\text{Mean Deviation} = (\textstyle\sum_{i=1}^{P} \mid \text{Typical Price} - \text{Moving Average} \mid) \div P$$

Average Directional Index(ADX) :

Trading in the direction of a strong trend reduces risk and increases profit potential. The average directional index (ADX) is used to determine when the price is trending strongly. In many cases, it is the ultimate trend indicator. After all, the trend may be your friend, but it sure helps to know who your friends are. In this article, we'll examine the value of ADX as a trend strength indicator.

ADX is used to quantify trend strength. ADX calculations are based on a moving average of price range expansion over a given period of time.

ADX is plotted as a single line with values ranging from a low of zero to a high of 100. ADX is non-directional; it registers trend strength whether price is trending up or down.

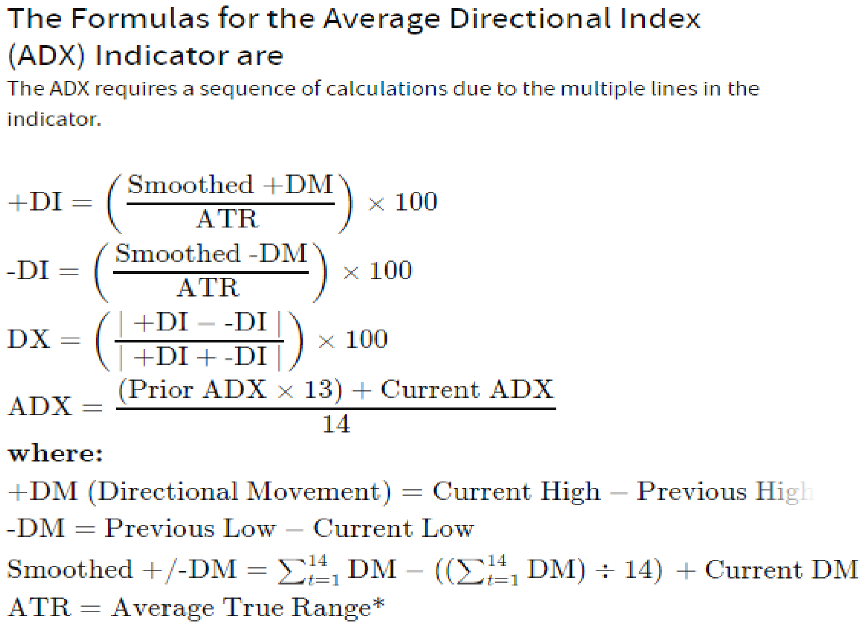
When the +DMI is above the -DMI, prices are moving up, and ADX measures the strength of the uptrend. When the -DMI is above the +DMI, prices are moving down, and ADX measures the strength of the downtrend.

ADX values help traders identify the strongest and most profitable trends to trade. The values are also important for distinguishing between trending and non-trending conditions. Many traders will use ADX readings above 25 to suggest that the trend is strong enough for trend-trading strategies. Conversely, when ADX is below 25, many will avoid trend-trading strategies.

ADX Value	**Trend Strength**
0-25	Absent or Weak Trend
25-50	Strong Trend
50-75	Very Strong Trend
75-100	Extremely Strong Trend

- Designed by Welles Wilder for commodity daily charts, but can be used in other markets or other timeframes.
- The price is moving up when +DI is above -DI, and the price is moving down when -DI is above +DI.

- Crosses between +DI and -DI are potential trading signals as bears or bulls gain the upper hand.
- The trend has strength when ADX is above 25. The trend is weak or the price is trendless when ADX is below 20, according to Wilder.
- Non-trending doesn't mean the price isn't moving. It may not be, but the price could also be making a trend change or is too volatile for a clear direction to be present.



The Formulas for the Average Directional Index (ADX) Indicator are

The ADX requires a sequence of calculations due to the multiple lines in the indicator.

$$+\text{DI} = \left(\frac{\text{Smoothed } +\text{DM}}{\text{ATR}}\right) \times 100$$

$$\text{-DI} = \left(\frac{\text{Smoothed -DM}}{\text{ATR}}\right) \times 100$$

$$\text{DX} = \left(\frac{|\,+\text{DI} - \text{-DI}\,|}{|\,+\text{DI} + \text{-DI}\,|}\right) \times 100$$

$$\text{ADX} = \frac{(\text{Prior ADX} \times 13) + \text{Current ADX}}{14}$$

where:

$+\text{DM (Directional Movement)} = \text{Current High} - \text{Previous High}$

$\text{-DM} = \text{Previous Low} - \text{Current Low}$

$\text{Smoothed } +/\text{-DM} = \sum_{t=1}^{14} \text{DM} - \left(\left(\sum_{t=1}^{14} \text{DM}\right) \div 14\right) + \text{Current DM}$

$\text{ATR} = \text{Average True Range*}$

4.3 Sentimental Analysis:

The purpose of this module is to obtain the sentiment value of latest news headlines regarding each stock and output its average as sentiment value to fuzzy module.

The steps used in this module are as follows:

1. **Data Collection:**

   The data is collected by crawling through Indian Financial news website www.moneycontrol.com. Minimum 4 news Headlines are scraped for each stock and stored against the company Symbol.

2. **Tokenizing:**

   Each news headline is broken down into sentences and then in turn broken down into the words.

3. **Lemmatizing:**

It is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form. For example, “the boy's cars are different colours” reduces to “the boy car be differ colour”.

4. **Finding Most Informative Features**

Words that contribute most in adding polarity to a sentence are found.

Positive	**Negative**
Buy	Sell
Up	Down
Rise	Dip
Jump	Hold
Strong	Bear
Support	Impact
Grow	Decline
Fold	Fall
Double	Loss
Bag	Debt

In .csv file we have to make separate entry for lowercase words.

5. Classifying features into positive and negative

These are then classified into positive and negative using nltk packages.

6. Adding these features to the sentiment analyser lexicon

These words are then added to the sentiment analyser wordlist with appropriate strength for positive and negative words

7. Classifying the testing data into positive and negative sentiments using training set.

5

Chapter # 5 : Testing

5.1 Testing Strategy

5.2 Testing Methods

5.3 Test Cases

5.1 Testing Strategy

Unit Testing:

Unit testing is defined as a type of software testing where individual units/ components of a software are tested.

Unit Testing of software applications is done during the development (coding) of an application. The objective of Unit Testing is to isolate a section of code and verify its correctness. In procedural programming, a unit may be an individual function or procedure. Unit Testing is usually performed by the developer.

Unit testing is first level of testing done before integration testing. Unit testing is a White Box testing technique that is usually performed by the developer. Though, in a practical world due to time crunch or reluctance of developers to tests, QA engineers also do unit testing.

Integration Testing:

This testing is done after all the groups have been completed to test their relation. This is particularly necessary and important as these modules have been made solely for the purpose of integration with the existing organization.

It will be used to check check if all the variables data work together in cohesion. Our system work in cohesion of different components.

Validation Testing:

The tester should look at the project from organization perspective and make sure it defines and reflects what customers have in mind. It should align with client’s goals without being biased towards technicalities.

Does input and output meet the requirements, i.e. input output synchronization according to the proposed algorithm.

Validation Testing ensures that the product actually meets the client's needs.

System Testing:

System Testing is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

It is a black box testing technique performed to evaluate the complete system the system's compliance against specified requirements. It tests overall behavior of the system from end-user’s point of view.

5.2 Test Methods:

Unit Test Plan:

There are three components in our system: Prediction using Learning models, Sentimental Analysis, Technical Analysis. We will test all these components. Check to see if the source data can be read by the module and produce the required result.

We have taken three learning models into account. We will check each model whether it is performing or not.

Integration Test Plan:

In machine learning algorithms-based components, we have taken three different learning models.

Sentimental analysis and Technical Analysis components are totally different and independent. Each of these components will be tested for each company in this Testing plan.

In Unit testing we tested all learning models independently. In this test, combination of all learning algorithms is integrated together.

Validation Test Plan:

This test is performed. Full table of Sentimental and Technical analysis components is being checked with proper input output entries. The predicted stock price for companies is also being checked by plotting in the graph.

System Test Plan:

All components of program along with all the components and data will be executed and the output will be checked and when some code segment will be found inefficient or buggy, it is being debugged by using PyCharm IDE.

5.3 Test Cases:

Pre-condition: User requires internet connection to connect with websites like nseindia.com and moneycontrol.com to collect data as mentioned earlier.

Unit Testing:

S No.	Unit to be tested	Expected Result	Actual Result	Status (Pass/Fail)
1.	Get data from nseindia.com using numpy library	Dataframe should be returned consisting last 4 years stock price	Data is getting successfully, but when server is down or due to unforeseen problems, data cannot be received.	Partially Passed
2.	Save data on local computer in .csv format	The data should be stored on local computer in default directory (Desktop) in .csv format.	Data successfully stored.	Passed
3.	Train the data using SVR model	Data should be trained using most appropriate learning parameters as specified in chapter 4 with appropriate error function.	Data is trained successfully.	Passed

4.	Train the data using LSTM model.	Data should be trained using most appropriate hidden layers and time steps as specified in chapter 4 with appropriate error function.	Data trained successfully	Passed
5.	Train data using ARIMA model	Data should be trained using most appropriate lag value	Data trained successfully	Passed
6.	Get financial news from moneycontrol.com	Each word of the text should be extracted	Each word of the news article except articles (a, an, the) and special symbols.	Passed
7.	Sentimental Analysis	Each extracted word should be compared with each word in the dictionary, specifying positive or negative sentiment. Count positive and negative sentiment.	Each extracted word is compared with each word in the dictionary, specifying positive or negative sentiment. Counting of positive and negative sentiment performed successfully.	Passed
8.	Technical Analysis using Historical Data	Technical Indicators like CCI, RSI, ADX and SMA should be computed.	All indicators are computed successfully.	Passed

Integration Testing:

S No.	Testing Step	Expected Result	Actual Result	Status (Pass/Fail)
1.	Checking all the learning models and getting data from nseindia.com	The data got from nseindia.com should be trained using proposed learning models and then plot the actual data and predicted data on the Graph	The operation performed successfully.	Passed
2.	Prepare .csv file containing details of Sentimental Analysis.	.csv file containing details of sentimental analysis should be generated.	.csv file generated successfully.	Passed
3.	Prepare .csv file containing details of Technical Analysis.	.csv file containing details of sentimental analysis should be generated.	.csv file generated successfully.	Passed

Validation Testing:

S No.	Testing Step	Expected Result	Actual Result	Status (Pass/Fail)
1.	Check if all the functional requirements are satisfied.	All the functional requirements specified in Section 2.7	The operation performed paritally.	Partially Passed

System Test:

All components of program along with all the components and data is executed and the output is checked and when some code segment will be found inefficient or buggy, it is being debugged by using PyCharm IDE. And the final version of project is deployed after this test.

6

Chapter # 6: Conclusion

6.1 Findings

6.2 Conclusion

6.3 Future Work

6.1 Findings:

The system evaluation on the stocks of top 100 companies from India's National Stock Exchange (NSE) is carried out. For given day's closing value and adjacent values along with the stock news textual data, our system will predict the closing index value for particular trading day. User just has to mention after how many days they want the stock price.

Our predictive model is evaluated on NSE market on the financial historical stock data over the training period of June 2014 to May 2015. The news data is collected from the financial web site www.nseindia.com using nsepy library and news articles from www.moneycontrol.com. The news data is collected day-wise. The stock quotes corresponding to each trading day were downloaded from nseindia.com.

The accuracy of the system is measured as the percentage of the predictions that were correctly determined by the system. For instance, if the system forecasts an upward trend and the index indeed goes up, it is supposed to be correct, otherwise, if the index goes down or remains stable for an uptrend, it is assumed to be wrong. It is not directly displayed to the end user, but on the backend we are constantly doing that and according to which we modify our learning model to increase performance and efficiency.

6.2 Conclusion:

Evaluating the Stock market prediction has at all times been tough work for analysts. Thus, we attempt to make use of vast written data to forecast the stock market indices. If we join both techniques of textual mining and numeric time series analysis, the user can more accurately get idea of how the market is following trends and can extract useful information. Variants of Artificial neural network, like RNN and LSTM are qualified to forecast NSE market upcoming trends.

Financial analysts, investors can use this prediction model to take trading decision by observing market behavior.

6.3 Future Work:

- Sentimental analysis not programming well enough, so this functionality needs to be improved.
- More learning models could be implemented to predict better future prices. Performance and efficiency could be enhanced further.
- GUI can be enhanced, with better quality of graph plotting.
- Sentimental analysis could be done using twitter. A learning model for this functionality could be proposed.
- More work on refining key phrases extraction will definitely produce better results. Enhancements in the preprocessor unit of this system will help in improving more accurate predictability in stock market.
- Twitter feeds message board, Extracting RSS feeds and news.
- Considering internal factors of the company likes Sales, Assets etc.

References

[1] https://www.investopedia.com/articles/stocks/09/indian-stock-market.asp

[2]Raj Kumar, Anil Balara(2014), “Time Series Forecasting Of Nifty Stock Market Using Weka”, JRPS International Journal for Research Publication & Seminar Vol 05 Issue 02.

[3] Bashambu Shallu, Sikka Aman, Negi Pallav, “Stock Price Prediction using Neural Networks”, International Journal of Advance Research, Ideas and Innovations in Technology.

[4] A similar project report: https://www.slideshare.net/anilsth91/stock-market-analysis-and-prediction

[5]https://www.investopedia.com/terms/t/timeseries.asp

[6] https://image.slidesharecdn.com/efficientmarkethypothesis-111019030158-phpapp01/95/efficient-market-hypothesis-5-728.jpg?cb=1318993357

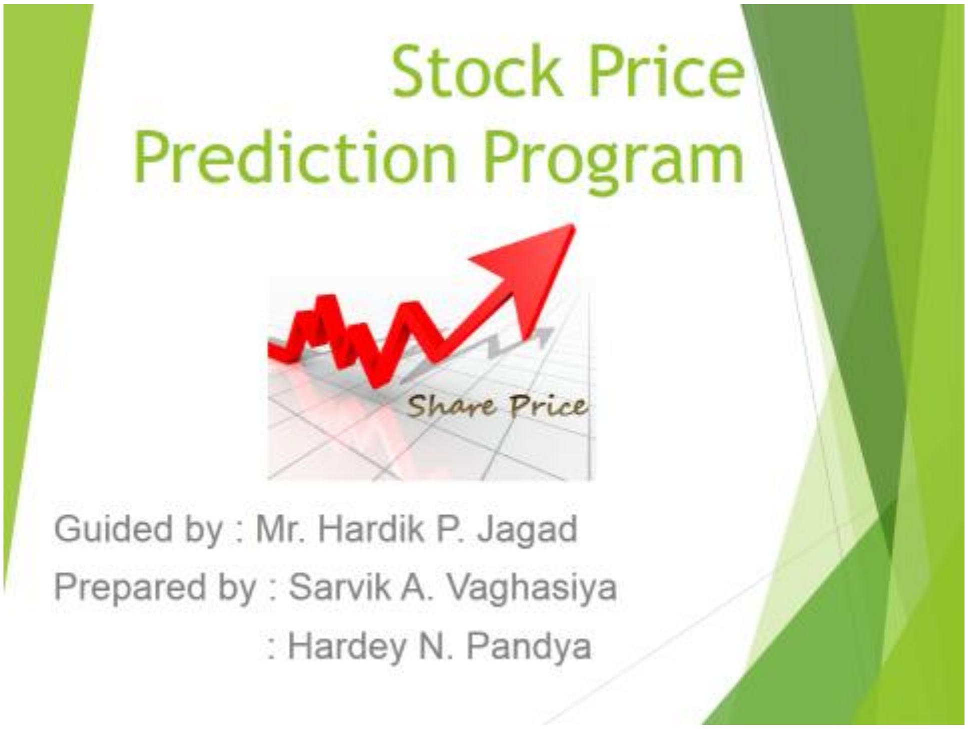
[7] Luca Di Persio, Oleksandr, Honchar(2016), “Artificial Neural Networks architectures for stock price prediction: comparisons and applications”, International Journal of Circuits, Systems and Signal Processing Volume 10.

[8] “Efficient Machine Learning Techniques for Stock Market Prediction”

[9] Xinjie Di (2014),“Stock Trend Prediction with Technical Indicators using SVM”, SCPD student from Apple Inc.

[10] Bashar Al-hnaity and Maysam Abbod, “Predicting Financial Time Series Data Using Hybrid Model”

Appendix



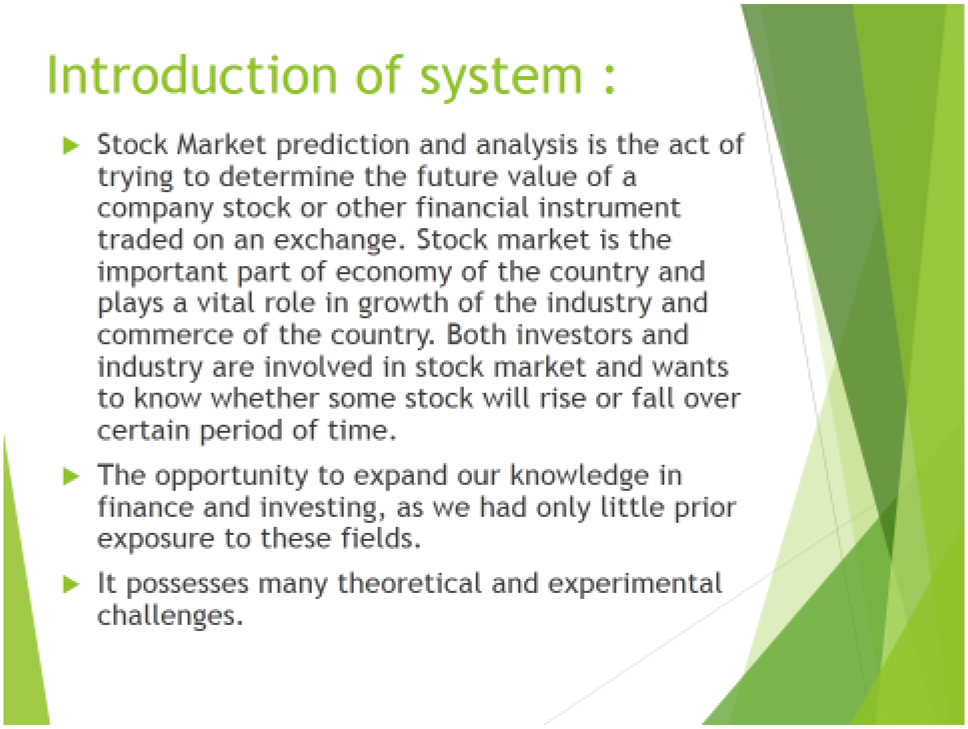
Stock Price Prediction Program

Share Price

Guided by : Mr. Hardik P. Jagad

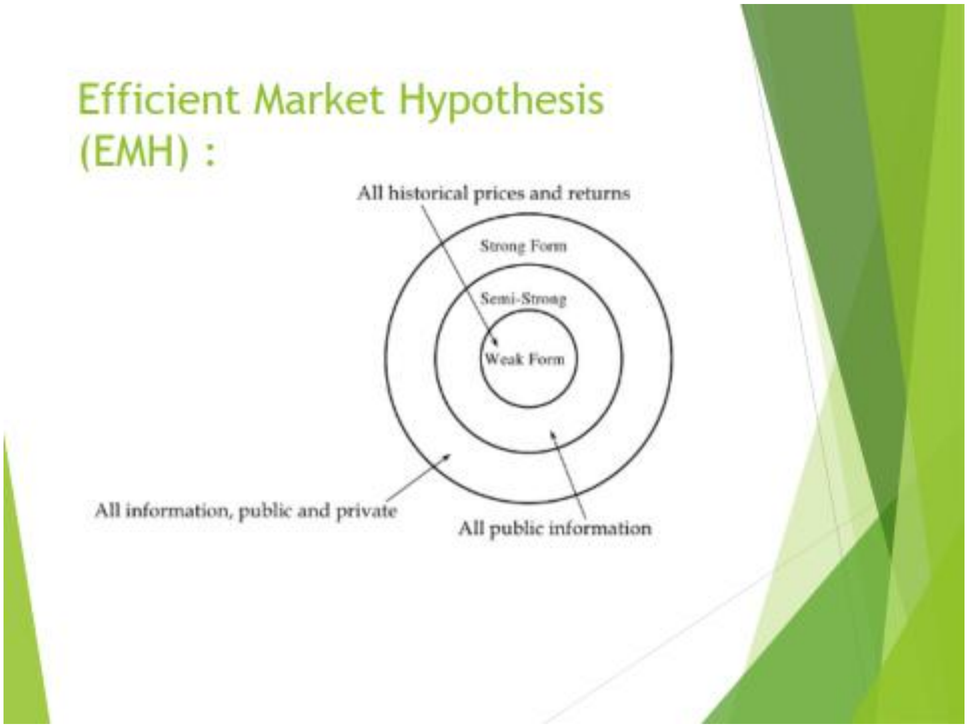
Prepared by : Sarvik A. Vaghasiya

: Hardey N. Pandya



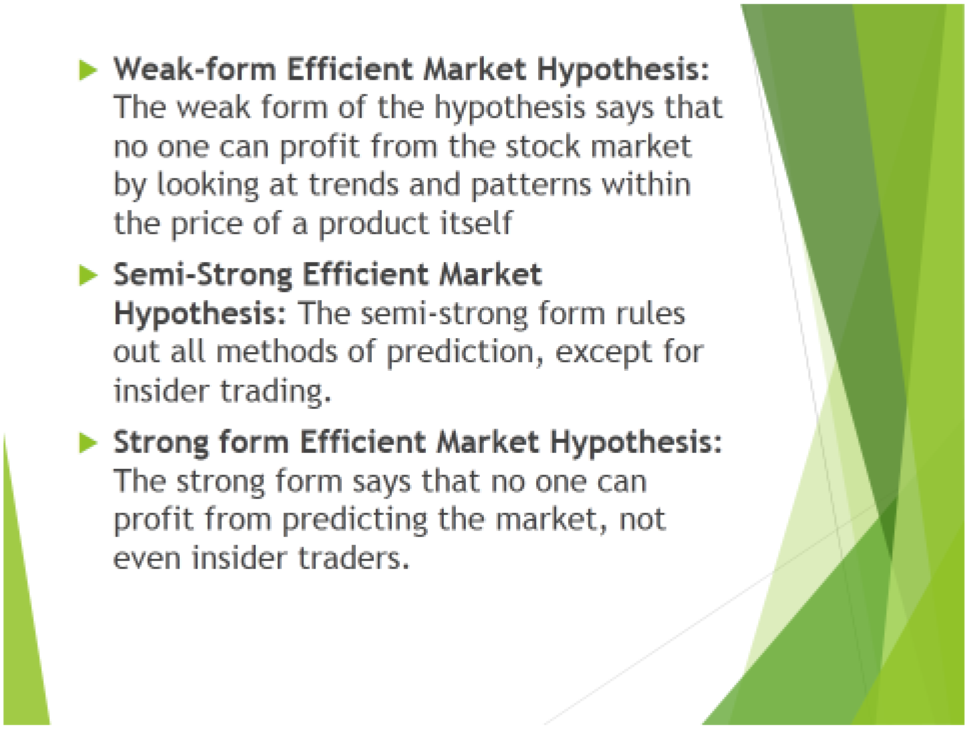
Introduction of system :

- Stock Market prediction and analysis 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 country and plays a vital role in growth of the industry and commerce of the country. Both investors and industry are involved in stock market and wants to know whether some stock will rise or fall over certain period of time.
- The opportunity to expand our knowledge in finance and investing, as we had only little prior exposure to these fields.
- It possesses many theoretical and experimental challenges.

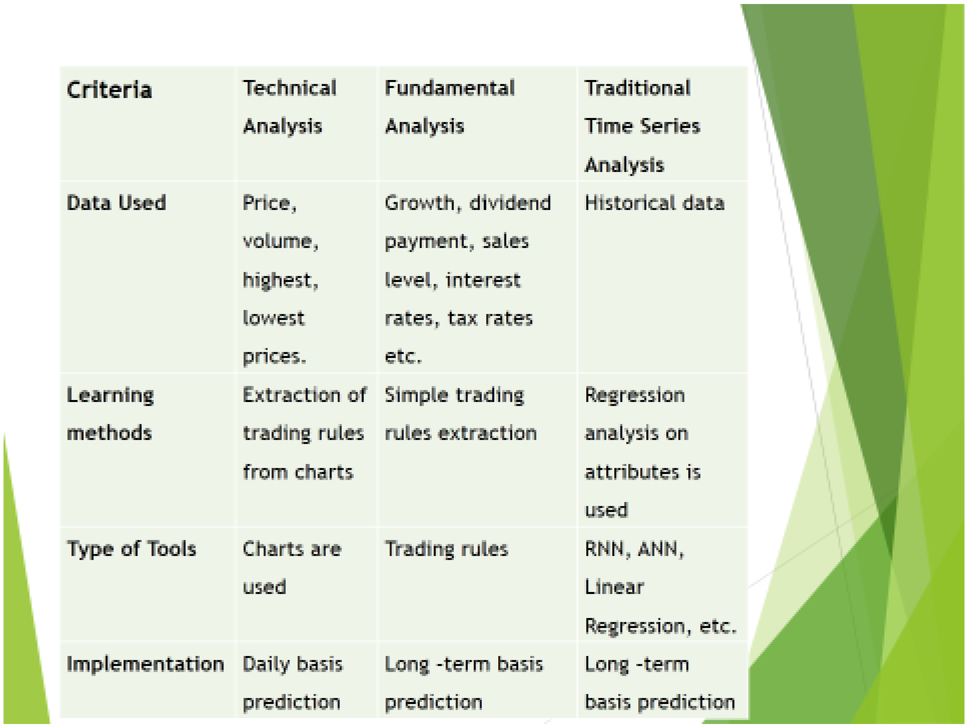


Efficient Market Hypothesis (EMH) :

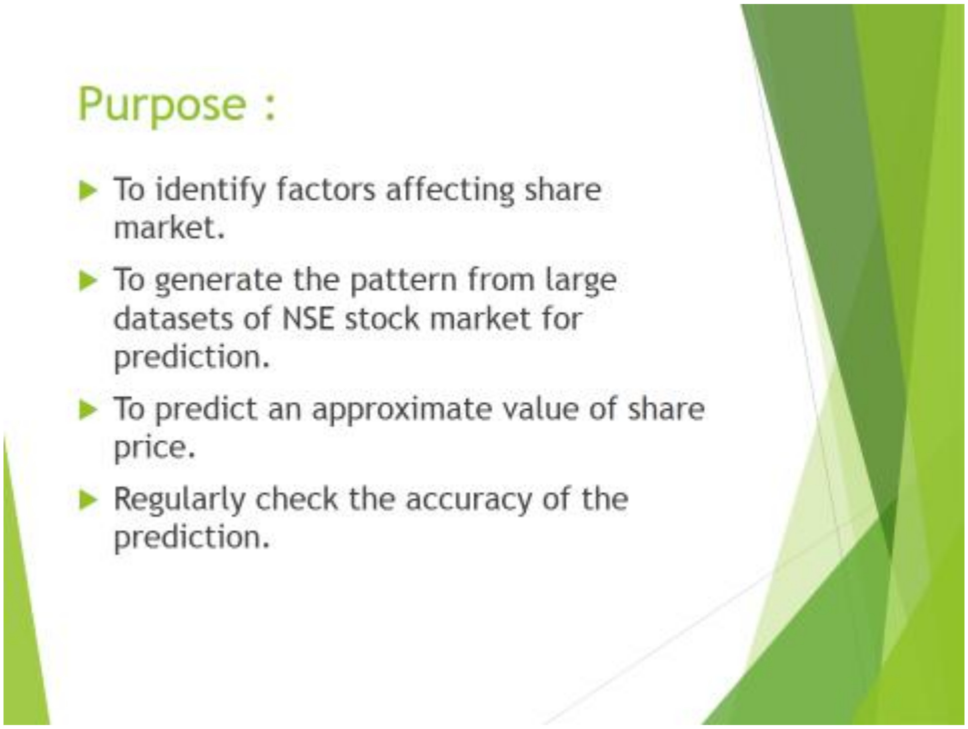
All historical prices and returns
Strong Form
Semi-Strong
Weak Form
All information, public and private
All public information



- **Weak-form Efficient Market Hypothesis:** The weak form of the hypothesis says that no one can profit from the stock market by looking at trends and patterns within the price of a product itself
- **Semi-Strong Efficient Market Hypothesis:** The semi-strong form rules out all methods of prediction, except for insider trading.
- **Strong form Efficient Market Hypothesis:** The strong form says that no one can profit from predicting the market, not even insider traders.

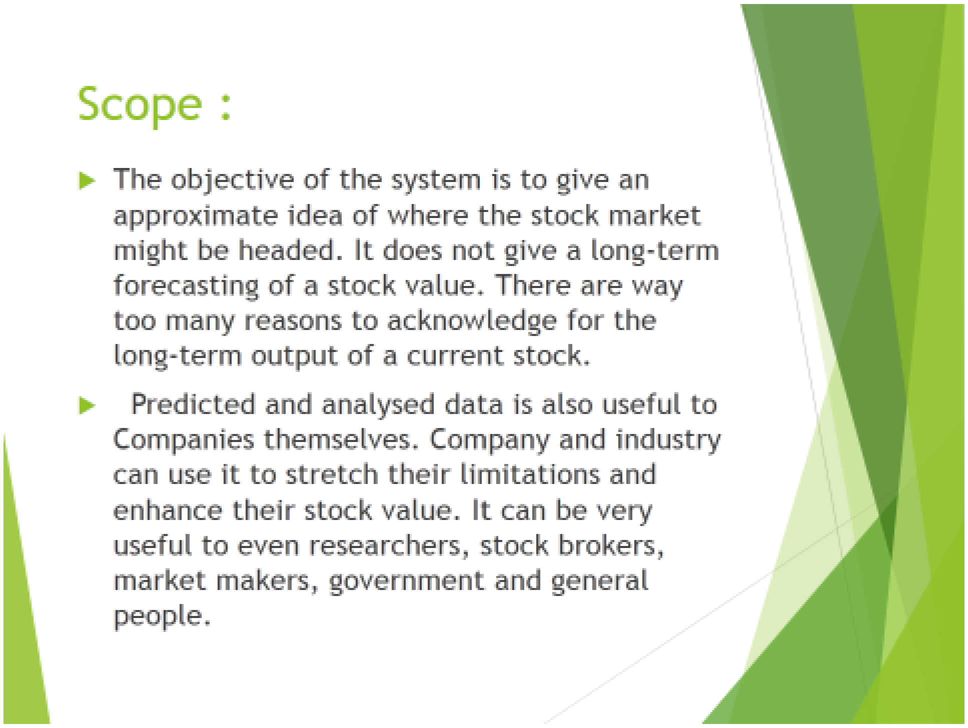


Criteria	Technical Analysis	Fundamental Analysis	Traditional Time Series Analysis
Data Used	Price, volume, highest, lowest prices.	Growth, dividend payment, sales level, interest rates, tax rates etc.	Historical data
Learning methods	Extraction of trading rules from charts	Simple trading rules extraction	Regression analysis on attributes is used
Type of Tools	Charts are used	Trading rules	RNN, ANN, Linear Regression, etc.
Implementation	Daily basis prediction	Long -term basis prediction	Long -term basis prediction



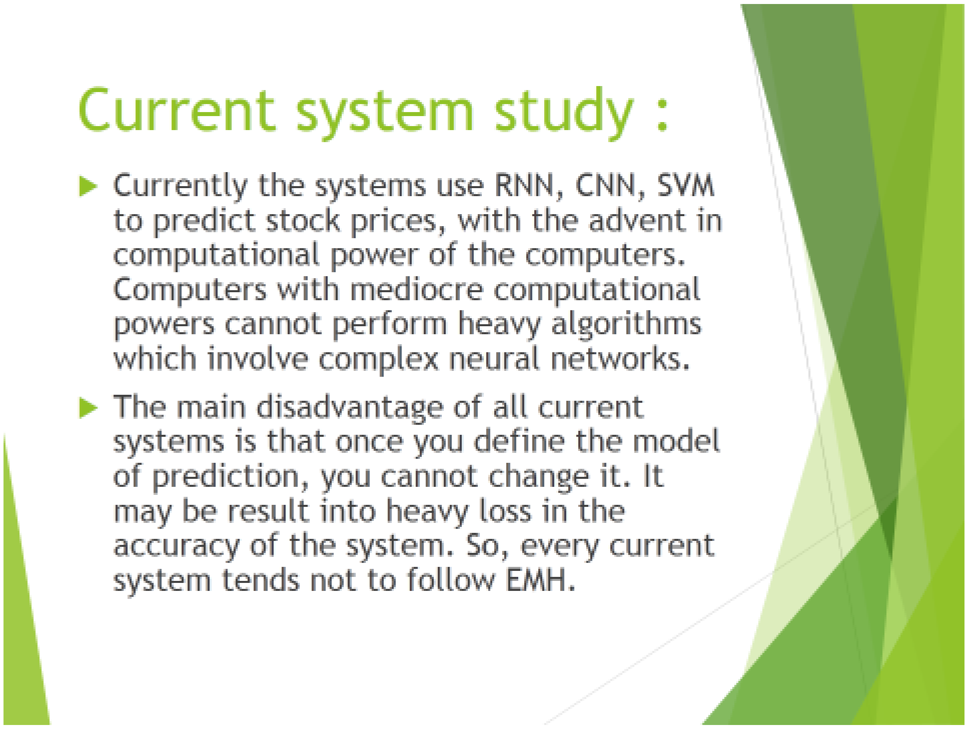
Purpose :

- To identify factors affecting share market.
- To generate the pattern from large datasets of NSE stock market for prediction.
- To predict an approximate value of share price.
- Regularly check the accuracy of the prediction.



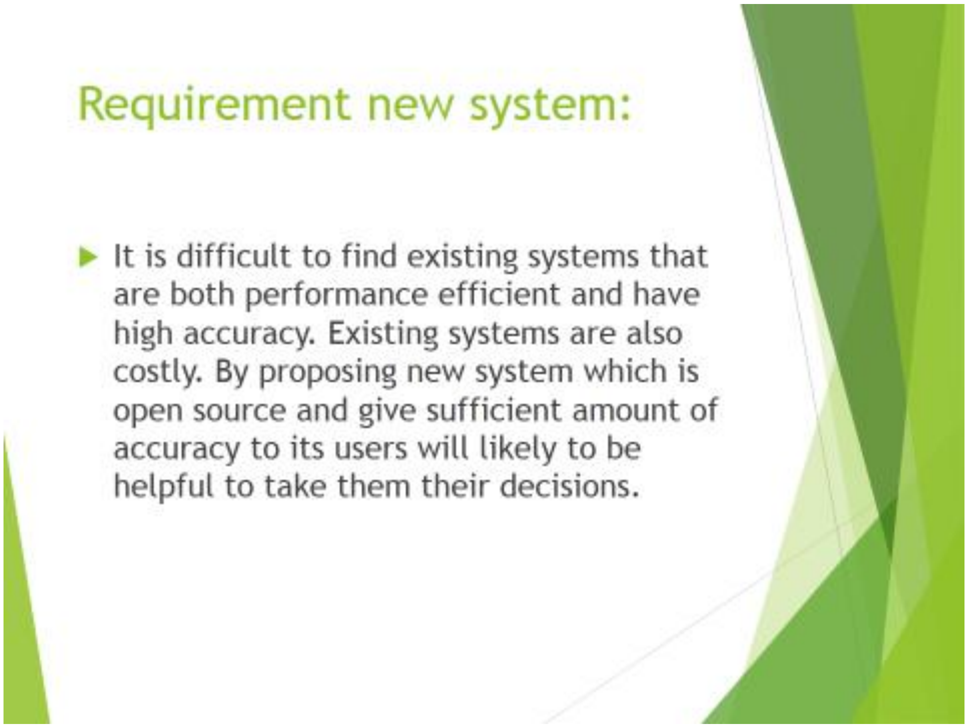
Scope :

- The objective of the system is to give an approximate idea of where the stock market might be headed. It does not give a long-term forecasting of a stock value. There are way too many reasons to acknowledge for the long-term output of a current stock.
- Predicted and analysed data is also useful to Companies themselves. Company and industry can use it to stretch their limitations and enhance their stock value. It can be very useful to even researchers, stock brokers, market makers, government and general people.



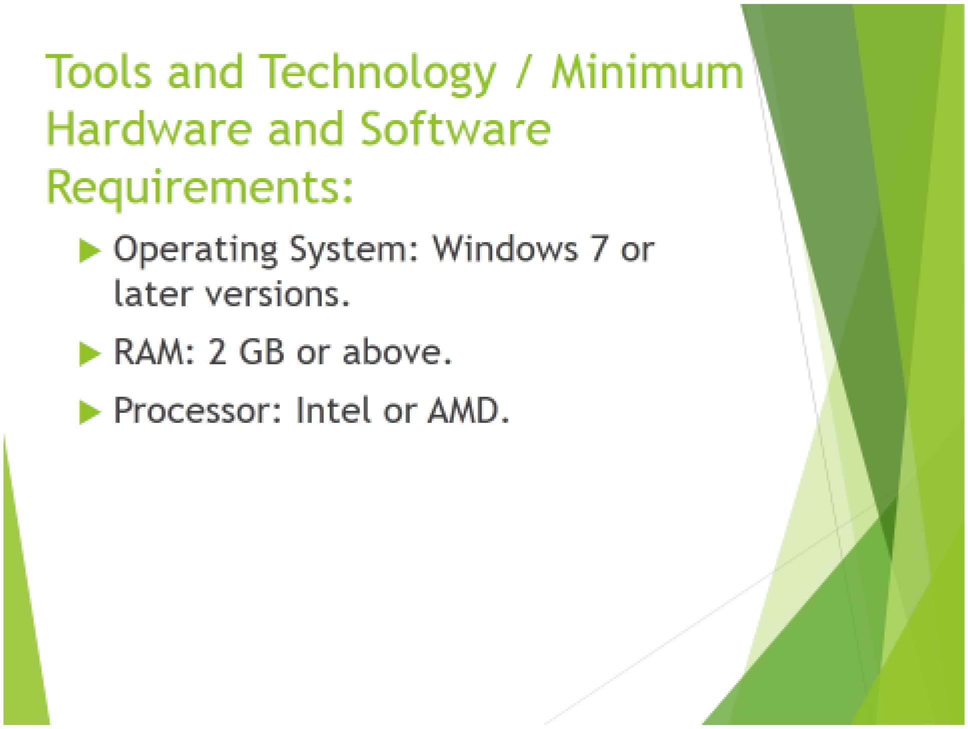
Current system study :

- Currently the systems use RNN, CNN, SVM to predict stock prices, with the advent in computational power of the computers. Computers with mediocre computational powers cannot perform heavy algorithms which involve complex neural networks.
- The main disadvantage of all current systems is that once you define the model of prediction, you cannot change it. It may be result into heavy loss in the accuracy of the system. So, every current system tends not to follow EMH.



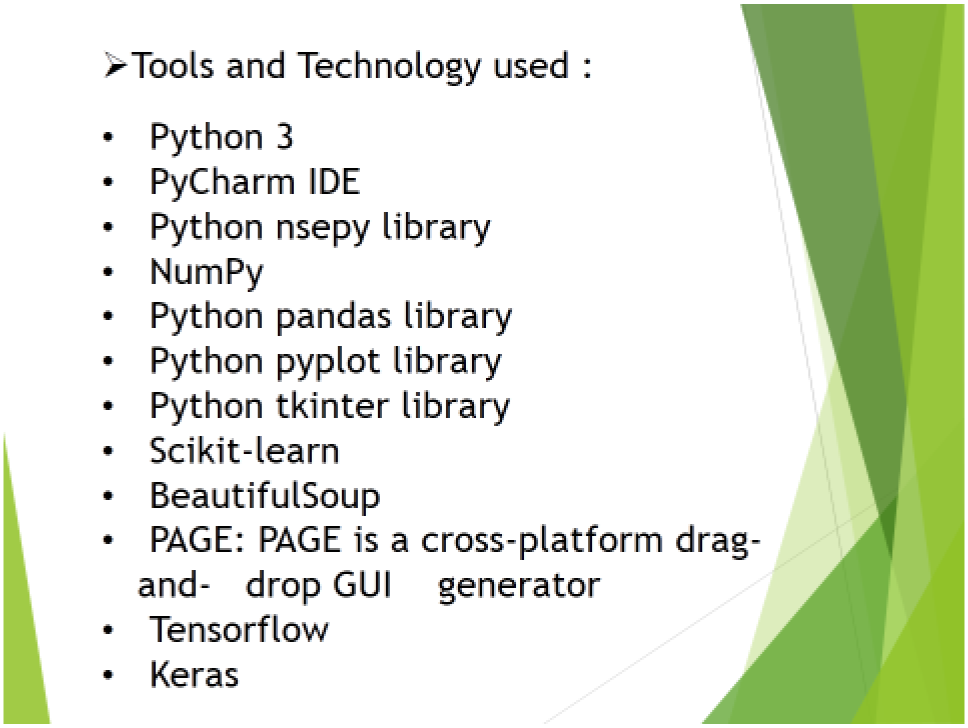
Requirement new system:

- It is difficult to find existing systems that are both performance efficient and have high accuracy. Existing systems are also costly. By proposing new system which is open source and give sufficient amount of accuracy to its users will likely to be helpful to take them their decisions.



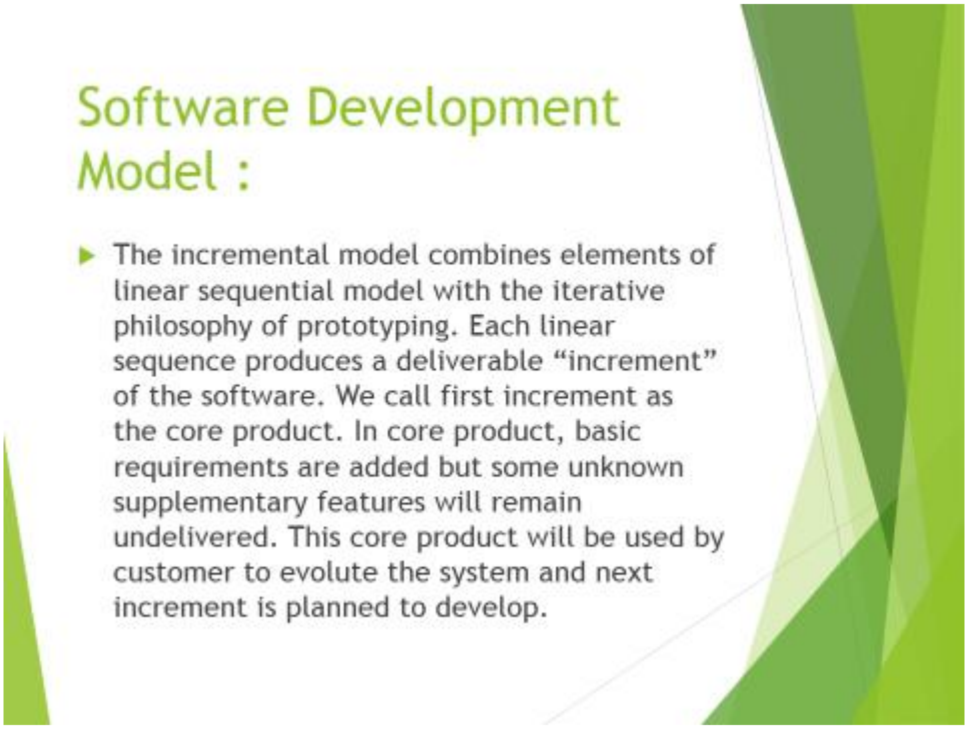
Tools and Technology / Minimum Hardware and Software Requirements:

- Operating System: Windows 7 or later versions.
- RAM: 2 GB or above.
- Processor: Intel or AMD.



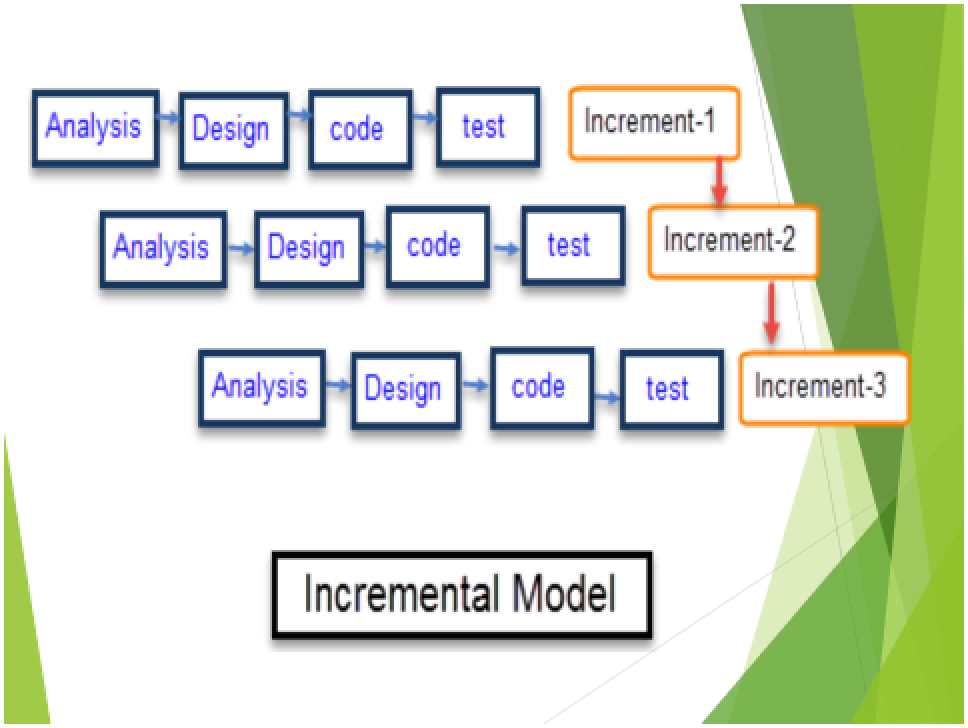
➢Tools and Technology used :

- Python 3
- PyCharm IDE
- Python nsepy library
- NumPy
- Python pandas library
- Python pyplot library
- Python tkinter library
- Scikit-learn
- BeautifulSoup
- PAGE: PAGE is a cross-platform drag-and- drop GUI generator
- Tensorflow
- Keras

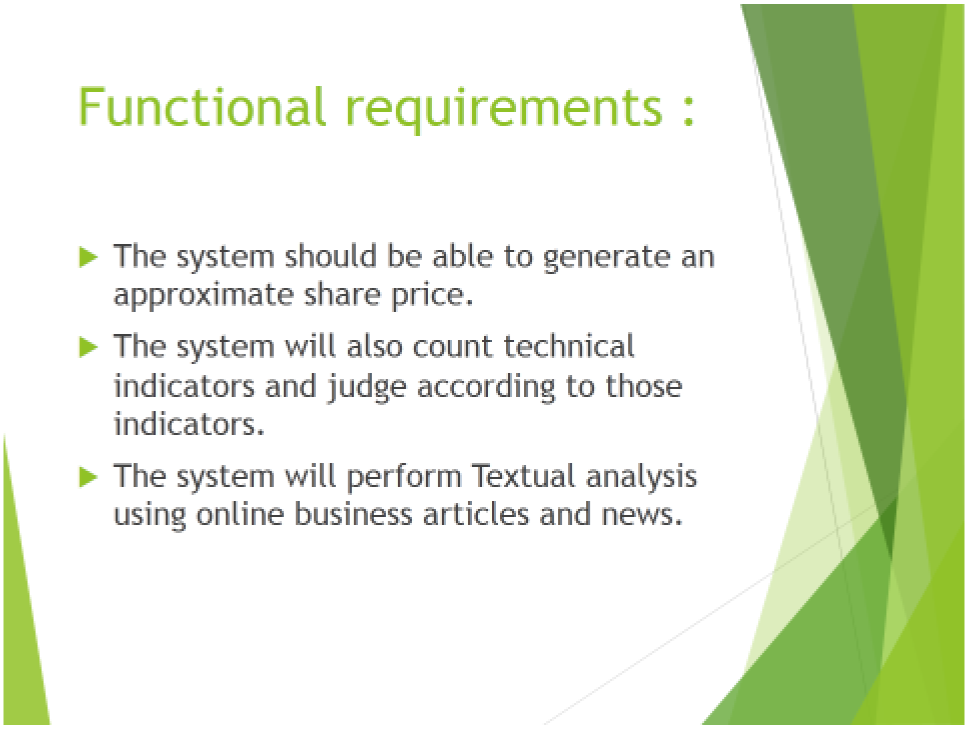


Software Development Model :

▶ The incremental model combines elements of linear sequential model with the iterative philosophy of prototyping. Each linear sequence produces a deliverable “increment” of the software. We call first increment as the core product. In core product, basic requirements are added but some unknown supplementary features will remain undelivered. This core product will be used by customer to evolute the system and next increment is planned to develop.

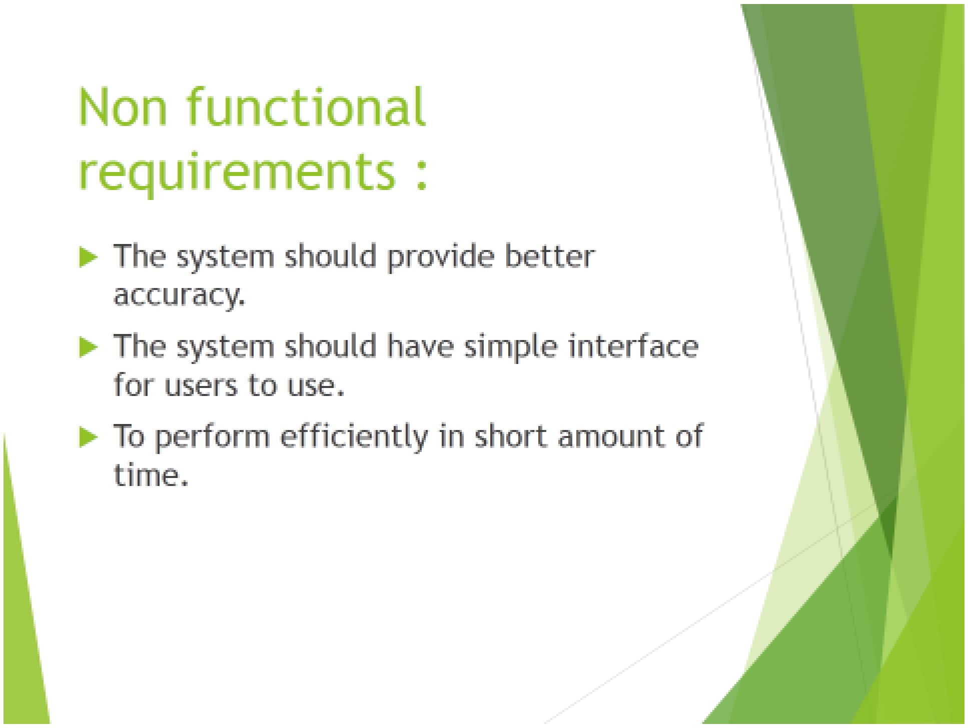


Incremental Model



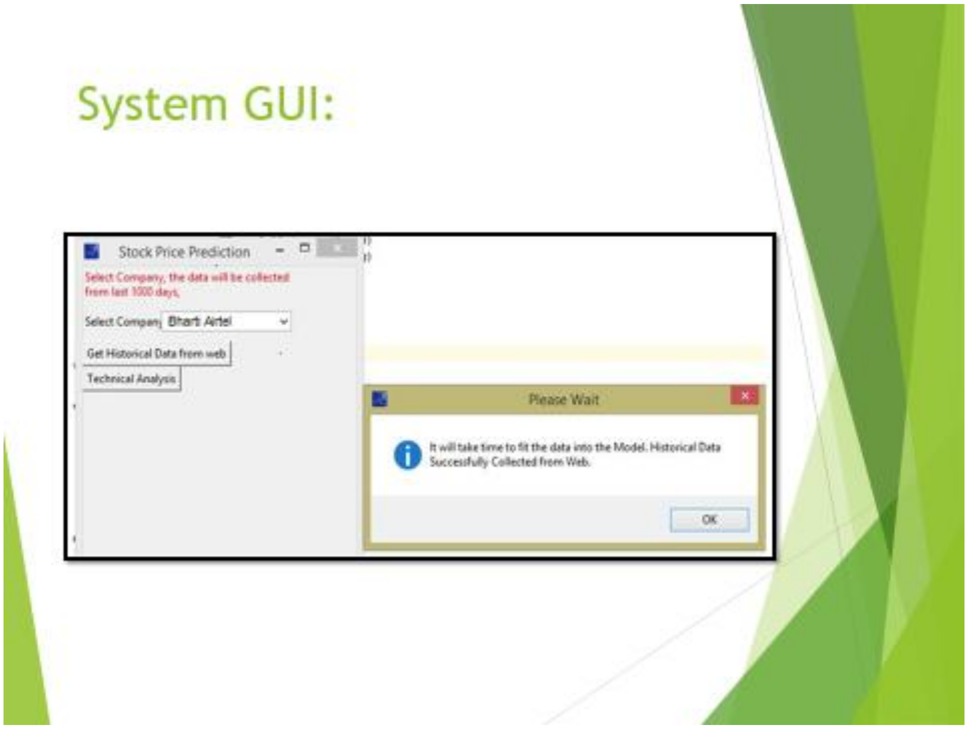
Functional requirements :

- The system should be able to generate an approximate share price.
- The system will also count technical indicators and judge according to those indicators.
- The system will perform Textual analysis using online business articles and news.



Non functional requirements :

- The system should provide better accuracy.
- The system should have simple interface for users to use.
- To perform efficiently in short amount of time.



System GUI:

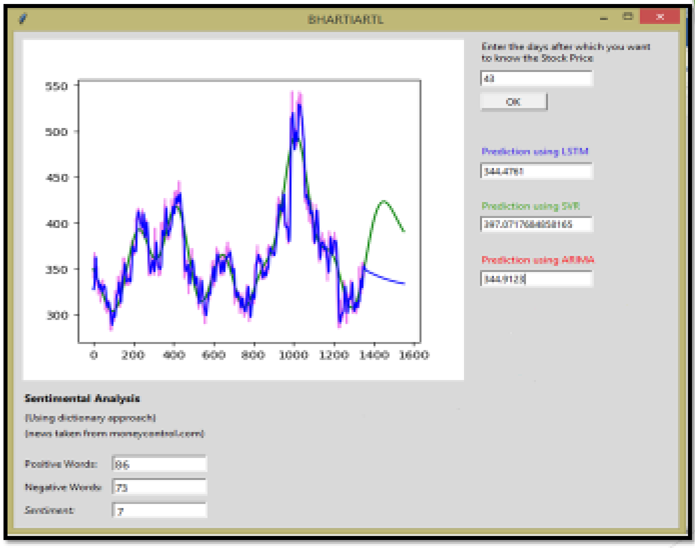
Stock Price Prediction

Select Company, the data will be collected from last 1000 days,

Select Compan: Bharti Airtel

Please Wait

It will take time to fit the data into the Model. Historical Data Successfully Collected from Web.



Enter the days after which you want to know the Stock Price

43

Prediction using LSTM

344.4781

Prediction using SVR

397.0717684058165

Prediction using ARIMA

344.9123

Sentimental Analysis

(Using dictionary approach)

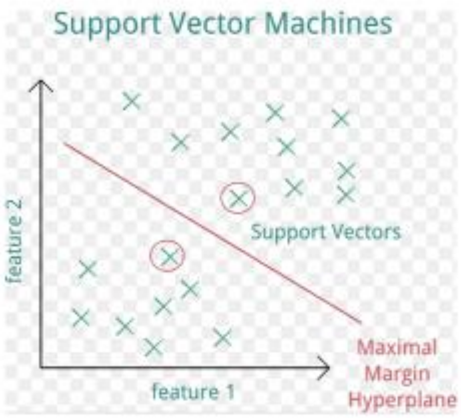
(news taken from moneycontrol.com)

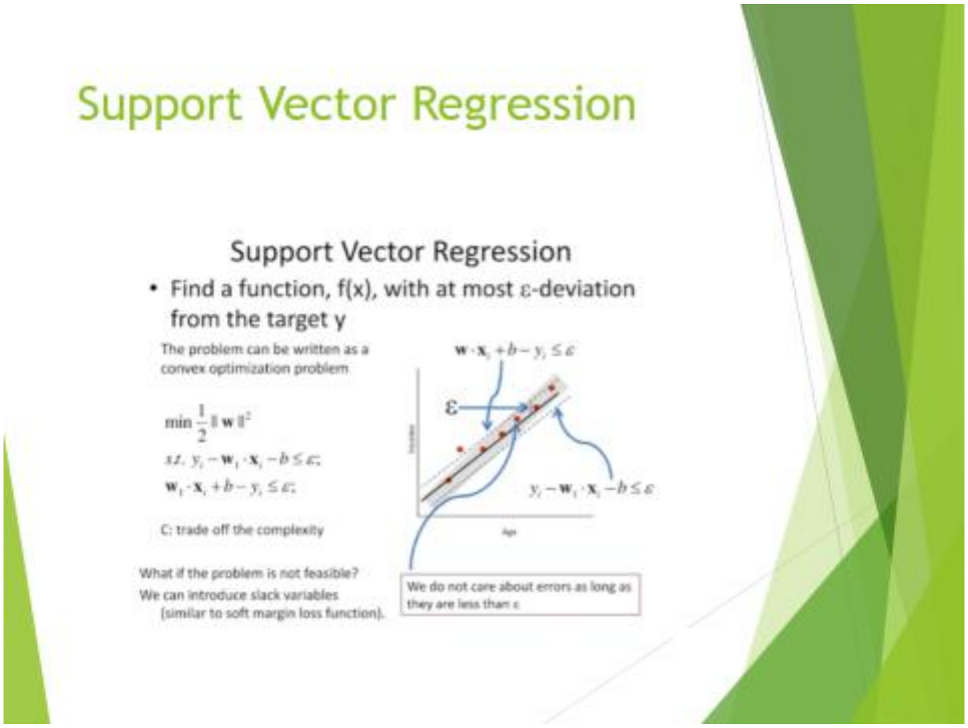
Positive Words: 86

Negative Words: 73

Sentiment: 7

Support Vector Machine





Support Vector Regression

Support Vector Regression

- Find a function, f(x), with at most ε-deviation from the target y

The problem can be written as a convex optimization problem

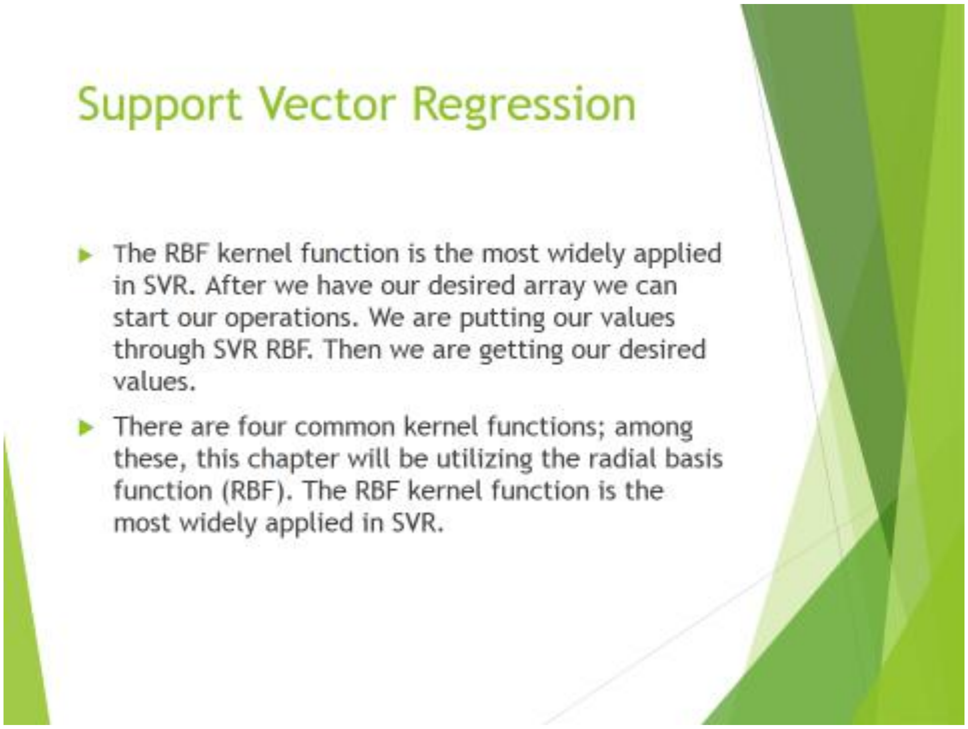
$$\min \frac{1}{2} \| \mathbf{w} \|^2$$
$$s.t.\ y_i - \mathbf{w}_1 \cdot \mathbf{x}_i - b \le \varepsilon;$$
$$\mathbf{w}_1 \cdot \mathbf{x}_i + b - y_i \le \varepsilon;$$

C: trade off the complexity

What if the problem is not feasible?
We can introduce slack variables
(similar to soft margin loss function).

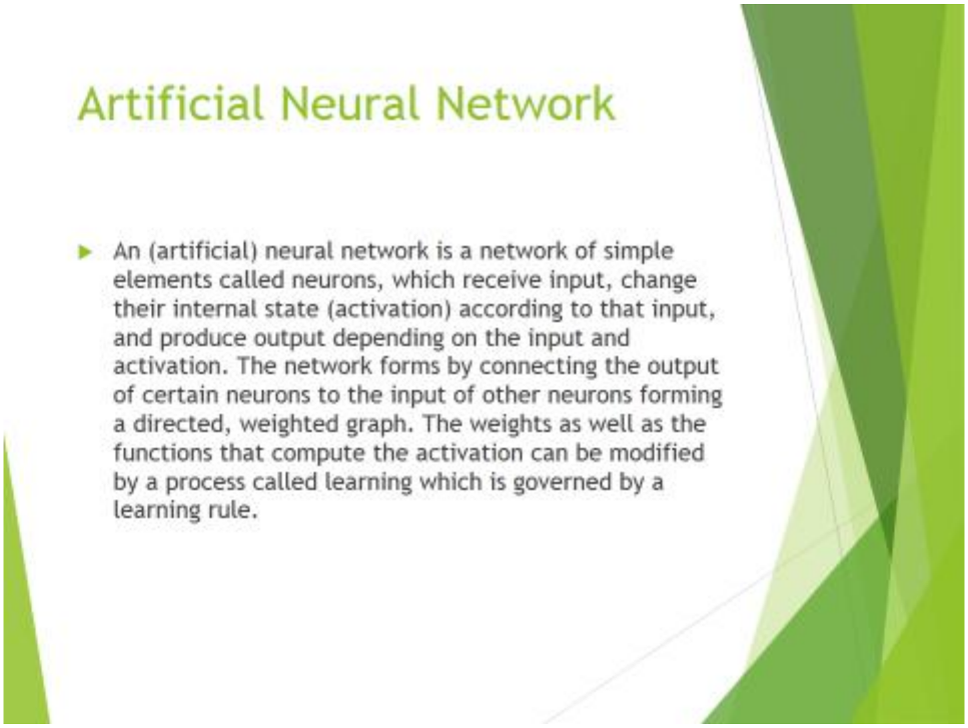
We do not care about errors as long as they are less than ε

$\mathbf{w} \cdot \mathbf{x}_i + b - y_i \le \varepsilon$
ε
$y_i - \mathbf{w}_1 \cdot \mathbf{x}_i - b \le \varepsilon$
Age



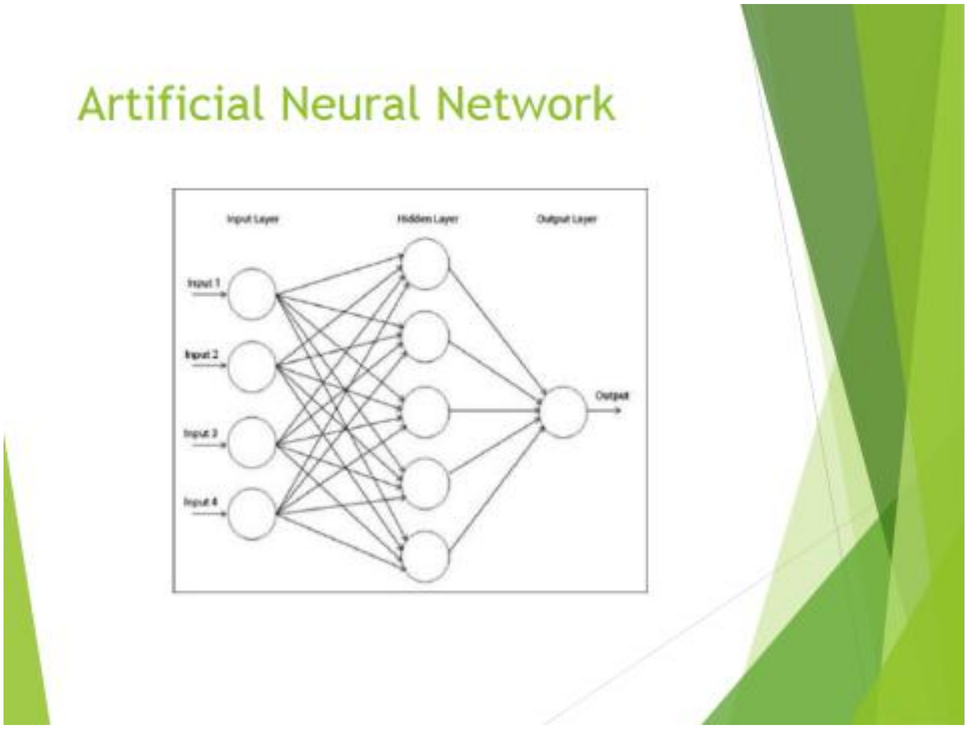
Support Vector Regression

- The RBF kernel function is the most widely applied in SVR. After we have our desired array we can start our operations. We are putting our values through SVR RBF. Then we are getting our desired values.
- There are four common kernel functions; among these, this chapter will be utilizing the radial basis function (RBF). The RBF kernel function is the most widely applied in SVR.

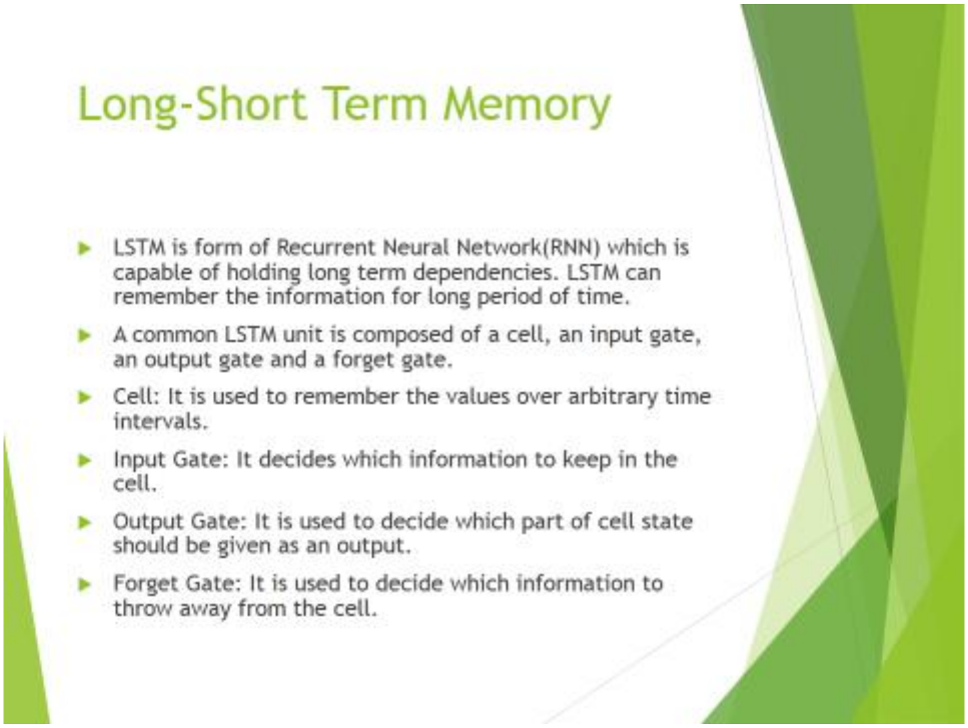


Artificial Neural Network

- An (artificial) neural network is a network of simple elements called neurons, which receive input, change their internal state (activation) according to that input, and produce output depending on the input and activation. The network forms by connecting the output of certain neurons to the input of other neurons forming a directed, weighted graph. The weights as well as the functions that compute the activation can be modified by a process called learning which is governed by a learning rule.

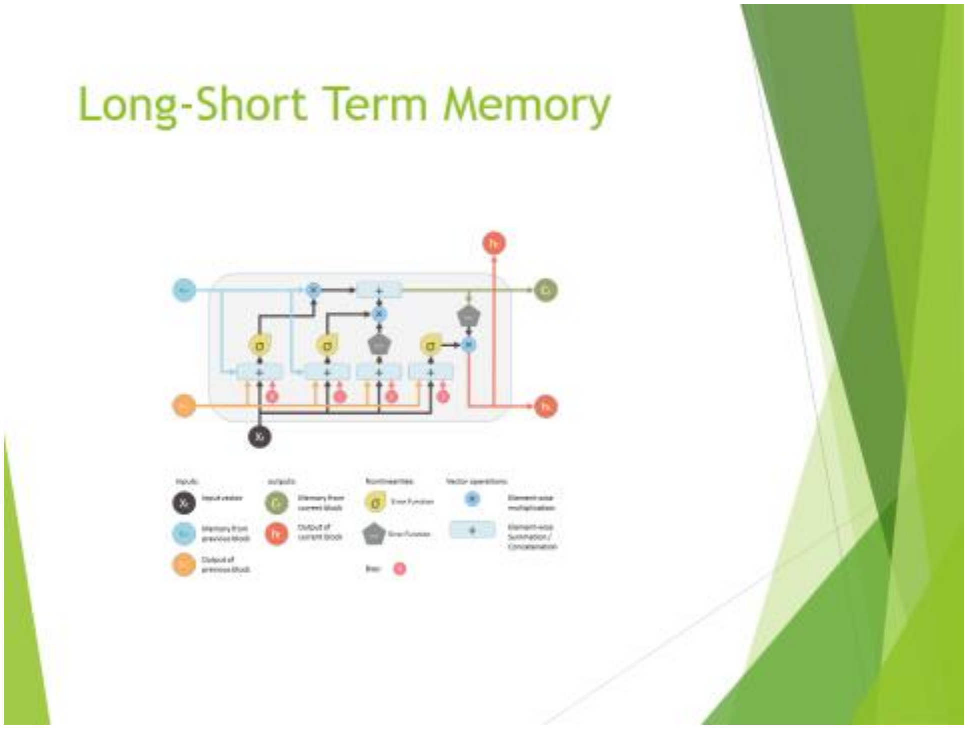


Artificial Neural Network



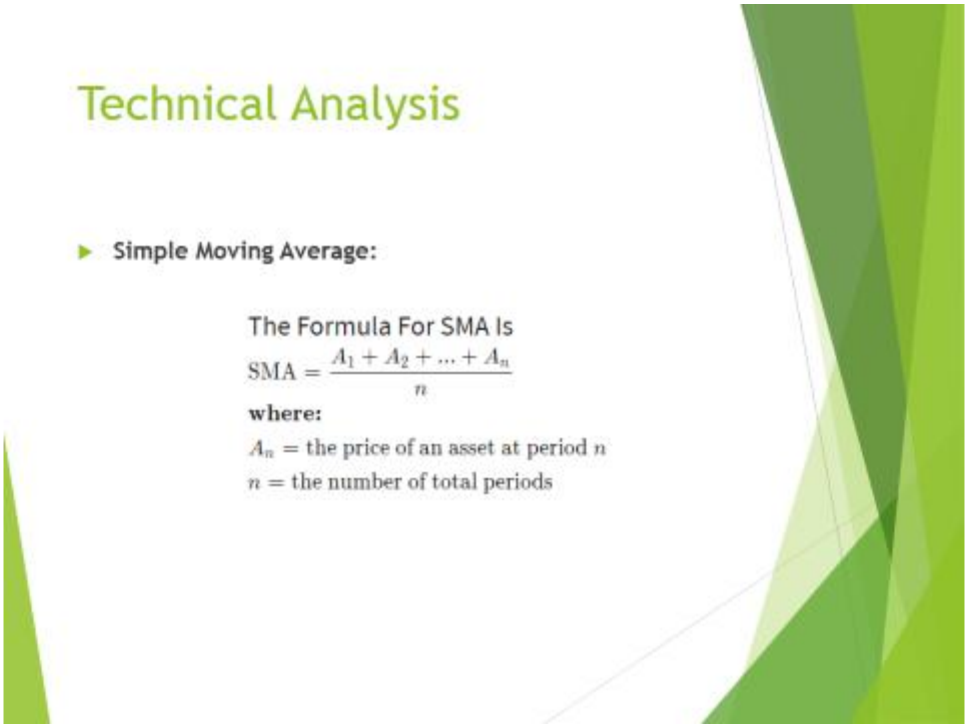
Long-Short Term Memory

- LSTM is form of Recurrent Neural Network(RNN) which is capable of holding long term dependencies. LSTM can remember the information for long period of time.
- A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate.
- Cell: It is used to remember the values over arbitrary time intervals.
- Input Gate: It decides which information to keep in the cell.
- Output Gate: It is used to decide which part of cell state should be given as an output.
- Forget Gate: It is used to decide which information to throw away from the cell.



Long-Short Term Memory

σ
σ
σ
Inputs:
Input vector
Memory from previous block
Output of previous block
outputs:
Memory from current block
Output of current block
Nonlinearities:
σ
Bias:
Vector operations:
Element-wise multiplication
Element-wise Summation / Concatenation



Technical Analysis

- Simple Moving Average:

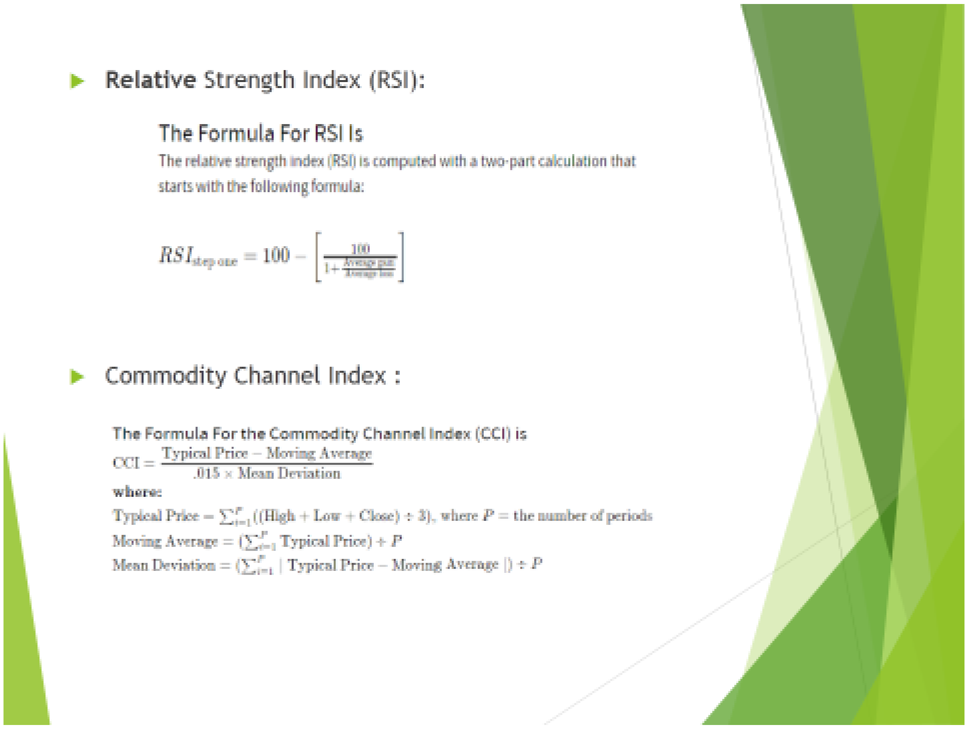
The Formula For SMA Is

$$\text{SMA} = \frac{A_1 + A_2 + \dots + A_n}{n}$$

where:

A_n = the price of an asset at period n

n = the number of total periods



- **Relative** Strength Index (RSI):

The Formula For RSI Is

The relative strength index (RSI) is computed with a two-part calculation that starts with the following formula:

$$RSI_{\text{step one}} = 100 - \left[\frac{100}{1 + \frac{\text{Average gain}}{\text{Average loss}}} \right]$$

- Commodity Channel Index :

The Formula For the Commodity Channel Index (CCI) is

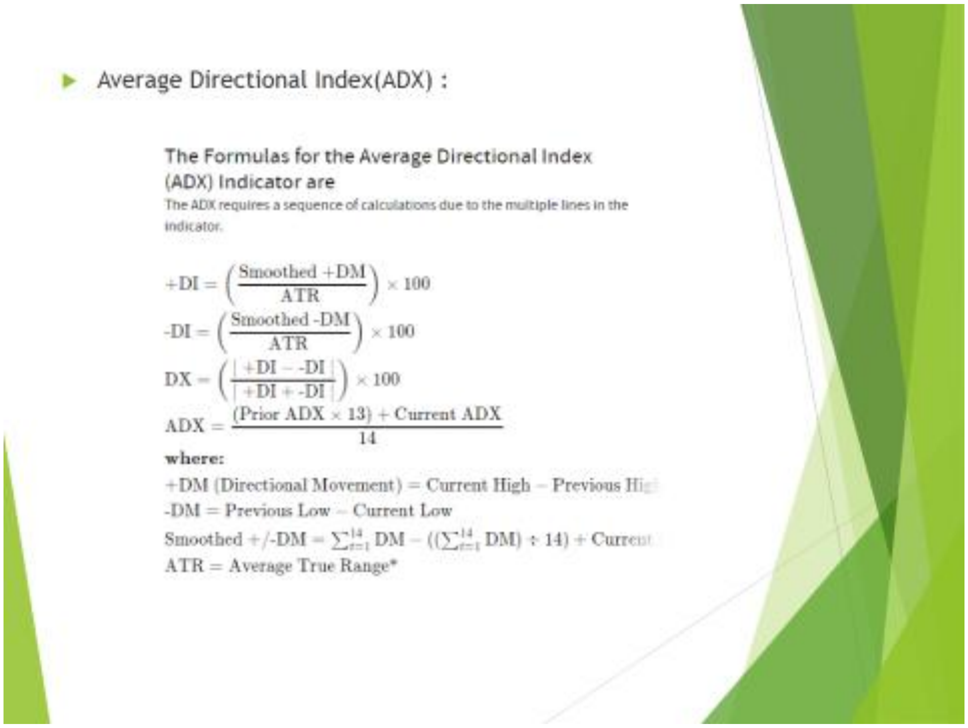
$$\text{CCI} = \frac{\text{Typical Price} - \text{Moving Average}}{.015 \times \text{Mean Deviation}}$$

where:

Typical Price = $\sum_{i=1}^{P}((\text{High} + \text{Low} + \text{Close}) \div 3)$, where P = the number of periods

Moving Average = $(\sum_{i=1}^{P} \text{Typical Price}) \div P$

Mean Deviation = $(\sum_{i=1}^{P} \mid \text{Typical Price} - \text{Moving Average} \mid) \div P$



- Average Directional Index(ADX) :

The Formulas for the Average Directional Index (ADX) Indicator are

The ADX requires a sequence of calculations due to the multiple lines in the indicator.

$$+\mathrm{DI} = \left(\frac{\text{Smoothed } +\mathrm{DM}}{\mathrm{ATR}}\right) \times 100$$

$$-\mathrm{DI} = \left(\frac{\text{Smoothed } -\mathrm{DM}}{\mathrm{ATR}}\right) \times 100$$

$$\mathrm{DX} = \left(\frac{\left| +\mathrm{DI} - -\mathrm{DI} \right|}{\left| +\mathrm{DI} + -\mathrm{DI} \right|}\right) \times 100$$

$$\mathrm{ADX} = \frac{(\text{Prior ADX} \times 13) + \text{Current ADX}}{14}$$

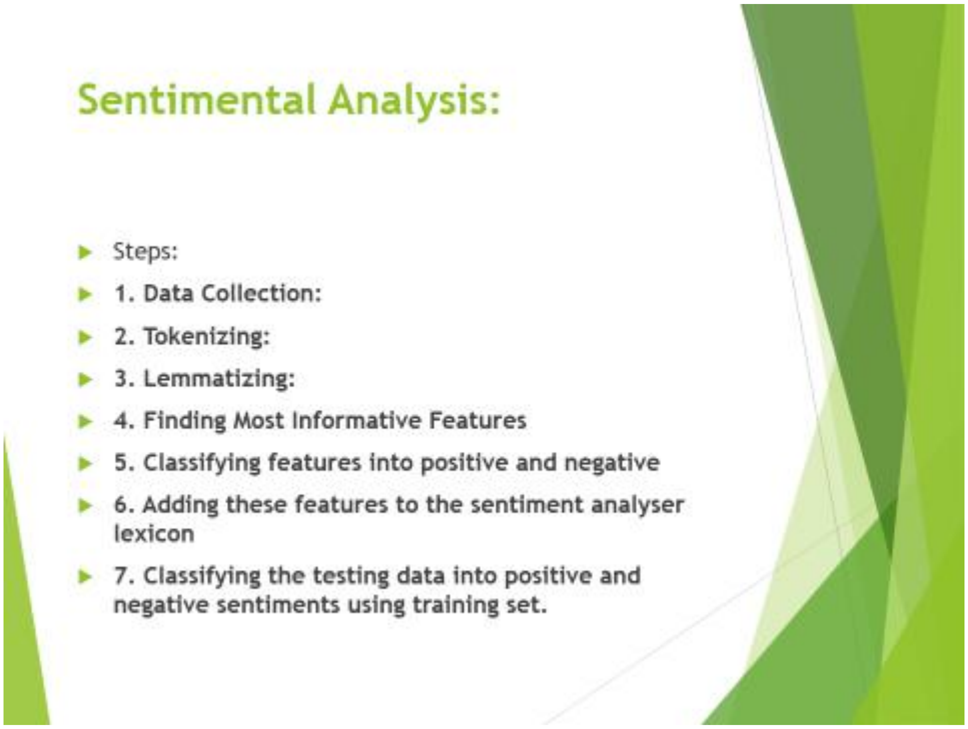
where:

$+\mathrm{DM}$ (Directional Movement) = Current High − Previous Hi

$-\mathrm{DM}$ = Previous Low − Current Low

Smoothed $+/-\mathrm{DM} = \sum_{t=1}^{14} \mathrm{DM} - \left(\left(\sum_{t=1}^{14} \mathrm{DM}\right) \div 14\right) +$ Curren

ATR = Average True Range*



Sentimental Analysis:

- Steps:
- **1. Data Collection:**
- **2. Tokenizing:**
- **3. Lemmatizing:**
- **4. Finding Most Informative Features**
- **5. Classifying features into positive and negative**
- **6. Adding these features to the sentiment analyser lexicon**
- **7. Classifying the testing data into positive and negative sentiments using training set.**