VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI – 590018, Karnataka

INTERNSHIP REPORT

ON

"Stockport | Prediction Sentiment Analysis"

Submitted in partial fulfilment for the award of degree(18CSI85)

BACHELOR OF ENGINEERING IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by:
AYESHA SIDIQA A MULLA

2GJ19CS001

Conducted at
VARCONS TECHNOLOGY PVT LTD

GIRIJABAI SAIL INSTITUTE OF TECHNOLOGY,KARWAR

(Utkagali, Adjacent To Karnataka-Goa Border Check Post, NH 66, Majali, Karwar, Uttar Kannada, Karnataka-581345)

GIRIJABAI SAIL INSTITUTE OF TECHNOLOGY

Department Of Computer science and engineering

CERTIFICATE

This is to certify that the Internship titled **"Stockport | Predictive Sentiment Analysis"** carried out by **Ms.Ayesha Sidiqa A Mulla ,** a bonafide student of **Girijabai Sail Institute of Technology**, in partial fulfillment for the award of **Bachelor of Engineering**, in **Computer Science and Engineering** under Visvesvaraya Technological University, Belagavi, during the year 2022-2023. It is certified that all corrections/suggestions indicated have been incorporated in the report.

The project report has been approved as it satisfies the academic requirements in respect of Internship prescribed for the course Internship / Professional Practice (18CSI85)

Signature of Guide **Signature of HOD External Viva:** **Signature of Principal**

Name of the Examiner Signature with Date

1) _____

_____

2) _____

_____

D E C L A R A T I O N

I, **Ayesha Sidiqa A Mulla** final year student of Computer Science and Engineering, Girijabai Sail Institute Of Technology, declare that the Internship

has been successfully completed, in **Varcons Technology Pvt Ltd.** This report is submitted in partial fulfillment of the requirements for award of Bachelor Degree in Branch name, during the academic year 2022-2023.

Date :23-09-2022 :

Place : KARWAR

-2GJ19CS001

-AYESHA SIDIQA A MULLA

OFFER LETTER

Internship Offer Letter

Varcons Technologies Pvt Ltd

Date: **23rd August, 2022**

Name: **Ayesha Sidiqa A Mulla**
USN: **2gj19cs001**

Dear Student,

We would like to congratulate you on being selected for the **Machine Learning With Python(Research Based)** Internship position with **Varcons Technologies Pvt Ltd**, effective Start Date **23rd August, 2022**, All of us are excited about this opportunity provided to you!

This internship is viewed as being an educational opportunity for you, rather than a part-time job. As such, your internship will include training/orientation and focus primarily on learning and developing new skills and gaining a deeper understanding of concepts of **Machine Learning With Python(Research Based)** through hands-on application of the knowledge you learn while you train with the senior developers. You will be bound to follow the rules and regulations of the company during your internship duration.

Again, congratulations and we look forward to working with you!

Sincerely,

Spoorthi H C
Director
Varcons Technologies Pvt Ltd
213, 2st Floor,
18 M G Road, Ulsoor,
Bangalore-560001

A C K N O W L E D G E M E N T

This Internship is a result of accumulated guidance, direction and support of several important persons. We take this opportunity to express our gratitude to all who have helped us to complete the Internship.

We express our sincere thanks to our Principal, for providing us adequate facilities to undertake this Internship.

We would like to thank our Head of Dept – branch code, for providing us an opportunity to carry out Internship and for his valuable guidance and support.

We would like to thank our (Lab assistant name) Software Services for guiding us during the period of internship.

We express our deep and profound gratitude to our guide, Guide name, Assistant/Associate Prof, for her keen interest and encouragement at every step in completing the Internship.

We would like to thank all the faculty members of our department for the support extended during the course of Internship.

We would like to thank the non-teaching members of our dept, for helping us during the Internship.

Last but not the least, we would like to thank our parents and friends without whose constant help, the completion of Internship would have not been possible.

AYESHA SIDIQA A MULLA
2GJ19CS001

ABSTRACT

The application addressed in this paper studies whether Twitter feeds, expressing public opinion concerning companies and their products, are a suitable data source for forecasting the movements in stock closing prices. We use the term predictive sentiment analysis to denote the approach in which sentiment analysis is used to predict the changes in the

phenomenon of interest. In this paper, positive sentiment probability is proposed as a new indicator to be used in predictive sentiment analysis in finance. By using the Granger causality test we show that sentiment polarity (positive and negative sentiment) can indicate stock price movements a few days in advance. Finally, we adapted the Support Vector Machine classification mechanism to categorize tweets into three sentiment categories (positive, negative and neutral), resulting in improved predictive power of the classifier in the stock market application.

Keywords: stock market, Twitter, predictive sentiment analysis, sentiment classification, positive sentiment probability, Granger causality.

Table of Contents

CHAPTER 1 COMPANY PROFILE

1. COMPANY PROFILE

A Brief History of Varcons Technologies

Varcons Technologies, was incorporated with a goal ”To provide high quality and optimal Technological Solutions to business requirements of our clients”. Every business is a different and has a unique business model and so are the technological requirements. They understand this and hence the solutions provided to these requirements are different as well. They focus on clients requirements and provide them with tailor made technological solutions. They also understand that Reach of their Product to its targeted market or the automation of the existing process into e-client and simple process are the key features that our clients desire from Technological Solution they are looking for and these are the features that we focus on while designing the solutions for their clients.

Sarvamoola Software Services. is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Sarvamoola Software Services. specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements.

Varcons Technologies, strive to be the front runner in creativity and innovation in software development through their well-researched expertise and establish it as an out of the box software development company in Bangalore, India. As a software development company, they translate this software development expertise into value for their customers through their professional solutions.

They understand that the best desired output can be achieved only by understanding the clients demand better. Varcons Technologies work with their clients and help them to defiine their exact solution requirement. Sometimes even they wonder that they have completely redefined their solution or new application requirement during the brainstorming session, and here they position themselves as an IT solutions consulting group comprising of high caliber consultants.

They believe that Technology when used properly can help any business to scale and achieve new heights of success. It helps Improve its efficiency, profitability, reliability; to put it in one sentence ” Technology helps you to Delight your Customers” and that is what we want to achieve.

CHAPTER 2 ABOUT THE COMPANY

2. ABOUT THE COMPANY

Varcons Technologies is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Varcons Technologies specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements. The organization where they have a right mix of professionals as a stakeholders to help us serve our clients with best of our capability and with at par industry standards. They have young, enthusiastic, passionate and creative Professionals to develop technological innovations in the field of Mobile technologies, Web applications as well as Business and

Enterprise solution. Motto of our organization is to “Collaborate with our clients to provide them with best Technological solution hence creating Good Present and Better Future for our client which will bring a cascading a positive effect in their business shape as well”. Providing a Complete suite of technical solutions is not just our tag line, it is Our Vision for Our Clients and for Us, We strive hard to achieve it.

Products of Varcons Technologies.

Android Apps

It is the process by which new applications are created for devices running the Android operating system. Applications are usually developed in Java (and/or Kotlin; or other such option) programming language using the Android software development kit (SDK), but other development environments are also available, some such as Kotlin support the exact same Android APIs (and bytecode), while others such as Go have restricted API access.

The Android software development kit includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and zutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but softwaredevelopment is possible by using specialized Android applications.

Web Application

It is a client–server computer program in which the client (including the user interface and client- side logic) runs in a web browser. Common web applications include web mail, online

retail sales, online auctions, wikis, instant messaging services and many other functions. web applications use web documents written in a standard format such as HTML and JavaScript,which are supported by a variety of web browsers. Web applications can be considered as a specifific variant of client–server software where the client software is downloaded to the client machine when visiting the relevant web page, using standard procedures such as HTTP. The Client web software updates may happen each time the web page is visited. During the session, the web browser interprets and displays the pages, and acts as the universal client for any web application. The use of web application frameworks can often reduce the number of errors in a program, both by making the code simpler, and by allowing one team to concentrate on the framework while another focuses on a specifified use case. In applications which are exposed to constant hacking attempts on the Internet, security-related problems can be caused by errors in the program.

Frameworks can also promote the use of best practices such as GET after POST. There are some who view a web application as a two-tier architecture. This can be a “smart” client that performs all the work and queries a “dumb” server, or a “dumb” client that relies on a “smart” server. The client would handle the presentation tier, the server would have the database (storage tier), and the business logic (application tier) would be on one of them or on both. While this increases the scalability of the applications and separates the display and the database, it still doesn‟t allow for true specialization of layers, so most applications will outgrow this model. An emerging strategy for application software companies is to provide web access to software previously distributed as local applications. Depending on the type of application, it may require the development of an entirely different browser-based interface, or merely adapting an existing application to use different presentation technology. These programs allow the user to pay a monthly or yearly fee for use of a software application without having to install it on a local hard drive. A company which follows this strategy is known as an application service provider (ASP), and ASPs are currently receiving much attention in the software industry.

Security breaches on these kinds of applications are a major concern because it can involve both enterprise information and private customer data. Protecting these assets is an important part of any web application and there are some key operational areas that must be included in the development process. This includes processes for authentication, authorization, asset handling, input, and logging and auditing. Building security into the applications from the beginning can be more effective and less disruptive in the long run.

Web design

It is encompasses many different skills and disciplines in the production and maintenance of websites. The different areas of web design include web graphic design; interface design; authoring, including standardized code and proprietary software; user experience design; and search engine optimization. The term web design is normally used to describe the design process relating to the front-end (client side) design of a website including writing mark up.

Web design partially overlaps web engineering in the broader scope of web development. Web designers are expected to have an awareness of usability and if their role involves creating mark up then they are also expected to be up to date with web accessibility guidelines. Web design partially overlaps web engineering in the broader scope of web development.

Departments and services offered

Varcons Technologies plays an essential role as an institute, the level of education, development of student's skills are based on their trainers. If you do not have a good mentor then you may lag in many things from others and that is why we at Varcons Technologies gives you the facility of skilled employees so that you do not feel unsecured about the academics. Personality development and academic status are some of those things which lie on mentor's hands. If you are trained well then you can do well in your future and knowing its importance of Varcons Technologies always tries to give you the best.

They have a great team of skilled mentors who are always ready to direct their trainees in the best possible way they can and to ensure the skills of mentors we held many skill development programs as well so that each and every mentor can develop their own skills with the demands of the companies so that they can prepare a complete packaged trainee.

Services provided by Varcons Technologies.

• Core Java and Advanced Java

• Web services and development

• Dot Net Framework

• Python

• Selenium Testing

• Conference / Event Management Service

• Academic Project Guidance

• On The Job Training

• Software Training

CHAPTER 3 INTRODUCTION

3. INTRODUCTION

Introduction to ML

This project “Predictive Analysis of Stock Market using Sentiment Analysis of Twitter” aims to be a standout and accurate guidance for all kinds of investors in the country. The stock market is highly affected by political situation within a country. This is the reason the proposed methodology consists of sentiment analysis to analyze political situation within a country. Experienced investors know the rise and fall of stock market, their investment largely depends on their past experience and they get support from stock market to invest or withdraw their investments from it. Inexperienced or common investors are not aware of such techniques. By making my proposed system intelligent with sentiment analysis of tweets, we aim to provide a helpful platform for all kind of investors. The project makes use of cutting-edge 21st century technology to ensure that whatever it claims is backed-up by correct information and error-free processing. To name them, Python and Machine Learning technologies such as Scikit-learn, Pandas and NLTK, TEXTBLOB for sentiment analysis of twitter have been used to ensure good-handling and processing of data. The major modules of the project include extracting tweets,

preprocessing of tweets, sentiment analysis of tweets and classification of tweets as positive, negative and neutral using Machine Learning algorithms Naïve Bayes and SVM**.**

Motivation

Stock Market prediction is one of the difficult tasks to accomplish. This has been a topic of several researches (Shams and Muhammed 2005) (Deng et al. 2017) and researchers have tried to predict it by using Machine learning algorithms such as Recurrent Neural Networks (RNN), Regression Algorithms, Time Series models, Long Short Term Memory (LSTM), etc. but they wouldn’t succeed in

building an efficient model. That is largely due to the fact that the stock market is highly volatile, and it is affected by several factors cannot be taken into consideration for quantification. One of those factors is “tweets” that impact stock market. Its effect on stock market is something that cannot be computed so easily. Also, thing has not really been achieved so far.

Problem Statement

Experienced as well as inexperienced investors have to face loss due to uncertain behavior of stock market. This uncertain behavior depends on financial situation of the company as well as political situation within a country.

CHAPTER 4
SYSTEM ANALYSIS

4. SYSTEM ANALYSIS

1. Existing System

➢The existing system, Uses knowledge base approach to classify the tweets into either positive, negative or neutral. But, employing this method results in less accuracy of the classification.

DISADVANTAGES OF EXISTING SYSTEM:

➢In Existing System, They have employed Lexicon based method to compute the sentiment of the data coming from twitter which resulted in lower accuracy rate.

Also, there is a lot of overhead while computing the sentiment of a sentence, Because for each word this method retrieves the sentiment from a predefined word dictionary(Generally Senti-word)

2. Proposed System

- In the proposed system, we t1Y to analyze the sentiment of the twitter posts about electronic products like mobiles, laptops etc using Data Mining approach.
- By doing sentiment analysis in a specific domain, it is possible to identify the effect of domain information in sentiment classification.

In proposed system we are doing a comparative study on finding the sentiment using two different algorithms they are Naïve Baye's Method and Support Vector Machine(SVM)

ADVANTAGES OF PROPOSED SYSTEM:

- In proposed system we have used Data Mining Techniques which resulted in increasing the accuracy rate for finding the sentiment of data.
- Because of absence of the predefined datasets to find out the sentiment of each word. So, as a result the overhead on the algorithms has been reduced drastically, which directly resulted in the increase of

the efficiency

3. Objective of the System

➢ In this report we intend to use social media platform for collecting the twitter feeds and conduct sentiment analysis of the feeds for predicting accurately the stock market trend.

- Identifying and comparing the machine learning algorithms best suited for stock market prediction.

- Studying the processing and implementation of human feeds from social media based on the stocks. Internship report 2021-2022 29

- Performing Sentiment analysis on social media feeds using natural language processing to understand the trends in stock market

CHAPTER 5 REQUIREMENT ANALYSIS

5. REQUIREMENT ANALYSIS

Hardware Requirement Specification

OS	Windows 7, Windows 8 or Windows 10
RAM	4GB
Free Disc Space	1GB
PC	Laptop OR Desktop Computer with internet connection

Software Requirement Specification

➢ Backend Development

The entire backend is implemented using programming language “Python”. The version which is use in implementation is Python 3.6.

PYTHON Libraries

➢ Pandas:

It is an open source Python library that helps programmers a lot in manipulating datastructures. It also provides tools for data analysis in Python

. ➢ Re:

This is a module provided by Python for supporting regular expressions. It is used to find out a string or set of strings that match the sequence of characters in the pattern of regular expression . ➢ Sklearn:

This library is used to get function “Count Vectorizer” . This function extracts bag of words from tweets.

➢ Numpy:

This library in Python is used to manipulate arrays and to perform operations on Internship report 2021-2022 21 arrays. These operations include mathematical and many other operations.

➢ Nltk:

This is used for pre-processing of tweets. For example, it provides

functions to remove stop words from tweets and for performing stemming on tokenized words

. ➢ Matplotlib:

This is a Python library which is used to plot different types of graphs such as bar charts, pie graphs, scatter plot, etc. One can set plot size, figure size, labels and legends using this library in Python.

➢ Tensor flow:

This library is used in the project to import very famous Deep Learning Algorithm Long Short- Term Memory LSTM. This algorithm is used in training both types of data i.e. sentiment data of twitter and historical data of stocks.

Other Requirements:

Along with mentioned software and hardware requirements, other requirements include internet connection, web browsers e.g. Google Chrome, Mozilla Firefox, Internet Explorer, etc.

CHAPTER 6 DESIGN ANALYSIS

6. DESIGN & ANALYSIS

Stock market contains hundreds of companies and these vary on the basis of their size. The researcher chose one big company of US stock market namely Apple. Proposed system will give predictions of this company.

System Architecture:

The flow diagram of my project is as follows:

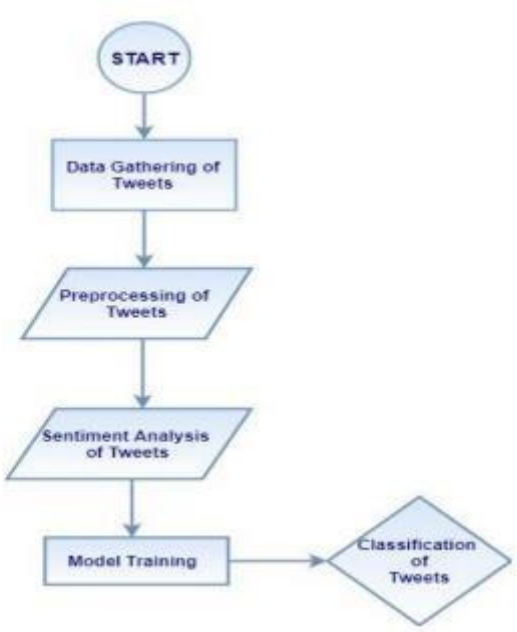


Figure : Flow Diagram of Proposed Solution

Natural Language Processing

Just like humans converse with each other, we can converse with computer as we do with Cortana, Google Assistant using Natural language processing. Being a part of the Artificial Intelligence the NLP has applications such as Sentiment analysis, Content modeling and categorization, Extraction etc. They can be used for spelling and error identification, text

classification, chatbots. Natural language can be used in python by using NLTK, TextBlob Library.

Data preprocessing

This step includes processing of data we're receiving and formatting it into our desired shape. The pre-processing of tweets includes tokenization, removing stop-words, lemmatization, and removal of all non-alphabetical words. In the pre-processing part of it all the tweets files were generated in a way that every day's tweet was under one file. Here comes the most interesting part of our project. It is also the second major module of the project that processing the data we're receiving and formatting it into our desired shape. When it came to news data, it really has to be processed as it was not in numerical form. And most importantly it did not include only English sentences. To ensure that the researcher could deal with the problem of not being typesafe, the researcher used default Unicode Transformation Format 8-bit Encoding (UTF-8). Then, the news was parsed, lowercased, its punctuation removed and lemmatized to ensure cleanliness and non-brevity. So, the researcher used NLTK library for this purpose of removing irrelevant sentences and words which were of no use to us. This includes removal of the stop words, removal of punctuations, and any other operator or word then the alphabets. Also, we did lemmatization of the words using the same library.

Lemmatization was an important step as this step reduces text to its stem and then checks whether this stem is present in the English dictionary or not. This is a very resource intensive step, but we had to do it to make our news cleaner and then use it for net effect calculation. Before lemmatization, the researcher was performing stemming on the news which proved to be bad choice. It works very similarly to lemmatization by converting a word to its stem. Though it was resource friendly, it didn't used to check whether the reduced stem was in the English dictionary or not. This was a bad choice because it used to remove lots of words in the tweets that

were fundamental to the financial tweet's sentiment calculation effect. After lemmatization the

tweets were broken into the tokens by using the NLTK library. All the preprocessing steps are explained in detail:

• **Removing special characters, numbers and punctuation**

In this step, special characters such as @, #, numbers and punctuation such as !, “”, etc are removed from tweets. For example:
5G reportedly coming to premium iPhones in 2020, all models in 2021 http://dlvr.it/R6mGC3 - @TechCrunch #Apple #iPhone After removing special characters, numbers and punctuation from above sentence new sentence becomes: reportedly coming to premium iPhones in all models in http://dlvr.it/R6mGC3 Techcrunch Apple iPhone

• **Removing URLs**

In this step URLs from tweets are removed to clean data and for further processing of tweets. For example: reportedly coming to premium iPhones in all models in http://dlvr.it/R6mGC3 Techcrunch Apple iPhone Internship report 2021-2022 27 After removing URL above tweet becomes: reportedly coming to premium iPhones in all models in Techcrunch Apple iPhone

• **Converting all alphabets to lowercase**

This step performs further processing on tweets. In this step all upper case alphabets are converted to lowercase. This is illustrated by following example: reportedly coming to premium iPhones in all models in Techcrunch Apple iPhone After converting alphabets to lower case the above tweet becomes: reportedly coming to premium iphones in all models in techcrunch apple iphone

• **Removing stop words**

This step includes removal of stop words such as a, an, the, from, what, etc. It is shown by following example: reportedly coming to premium iphones in all models in techcrunch apple

iphone After removing stop words i.e. to, in from above tweet it becomes: reportedly coming premium iphones models techcrunch apple iphone

• **Tokenization**

Tokenization splits a line of text into smaller parts that are known as tokens. This step is further illustrated by following example: reportedly coming premium iphones all models techcrunch apple iphone After tokenization above line of text becomes: [‘reportedly’ , ‘coming’ , ‘premium’ , ‘iphones’ , ‘models’ , ‘techcrunch’ , ‘apple’ , ‘iphone’]

• **Stemming**

Different variants of a word are produced morphologically in stemming procedure. E.g. words computing, computed stems to word “compute”. This procedure is also illustrated by following example: [‘reportedly’ , ‘coming’ , ‘premium’ , ‘iphones’ , ‘models’ , ‘techcrunch’ , ‘apple’ , ‘iphone’] After stemming word “coming” becomes “come” [‘reportedly’ , ‘come’ , ‘premium’ , ‘iphones’ , ‘models’ , ‘techcrunch’ , ‘apple’ , ‘iphone’]

• **Lemmatization**

Lemmatization returns words to lemma which is dictionary form of word by doing analysis according to vocabulary and morphology of words. For example: [‘reportedly’ , ‘come’ , ‘premium’ , ‘iphones’ , ‘models’ , ‘techcrunch’ , ‘apple’ , ‘iphone’] After lemmatization it becomes: [‘reportedly’ , ‘come’ , ‘premium’ , ‘iphones’ , ‘model’ , ‘techcrunch’ , ‘apple’ , ‘iphone’]

• **Labeling of Tweets**

For labeling of tweets, I have used library “TextBlob”. TextBlob is used for processing of textual data and natural language processing which includes sentiment analysis. TextBlob performs

various preprocessing operations on tweets data which include tokenization, removing of stopwords. It has a sentiment classifier which takes the tokens as input and returns the polarity for each tweet from -1 to 1.

Classification of Tweets

The biggest challenge before classification was to label data because classification Internship report 2021-2022 29 algorithms take labeled data as input. So, the researcher did label of data using TextBlob library. Sentiment of each tweet was written against each tweet. This labeled was saved in a .csv file that has to be use in future for classification and prediction purposes. The researcher used two classification algorithms i.e. Naïve Bayes and Support Vector Machines (SVM) for classification of tweets. The researcher tried different approaches with these algorithms to achieve maximum accuracy. Classifiers take input in the form of numerical data. In order to make these algorithms work with textual data, data is first converted into numeric form. This can be done using different approaches e.g. Bag of Words, TF-IDF and word-count.

Naïve Bayes

It is a grouping method dependent on Bayes' Theorem with a presumption of independence among indicators. In basic terms, a Naive Bayes classifier expects that the nearness of a specific component in a class is irrelevant to the nearness of some other element. For instance, a natural product might be viewed as an apple on the off chance that it is red, round, and around 3 creeps in distance across. Regardless of whether these features rely upon one another or upon the presence of different features, these properties autonomously add to the likelihood that this natural product is an apple and that is the reason it is known as 'Naive'. It's called naive because it assumes that all of the predictors are independent from one another. Naive Bayes is mostly used for binary or multiclass classification. They provide us a way of calculating the probability of posterior. Naïve Bayes is based on Bayes Theorem. Bayes Theorem works on “conditional probability”. This

probability says that if something has already occurred then something will happen. The formula of conditional probability is as follows

$$: P(H|E) = \frac{P(E|H) * P(H)}{\mathrm{P(E)}}$$

Support Vector Machine (SVM)

SVM is a Machine Learning Algorithm that does analysis on textual data to classify it and regression analysis of numerical data. It works on labeled data as it is a supervised learning algorithm and it classifies data separately into separate classes. Applications of SVM include classification of text, classification of images, document classification, etc. SVM is defined by a hyperplane which separates different classes. It gives hyperplane as an output which differentiates different classes and puts each class into a separate category. This hyperplane lies in a two-dimensional space which separates the plane into two parts and each class lay on any one of the either side. For example, based on the height and weight of an individual each and every point or feature is mapped onto an n dimensional space and we are trying to classify them into two different classes by using a hyperplane. Most importantly what you need to understand here is that we are trying to build this hyperplane so that the two classes that are separated as wide as possible. SVM can only be used on data that is linearly separable (i.e. a hyper-plane can be drawn between the two groups). There are established ways to do it, they are called Kernels. By using a combination of these Kernels, and tweaking their parameters, you'll most likely achieve better results than making up your own way. The advantage of SVMs are that you can use them, in case of features, when compared, you can use very little data as in each of your data points has. SVM uses a set of functions of mathematics that is known as kernel. Kernel takes data in the form of input and then performs different transformations on data to get the required form of data. Different forms of kernel are linear, non-linear, radial basis function (RBF), etc. Classifiers take input in the form of numerical data. In order to make these algorithms work with

textual data, data is first converted into numeric form. This can be done using Internship report 2021-2022 31 different approaches e.g. **Bag of Words, TF-IDF, N-grams** and **Word-count.**

Bag of Words

For example, following are three tweets:

- Tweet1: Apple is going to launch some new features in MAC Operating System.
- Tweet2: Samsung has revealed its new mobile model, but apple has not yet. All unique words are gathered in a vocabulary in Bag of Words approach. For the above given example Bag of Words will be:
- Vocabulary= [Apple, is, going, to, launch, some, new, features, in, MAC, Operating, System, Samsung, has, revealed, its, mobile, model, but, not, yet] In the next step each tweet is converted to a feature vector. In feature vector, if a word is in vocabulary and it is also found in tweet then that word is assigned number "1" in feature vector and if word is in vocabulary but it is not present in tweet then number "0" is assigned to that word in feature vector. Feature vector for Tweet2 is:
- Feature Vector: [1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1]

TF-IDF

TF-IDF approach gives more weightage to rare (specific) terms and less weightage to uncommon and irrelevant terms.TF stands for "Term Frequency" and IDF stands for "Inverse Document Frequency". Term Frequency is calculated by following formula:

$$Term\ Frequency\ (TF) = \frac{Frequency\ of\ a\ word\ in\ the\ document}{Total\ Number\ of\ words\ in\ the\ document}$$

Inverse Document Frequency is calculated by following formula:

$$Inverse\ Document\ Frequency\ (IDF) = \log\left(\frac{(Total\ number\ of\ documents)}{(Total\ Number\ of\ documents\ containing\ the\ word)}\right)$$

N-Grams

When we use only one-word feature for our model, we are using unigrams or 1-grams as a feature. But when we use more than one words or word sequence of two or three we are actually improving the predictive power of classifier. For example, if a sentence is “Don’t like chocolate” this sentence has word “like” which contributes towards positive sentiment but if we take all three words then this sentence has negative sentiment overall.

Word-Counts

This approach says that how many times a word appears in a sentence contributes more towards the overall sentiment of sentence. If a sentence has more positive words, then sentence has positive sentiment. For example, if a sentence has words “good”, “top”, “highquality” occurring repeatedly in a sentence then that sentence has positive sentiment. This is how word- count increases the predictive power of classifier.

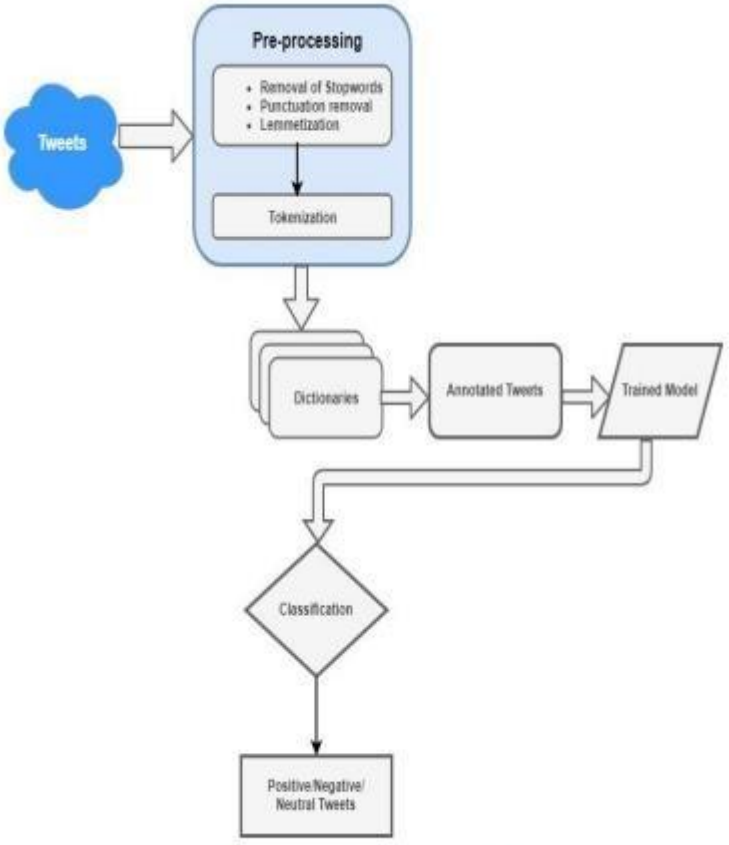


Figure High level diagram of sentiment analysis

Model Training using Historical data of Stocks and Sentiment Analysis of Twitter

After the preprocessing of tweets, sentiment analysis of tweets is done using TextBlob library of Python. TextBlob returns polarity and sentiment of tweets. Then different clusters are formed of tweets depending upon the polarity and sentiment value of tweets using kmeans algorithm. Historical data of stocks is downloaded from Yahoo Finance which contains the opening and closing price of stocks. Difference of opening and closing price of stock is calculated to give as input to Machine Learning models. Five features i.e. followers, polarity, sentiment confidence, clusters and difference (difference of opening and closing price of stock) are given as input to

SVR and LSTM. Two algorithms Support Vector Regression (SVR) and Long Short-Term Memory (LSTM) are used to give predictions for stock Market

Long Short-Term Memory (LSTM)

Long Short-Term Memory is a special kind of Recurrent Neural Network that is efficient enough to learn long-term events. RNN work similar to LSTM but they cannot work for long- term dependencies. For example, if we have a long sentence of English and we want to predict the last word of sentence then RNN would not be able to remember the long-term information and may not give right prediction. LSTM is designed to overcome that problem. E.g. If we have a sentence "I live in Canada and my nationality is Canadian" LSTM will successfully predict the last word of sentence by remembering the entire information but RNN can only store recent information. The activation function used with LSTM in my implementation is "ReLu (Rectified Linear Unit)". It gives zero output when input is equal to or less than zero. Otherwise it gives the same output as input. Long short-term memory networks or LSTM's are designed for Internship report 2021-2022 35 applications where the input is an ordered sequence where information from earlier in the sequence may be important. LSTM's are a type of recurrent network which are networks that reuse the output from a previous step as an input for the next step. Like all neural networks the node performs a calculation using the inputs and returns an output value in a recurrent Network. This output is then used along with the next element as the inputs for the next step and so on. In an LSTM the nodes are recurrent, but they also have an internal state the node uses an internal state as a working memory space which means information can be stored and retrieved over many time steps. The input value previous output and the internal state are all use in the node's calculations. The results of the calculations are used not only to provide an output value but also to update the state. Like any neural network LSTM nodes have parameters that determine how the inputs are used in the calculations. So, LSTM nodes are certainly more complicated than regular recurrent nodes, but this makes them better at learning the complex interdependencies in

sequences of data and ultimately, they're still just a node with a bunch of parameters and these parameters are learned during training just like with any other neural network.

Support Vector Regression (SVR)

Support Vector Regression (SVR) is very similar to Support Vector Machine (SVM) but it works on continuous data. The researcher chose SVR because he need to train it on numerical data of stocks and Twitter. Kernel of SVR is a function that changes the dimension of data from low to high. Hyperplane in SVR will help us in the prediction of continuous value. The data points which have minimum distance from boundary are known as support vectors. The motive of Support vector algorithm is without limiting or minimizing the size of violations on the margins between the two classes, they insist on involving or including the maximum data points Internship report 2021-2022 36 or instances between the margins while the margin the is minimized. The linear regression is performed by Support vector regression in high dimensional space. In support vector regression, in the training data set side of the hyperplane each of the instance or the data points represent their own dimension. In support vector regression, all the data point in the training data set of the hyperplane are evaluated and the higher dimensional sided all the test points have given the representation of ‘k’. The main aim of support vector regression or SVM is to limit the errors to min, where the errors are limited by maximizing the margins, coherency and without affecting the hyperplane. The advantage of Support vector regression over support vector machine that we are able to extend it to nonlinear data points where normal linear SVM cannot be applied.

CHAPTER 7

IMPLEMENTATION

7. IMPLEMENTATION

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification. It involves careful planning, investigation of the current system and it constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods a part from planning.

Two major tasks of preparing the implementation are education and training of the users and testing of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

TESTING

The testing phase is an important part of software development. It is the Information zed system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. Software testing is carried out in three steps:

1. The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether the objectives have been met. Errors are noted down and corrected immediately.
2. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.
3. The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole.

Script for extracting tweets from Twitter

In the following part, the researcher is only including supporting libraries that will scrap tweets from twitter

```
import time
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys  import
Keys  import csv
```

In the following lines, the researcher is opening Google Chrome browser through Python script and giving this browser URL of Twitter for a specific tag:

```
browser = webdriver.Chrome()

# write url here
# browser.get("https://twitter.com/hashtag/apple?f=news&vertical=news&src=rela")
browser.get("https://twitter.com/search?f=news&vertical=default&q=%23appl&src=typd")
```

In the following lines of code, Loop is running as many times as one wants to load the page of browser and get required number of tweets:

```
#Run this loop as many times as you want to load the
page  elm = browser.find_element_by_tag_name("html")
for x in range (3000):
 print(x)
 elm.send_keys(Keys.END)
time.sleep(10)
 elm.send_keys(Keys.HOME)  time.sleep(10)
```

In the following piece of code, the researcher is scrapping Tweets and Timestamp using CSS selector and saving tweets and time in “tweet_element” and “date_element” respectively.

```
# scrape the content by CSS SELECTOR
tweet_element = browser.find_elements(By.CSS_SELECTOR,'p[class="TweetTextSize jstweet-
text tweet-text"]')
date_element = browser.find_elements(By.CSS_SELECTOR,'a[class="tweet-timestamp
jspermalink js-nav js-tooltip"]')
```

Following chunk of code writes tweets along with timestamp in a csv file using function write row():

```
Script for Cleaning/ Pre-processing of Tweets This piece of code only imports libraries
and reads dataframe from .csv file. # writing data to csv file  with
open('Tweets_data_2.csv', 'w', newline='', encoding='utf-8') as csvfile:  autowriter =
csv.writer(csvfile)  autowriter.writerow(['Time','Tweets'])
for i in range(0,len(tweet_element)-1): tweet_text = tweet_element[i].text.encode("utf-8")
autowriter.writerow([date_element[i].get_attribute("title"), tweet_text])
```

Script for Cleaning/ Pre-processing of Tweets

This piece of code only imports libraries and reads dataframe from .csv file

```
import numpy as np  from
nltk.stem.porter import *
stemmer = PorterStemmer()
import warnings
warnings.filterwarnings("ignore",
category=DeprecationWarning)  from nltk.corpus import
stopwords  # df=pd.read_csv('data_dup.csv')
df=pd.read_csv('appletweets.csv')
df=df.dropna(axis=0,how='any')  stop_words =
stopwords.words('english')
```

Following code performs different cleaning operations on Twitter data which includes removal of stopwords, punctuation, special characters, tokenization, stemming,etc:

```
class Twitter():
def clean_tweet(self):  def
remove_pattern(input, pattern):
 r = re.findall(pattern, input)
for i in r:
input = re.sub(i, '', input)  return
input
```

```
# remove twitter handles (@user)

df['cleaned_tweet'] = np.vectorize(remove_pattern)(df['Tweets'], "@[\w]*")
# remove special characters df['cleaned_tweet'] =
df['cleaned_tweet'].str.replace("[^a-zA-Z#]", " ")  #
remove words less than length 3 df['cleaned_tweet'] =
df['cleaned_tweet'].apply(lambda i: ' '.join([word for word in i.split() if len(word) > 3]))
# remove URLs df['cleaned_tweet'] =
df['cleaned_tweet'].apply(lambda i: re.split('https:\/\/.*', str(i))[0])
df['cleaned_tweet'] = df['cleaned_tweet'].replace(r'[^A-Za-z0-9 ]+', '', regex=True)
# remove numbers df['cleaned_tweet'] =
df['cleaned_tweet'].str.replace(r'\d+', '')
# remove special character hashtag"#" df['cleaned_tweet'] =
df['cleaned_tweet'].apply(lambda i: i.replace('#', ' '))
# convert all uppercase letters to lowercase
 df['cleaned_tweet'] = df['cleaned_tweet'].apply(lambda i: i.lower())
# Tokenization  tokens = df['cleaned_tweet'].apply(lambda x:
x.split()) tokens.head()  tokens = tokens.apply(lambda y:
[stemmer.stem(i) for i in y]) #  stemming #
df['tokens']=tokenized_tweet
# df.to_csv('Tokens.csv')
for i in range(len(tokens)):
tokens[i] = '
'.join(tokens[i])
df['cleaned_tweet'] = tokens
cleaned_tweets =  tokens
 # writing cleaned tweets to .csv file
df.to_csv('cleaning2.csv')  return
cleaned_tweets  if name == ' main
': tw=Twitter()  tw.clean tweet()
```

Script for Labeling Tweets

Following piece of code finds out the sentiments of tweets using TextBlob library. It checks if the polarity of a tweet is less than zero then tweet is negative, if polarity is greater than zero then tweet is positive and if it is equal to zero then tweet is neutral.

```
from textblob import TextBlob  import
pandas as pd  import csv
df=pd.read_csv('cleaning2.csv')
cleaned_tweets=df['cleaned_tweet']
class Labelling():  def
get_sentiment(self,cleaned_tweets):

 # sentiment analysis of tweets using
TextBlob  analysis =
TextBlob(cleaned_tweets)  if
analysis.sentiment.polarity > 0:  return
'positive'  elif analysis.sentiment.polarity ==
0:
return 'neutral'
else:
return 'negative'
```

Following function “get_tweets()” takes cleaned tweets and store text of tweets and sentiments in separate columns of dataframe:

```
def get_tweets(self):
#empty list to store tweets
list_of_tweets = []  #
iterating through tweets  for
tweet in cleaned_tweets:
# empty dictionary to store tweets' text and sentiment
tweet_dic = {}
# storing text part of tweet
tweet_dic['Tweets'] = tweet
# storing sentiment of tweet
 tweet_dic['Sentiment'] = self.get_sentiment(tweet)
```

In main function def main (), positive tweets are labeled as “positive”, negative tweets are labeled as “negative” and neutral tweets as “neutral.” Then all tweets are written along with their labels in csv file.

```
# appending parsed tweet to tweets list
            if tweet_dic not in list_of_tweets:
                list_of_tweets.append(tweet_dic)

                # return list of tweets
        return list_of_tweets

def main():

    # creating object of Labelling Class
    lab =Labelling()

    # calling function to get tweets
    tweets_data = lab.get_tweets()

    # selecting pos tweets from all gathered tweets
    ptweets_text=[tw for tw in tweets_data if tw['Sentiment'] == 'positive']

    # percentage of positive tweets
    print("Percentage of Positive tweets : {} %".format(100 * len(ptweets_text) / len(tweets_data)))

    # picking negative tweets from tweets
    ntweets_text=[tw for tw in tweets_data if tw['Sentiment'] == 'negative']

    # percentage of negative tweets
    print("Percentage of Negative tweets : {} %".format(100 * len(ntweets_text) / len(tweets_data)))

    # percentage of neutral tweets
    neutral_text=[tweet for tweet in tweets_data if tweet['Sentiment'] == 'neutral']
    tweet_length = len(tweets_data)
    nlength = len(ntweets_text)
    plength = len(ptweets_text)
    print("Percentage of Neutral tweets : {} % ".format(100 * (tweet_length - nlength - plength) /
tweet_length))

    csv_columns=['Tweets','Sentiment']
```

```
            with open('Labelled_tweets.csv', 'a') as csvfile:
        writer = csv.DictWriter(csvfile, fieldnames=csv_columns)
        writer.writeheader()
        for data in neutral_text:
            writer.writerow(data)

if _name_ == '_main_':
    main()
```

Model Training Using SVM

Following piece of code imports various libraries that are necessary for model training and reading data frames from .csv files.

```
import pandas as pd
from sklearn.svm import LinearSVC
from sklearn.feature_extraction.text import CountVectorizer from
sklearn.feature_extraction.text import TfidfVectorizer from
sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split from
nltk.stem.porter import *
stemmer = PorterStemmer() import sys
if not sys.warnoptions: import warnings
   warnings.simplefilter("ignore") df1=pd.read_csv('final_train.csv')
df2=pd.read_csv('final_test.csv') df1=df1.dropna(axis=0,how='any')
df2=df2.dropna(axis=0,how='any')
df=pd.read_csv('output_data.csv')
```

Following lines create a feature matrix for "Bag of Words" using "CountVectorizer" function:

```
bow= CountVectorizer(max_df=0.90, min_df=2, max_features=3000,
stop_words='english')
# Extracting features using bag of words approach
bag_of_words = bow.fit_transform(df['Tweets'])
print(bag_of_words)
```

These lines train SVM using n-gram approach.

N-grams increase the predictive power of classifiers because these find out the overall sentiment of word range i.e. two to three words.

```
ngram= CountVectorizer(binary=True, ngram_range=(1, 2))
ngram.fit(df['Tweets'])
X = ngram.transform(df['Tweets'])
X_test = ngram.transform(df2['Tweets'])
output=df['Sentiment']
X_train, X_val, y_train, y_val = train_test_split(
   X, output, train_size=0.75
)

c=1.0    #85.76
svm = LinearSVC(C=c)
svm.fit(X_train, y_train)
print("Accuracy with SVM(ngrams)  for C=%s: %s"
     % (c, accuracy_score(y_val, svm.predict(X_val))))
```

Following piece of code train SVM using TF-IDF approach.
TF-IDF increases accuracy because it gives more weightage to specific and relevant terms than irrelevant and common terms:

tfidf = TfidfVectorizer() tfidf.fit(df['Tweets'])
X = tfidf.transform(df['Tweets'])
X_test = tfidf.transform(df2['Tweets'])
target=df['Sentiment']
X_train, X_val, y_train, y_val = train_test_split(X, target, train_size=0.75) c=6.0 #86.1%
svm = LinearSVC(C=c) svm.fit(X_train, y_train) print("Accuracy with SVM(Tfidf)
for C=%s: %s" % (c, accuracy_score(y_val, svm.predict(X_val))))

Model Training using Naïve Bayes

Following lines of code just import supporting libraries for model training:

```
import pandas as pd
import numpy as np
from sklearn.svm import LinearSVC
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.metrics import accuracy_score
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from nltk.stem.porter import *
stemmer = PorterStemmer()
import re
import sys
if not sys.warnoptions:
   import warnings
   warnings.simplefilter("ignore")
df=pd.read_csv('output_data.csv')
```

In following lines, Naïve Bayes is classifying tweets using TF-IDF approach:

```
vect = CountVectorizer()
vect_count = vect.fit_transform(df['Tweets'])
tf_idf = TfidfTransformer().fit(vect_count)
vect_count = tf_idf.transform(vect_count)
print(vect_count)
X_train, X_test, y_train, y_test = train_test_split(vect_count, df['Sentiment'], test_size=0.25)
model_NB = MultinomialNB().fit(X_train, y_train)
predictions = model_NB.predict(X_test)
print("Accuracy with Naive Bayes (Tfidf):  %s"
      % ( accuracy_score(y_test, predictions)))
```

Using word-counts approach, Naïve Bayes is classifying tweets in following code:

```
word_count = CountVectorizer(binary=False)
word_counts = word_count.fit_transform(df['Tweets'])
X_train, X_test, y_train, y_test = train_test_split(word_counts, df['Sentiment'], test_size=0.25)
model_NB = MultinomialNB().fit(X_train, y_train)
predictions = model_NB.predict(X_test)
print("Accuracy with Naive Bayes (word_count):  %s"
% ( accuracy_score(y_test, predictions)))
```

Using n-grams approach, Naïve Bayes is classifying tweets in following code:

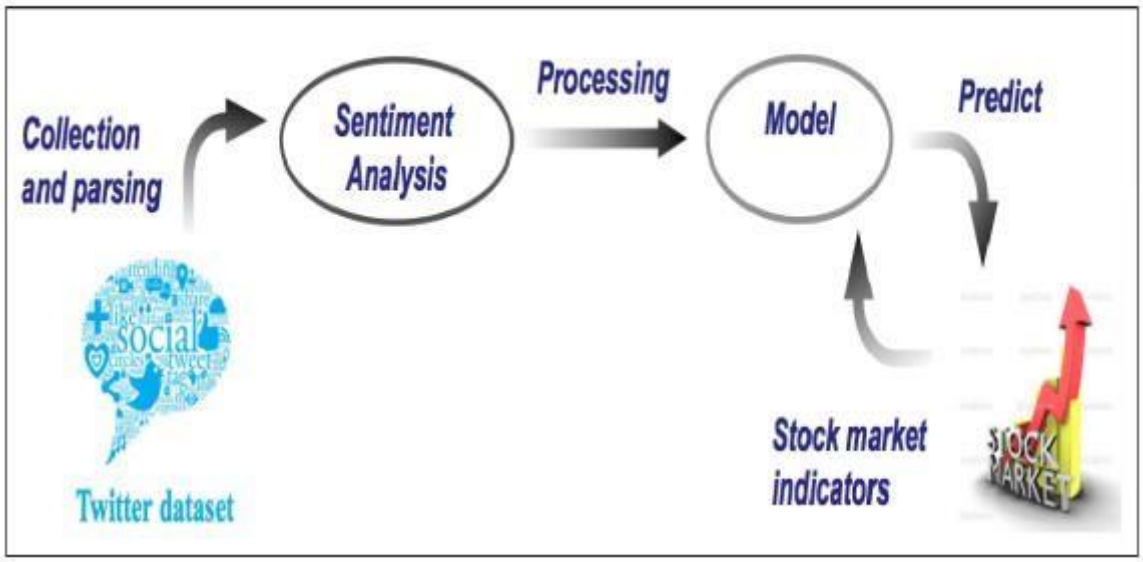
```
ngram= CountVectorizer(binary=True, ngram_range=(2, 3))
ngram.fit(df['Tweets'])
X = ngram.transform(df['Tweets'])
output=df['Sentiment']
X_train, X_test, y_train, y_test = train_test_split(
   X, output, train_size=0.75
)
model_NB = MultinomialNB().fit(X_train, y_train)
predictions = model_NB.predict(X_test)
print("Accuracy with Naive Bayes (N-grams):  %s"
% ( accuracy_score(y_test, predictions)))
```

CHAPTER 8

SNAPSHOTS

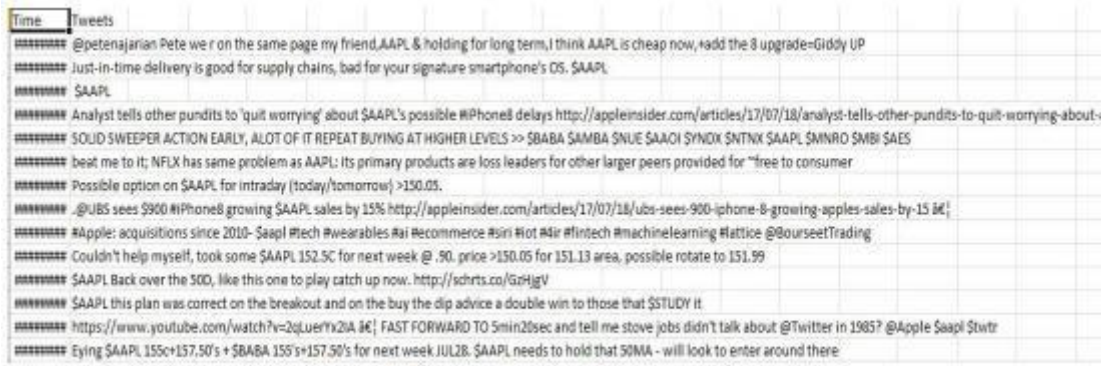
8. SNAPSHOTS


Prediction


Collection and parsing
Sentiment Analysis
Processing
Model
Predict
Twitter dataset
Stock market indicators

RESULTS Data Gathering of Tweets

Tweets for company Apple are gathered using two tags "#apple" and "#appl". Ten thousand tweets were scraped from Twitter by running my own written script for days. These tweets were then saved in a csv file along with their timestamps as it is shown in following figure:

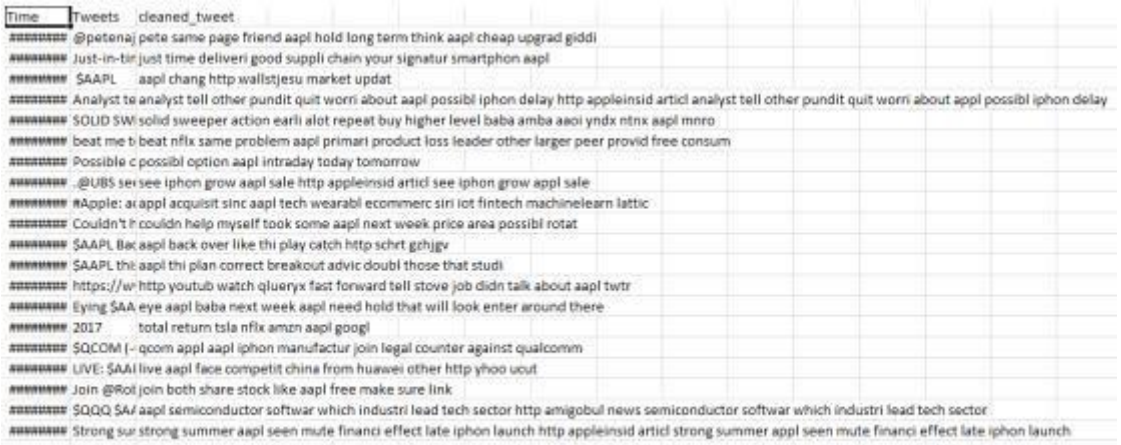


Time	Tweets
########	@petenajarian Pete we r on the same page my friend,AAPL & holding for long term,I think AAPL is cheap now, +add the 8 upgrade=Giddy UP
########	Just-in-time delivery is good for supply chains, bad for your signature smartphone's OS. $AAPL
########	$AAPL
########	Analyst tells other pundits to 'quit worrying' about $AAPL's possible #iPhone8 delays http://appleinsider.com/articles/17/07/18/analyst-tells-other-pundits-to-quit-worrying-about-
########	SOLID SWEEPER ACTION EARLY, ALOT OF IT REPEAT BUYING AT HIGHER LEVELS >> $BABA $AMBA $NUE $AAOI $YNDX $NTNX $AAPL $MNRO $MBI $AES
########	beat me to it; NFLX has same problem as AAPL: its primary products are loss leaders for other larger peers provided for "free to consumer
########	Possible option on $AAPL for intraday (today/tomorrow) >150.05.
########	.@UBS sees $900 #iPhone8 growing $AAPL sales by 15% http://appleinsider.com/articles/17/07/18/ubs-sees-900-iphone-8-growing-apples-sales-by-15 â€¦
########	#Apple: acquisitions since 2010- $aapl #tech #wearables #ai #ecommerce #siri #iot #dir #fintech #machinelearning #lattice @BourseetTrading
########	Couldn't help myself, took some $AAPL 152.5C for next week @ .90. price >150.05 for 151.13 area, possible rotate to 151.99
########	$AAPL Back over the 50D, like this one to play catch up now. http://schrts.co/GzHjgV
########	$AAPL this plan was correct on the breakout and on the buy the dip advice a double win to those that $STUDY it
########	https://www.youtube.com/watch?v=2qLuerYx2iA â€¦ FAST FORWARD TO 5min20sec and tell me steve jobs didn't talk about @Twitter in 1985? @Apple $aapl $twtr
########	Eying $AAPL 155c+157.50's + $BABA 155's+157.50's for next week JUL28. $AAPL needs to hold that 50MA - will look to enter around there

Figure Sample of gathered tweets

Cleaning of Tweets

After collecting tweets, gathered tweets were cleaned by removing stopwords, URLs, punctuation, tokenization and stemming. Natural Language Toolkit (NLTK) was used in cleaning and pre-processing of tweets. Cleaned tweets were then saved in a csv file as shown in following figure:



Time	Tweets	cleaned_tweet
########	@petenaj	pete same page friend aapl hold long term think aapl cheap upgrad giddi
########	Just-in-tir	just time deliveri good suppli chain your signatur smartphon aapl
########	$AAPL	aapl chang http wallstjesu market updat
########	Analyst te	analyst tell other pundit quit worri about aapl possibl iphon delay http appleinsid articl analyst tell other pundit quit worri about appl possibl iphon delay
########	SOLID SW	solid sweeper action earli alot repeat buy higher level baba amba aaoi yndx ntnx aapl mnro
########	beat me t	beat nflx same problem aapl primari product loss leader other larger peer provid free consum
########	Possible c	possibl option aapl intraday today tomorrow
########	.@UBS se	see iphon grow aapl sale http appleinsid articl see iphon grow appl sale
########	#Apple: a	appl acquisit sinc aapl tech wearabl ecommerc siri iot fintech machinelearn lattic
########	Couldn't h	couldn help myself took some aapl next week price area possibl rotat
########	$AAPL Ba	aapl back over like thi play catch http schrt gzhjgv
########	$AAPL thi	aapl thi plan correct breakout advic doubl those that studi
########	https://w	http youtub watch qlueryx fast forward tell stove job didn talk about aapl twtr
########	Eying $AA	eye aapl baba next week aapl need hold that will look enter around there
########	2017	total return tsla nflx amzn aapl googl
########	$QCOM (-	qcom appl aapl iphon manufactur join legal counter against qualcomm
########	LIVE: $AA	live aapl face competit china from huawei other http yhoo ucut
########	Join @Rol	join both share stock like aapl free make sure link
########	$QQQ $A/	aapl semiconductor softwar which industri lead tech sector http amigobul news semiconductor softwar which industri lead tech sector
########	Strong su	strong summer aapl seen mute financ effect late iphon launch http appleinsid articl strong summer appl seen mute financ effect late iphon launch

Figure 4 Cleaned Tweets

Labeling of Tweets

After cleaning data, now comes the step of labeling tweets by doing sentiment analysis. This is done using TextBlob library. It does sentiment analysis of tweets and label tweets according to the polarity of Tweets' sentiments. Tweets with polarity greater than zero are labeled as "positive", polarity with less than zero are labeled as "negative" and polarity with equal to zero are labeled as "neutral". Labeling of Tweets is shown in following figure:

Tweets	Sentiment
chart mas	positive
tech aapl	positive
nutshel p	neutral
happen a	neutral
hiph lalpi	positive
thi case d	neutral
bank ame	positive
imagin to	positive
these bab	neutral
australia s	positive
appl scien	neutral
said thi p	positive
time burn	neutral
held level	positive

Figure Sample of Labeled Tweets

Results of Classification using SVM

Firstly, the researcher applied SVM using N-grams approach that is discussed in detail in previous chapters. N-grams approach says that using more than one gram or one word feature for finding out the sentiment of entire sentence. The researcher used range of one to two grams means one to two word features for determining the sentiment of entire sentence. Regularization parameter was adjusted by trying different values of "C". With N-grams SVM gave accuracy of 91.2% at "C=1.0". Following screenshot shows the accuracy of SVM with n-grams:

```
Accuracy with SVM(ngrams)  for C=1.0: 0.9121171770972037
```

Figure 6 Accuracy of SVM with N-Grams

Then the researcher applied SVM using TF-IDF approach that is also described in detail in previous chapters. TF-IDF approach gives more weight to rare and specific terms than common and irrelevant terms. This is the reason that SVM gives more accuracy with TF-IDF. Following screenshot shows the accuracy of SVM with TF-IDF:

```
Accuracy with SVM(Tfidf)  for C=6.0: 0.9325343985796716
```

Figure Improved Accuracy of SVM with TF-IDF

Results of Classification using Naïve Bayes

In Naïve Bayes, the researcher has used two approaches that are N-grams and Word- counts. Ngrams approach is explained in previous paragraph. The researcher will explain about wordcounts approach. Word counts approach says that if a positive word occurs many times in a sentence it means this sentence has positive sentiment overall. The researcher got 73.7% accuracy using N grams approach. Following figure shows the accuracy of Naïve Bayes with

grams:

```
Accuracy with Naive Bayes (N-grams):  0.7376830892143809
```

Figure Accuracy of Naive Bayes (N-grams)

Then the researcher applied Word Counts approach. It gives more accuracy than N-grams because in word counts approach if a sentence contains more positive words, it means sentence has positive sentiment overall. Following is the screenshot that shows the improved accuracy of Naïve Bayes with word counts:

```
Accuracy with Naive Bayes (word_count):  0.7833999112294718
```

Figure Improved Accuracy of Naive Bayes

CHAPTER 9 CONCLUTION

9. CONCLUSION

The proposed solution is an excellent guidance to investors of all kinds. It is a fully functional, intelligent system that predicts stock market of "Apple" in the US Stock Market with an excellent accuracy. The proposed solution uses hybrid approach to predict stock market. Hybrid approach means to combine two different approaches. The proposed system combines sentiment analysis of Tweets as an extra feature along with historical or numerical data of the company. Five features which include followers (followers of a tweet), polarity (polarity of a tweet), sentiment confidence, clusters (clusters of tweets according to polarity and sentiment confidence) and difference (difference of open and close price of stock) are given as input to Machine Learning models SVR and LSTM.

The main challenge in the project is collecting tweets from social network website Twitter for this the researcher first used Twitter API's, but the problem with Twitter API's are that they give a very smaller number of tweets for free. After that the researcher changed methodology and the researcher scrapped twitter webpages in order to gather more data, when he researcher get enough data the researcher preprocess that data, remove time stamps, and remove stop words form the text and feed the cleaned text into SVM and Naïve Bayes.

Challenges

Predicting the stock market in itself is one of the biggest mysteries of this world. The first and foremost challenge was the project itself because it was a culmination of different modules that were huge challenges in themselves.

In light of that, the biggest challenges we faced include:

- **Coming-up with an accurate prediction model:** At first, predictions from the LSTM were not that good due to the fact that that the subsequent values were different to the ones on which the LSTM was previously trained on. To counter that, the researcher came up with retraining the

LSTM on daily basis so that it becomes acquainted with what's happening currently so that it can predict better.

- **Building prediction algorithm:** Building prediction algorithm was also a challenging task primarily due to the fact that the stock market was an unknown quantity for the researcher.

- **Learning new technologies:** Another mighty challenge the researcher came across was learning different Machine Learning related technologies.

Recommendations

Ordinary people may use Stock markets to make fortune while financial analysts may use them to determine a country's economy and analyze the growth pattern and impact of government policies and economic situation prevailing in the country. With the use of sentimental analysis along with web scrapping more accurate results on stock prediction can be made. Using Neural Networks and artificial intelligence embedded into such a system it will be possible for us to implement automatic training and analysis of data. A self-learning system can be built which trains itself from the data obtained and use that for further prediction. Once human behavior patterns are formulated this data could be used for finding out the best possible time to make investments and this data could be also used by companies for taking financial decisions that could make an impact on companies' profit margins. A more personalized system can also be made which helps them to make right investments with the funds available and then predicting the rights stocks for investment. Such a system could also predict financial crisis that could occur. Creating such a system will require huge processing and computational power since huge chunks of data needs to be processed and evaluated. With the use of AI and Neural Nets, occurrence of a particular scenario is understood and conditions that causes it are also taken into account so that when such conditions occur in future it could trace back and warn the user about the situation that could occur. As the

world, today is moving towards more automated artificial intelligence-based technologies, such a system implanted in the financial field could be a great boost to the economy.

The researcher has scheduled numerous improvements to the project as the researcher aim to turn it into a fully functioning product. These are as follows:

- Collect news data from different news channels and combine news sentiments with our existing sentiments to have more authentic sentiments of the current day as well as previous days.
- Generate data from tweets for US Stock market and we will combine it the historical data of US Stock market in order to predict US stocks.
- Using a multivariate LSTM to predict future stock prices by incorporating more factors that influence the stock market. However, since not all factors can be quantified, this is a big challenge in itself.
- At the moment, proposed system predicts stock prices for one stock namely APPLE. The researcher aim to predict future stock prices for all stocks listed in the US stock market.
- Using a multi-node HADOOP cluster to store data in a fault-tolerant manner and provide even more scalability to the project during its post-product life cycle.

Critical Appraisal

When the researcher first set out to do this project, it was hard for the researcher to Internship report 2021-2022 59 fathom how we would be able to get high prediction accuracy, particularly for daily closing prediction. This is highly attributed to the fact that the stock market shows irregular patterns, something that even outmuscles Artificial Intelligence many a times. Time series analysis was an impossible attempt to predict expected future stock prices because of the irregular nature of the data points we collected and not having many features to work with further added to the uncertainty of making the project a success. However, through astute decision making and a good system architecture design, the researcher was able to pull-off excellent results, particularly for our hourly price prediction. The researcher can further bring

accuracy to predictions through reinforcement learning or a multivariate LSTM while the daily closing predictions can also be improved by further research.

Student Reflection In this research work we have managed the proposed solution to show how the stock market prediction can be made using sentimental analysis can be carried for efficient prediction of stocks. The first step starts off with scrapping the tweets related to Apple products. The next steps involved are preprocessing and cleaning the tweets scrapped. The third step carried out is classifying the tweets as positive, negative and neutral. In the next steps, the data is fed into models, which are in turn trained by Naïve Bayes and SVM algorithm. These algorithms are then improved by changing or altering the parameters according in order to increase the accuracy. The next step includes the twitter sentiment analysis is then merged with the historical data to predict the stock market.

This project “Predictive Analysis of Stock Market using Sentiment Analysis of Twitter” aims to be a standout and accurate guidance for all kinds of investors in the country. The stock market is highly affected by political situation within a country. This is the reason the researcher proposed methodology that consists of sentiment analysis to analyze political situation within a country. The researcher also has managed to learn technically python coding, the algorithms for model training such as Naïve Bayes and SVM algorithm, the modules and packages used for web scrapping, preprocessing of tweets, classifying and labeling of tweets, prediction of stock market by comparing it with historical stock data.

10. REFERENCE

1. Bollen, J., Mao, H., and Zeng, X. (2011) 'Twitter Mood Predicts the Stock Market'. Journal of Computational Science 2 (1), 1–8

2. Brown, E.D. (2012) 'Will Twitter Make You a Better Investor? A Look at Sentiment, User Reputation and Their Effect on the Stock Market'. Proc. of SAIS, 36–42 3. Dash, R. and Dash, P.K. (2016) 'A Hybrid Stock Trading Framework Integrating Technical Analysis with Machine Learning Techniques'. The Journal of Finance and Data Science 2 (1), 42–57 4. Deng, Y., Bao, F., Kong, Y., Ren, Z., and Dai, Q. (2017) 'Deep Direct Reinforcement Learning for Financial Signal Representation and Trading'. IEEE Transactions on Neural Networks and Learning Systems 28 (3), 653–664

5. Goel, A. and Mittal, A. (2012) Stock Prediction Using Twitter Sentiment Analysis. Standford University, CS229. [online] available from

6. Hoepfl, M.C. (1997) 'Choosing Qualitative Research: A Primer for Technology Education

7. 8. Researchers'. Journal of Technology Education 9 (1)

9. 10. Huarng, K.-H., Rey-Mart, A., and Miquel-Romero, M.-J. (2018) 'Quantitative and Qualitative Comparative Analysis in Business'. Journal of Business Research 89, 171–174

11. 'Ijsetr.Org' (2019) in Ijsetr.Org [online] available from

12. Kamran, R. (2019) Prediction of Stock Market Performance by Using Machine Learning Techniques.

13. Karim, S., Abdullah, T., and Tayaba, U. (2018) 'Predicting Stock Market Trend from Twitter Internship Report 2021-22 Page 63

14. Feed and Building a Framework for Bangladesh'. Doctoral Dissertation, BRAC University 15. 16. L.Lima, Milson, P. Nascimento, T. (n.d.) 'Using Sentiment Analysis for Stock Exchange Prediction'. International Journal of Artificial Intelligence & Applications 7 (1), 59–67 20. M, M., S, S., and W, L. (2009) Stock Prediction Using Twitter Sentiment Analysis. 1

17.18. Porshnev, A., Redkin, I., and Shevchenko, A. (2013) 'Machine Learning in Prediction of Stock Market Indicators Based on Historical Data and Data from Twitter Sentiment Analysis'. . . In 2013 IEkEE 13th International Conference on Data Mining Workshops (Pp. 440-444). IEEE. 19. Seghal, V. and Song, C. (2007) 'SOPS: Stock Prediction Using Web Sentiment'. Proc. -
IEEE Int. Conf. Data Mining, ICDM 21–26 24. Shah, V.H. (2007) 'Machine Learning Techniques for Stock Prediction'. Foundations of Machine Learning| Spring, 1(1), 6–12 25. Shams, N.Z. and Muhammed, Z. (2005) 'Stock Price Prediction Using Artificial Neural

20. 21. Networks: Case Study'. Journal of Independent Studies and Research (JISR) 3 (2)

22. 23. Smailovic, J., Grcar, M., Lavrac, N., and nidaric, M. (2014) 'Predictive Sentiment Analysis of Tweets: A Stock Market Application. In International Workshop on Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data (Pp. 77-88).

24. Springer, Berlin, Heidelberg.' Information Sciences

25.26. Uc, X. and Yue, S. (2019) Stock Price Forecasting Using Information from Yahoo Finance and Google Trend.

27. V.S, P., K.N.R, C., G, P., and B, M. (2016) ‘Sentiment Analysis of Twitter Data for Predicting Stock Market Movements’. Scopes 6

28. Wenger, Z. (1991) ‘Qualitative Evaluation and Research Methods (2nd Ed.). By Michael

Quinn

Internship report