CHAPTERS

CHAPTER : - 1
INTRODUCTION

1.1 Introduction

Data analysis is the process of applying organized and systematic statistical techniques to describe, recap, check and condense data. It is a multistep process that involves collecting, cleaning, organizing and analyzing. Data mining is like applying techniques to mold data to suit our requirement. Data mining is needed because different sources like social media, transactions, public data, enterprises data etc. generates data of increasing volume, and it is important to handle and analyze such a big data. It won't be wrong to say that social media is something we live by. In the 21st century social media has been the game changer, be it advertising, politics or globalization, it has been estimated that data is increasing faster than before and by the year 2020; about 1.7 megabytes of additional data will be generated each instant for each person on the earth. More data has been generated in the past two years than ever before in the history of the mankind. It is clear from the fact that the number of internet users are now grown from millions to billions.

Database which is opted for the proposed study is from Twitter. It is now day's very popular service which provides facility of microblogging. In this people write short messages generally less than 140 characters, about 11 words on average. It is appropriate for analysis as the number of messages is large. It is much easier task as compared to searching blogs from the net. The objective of the proposed analysis, 'Sentiment Analysis', is the analysis of the enormous amount of data easily available from social media.

Algorithm generates an overall sentiment score from the inputted topic in terms of positive, negative or neutral, further it also works on finding the frequency of the words being used. Word cloud that is a pictorial representation of words based on frequency occurrence of words in the text is also generated. Calculation is actualized utilizing R attributable to its component rich, thorough and expressive abilities for measurable information.

1.2 Scope

This project will be helpful to the companies, political parties as well as to the common people. It will be helpful to political party for reviewing about the program that they are going to do or the program that they have performed. Similarly companies also can get review about their new product on newly released hardware or softwares. Also the movie maker can take review

on the currently running movie. By analyzing the tweets analyzer can get result on how positive or negative or neutral are peoples about it

1.3 Project summary and purpose

This project of analysing sentiments of tweets comes under the domain of *"Pattern Classification"* and *"Data Mining"*. Both of these terms are very closely related and intertwined, and they can be formally defined as the process of discovering "useful" patterns in large set of data, either automatically (unsupervised) or semi automatically (supervised). The project would heavily rely on techniques of "Natural Language Processing" in extracting significant patterns and features from the large data set of tweets and on *"Machine Learning"* techniques for accurately classifying individual unlabelled data samples (tweets) according to whichever pattern model best describes them.

1.4 Overview of the project

This proposal is a web application which is used to analyze the tweets. We will be performing sentiment analysis in tweets and determine where it is positive, negative or neutral. This web application can be used by any organization office to review their works or by political leaders or by any others company to review about their products or brands.

The main feature of our web application is that it helps to determine the opinion about the peoples on products, government work, politics or any other by analyzing the tweets. Our system is capable of training the new tweets taking reference to previously trained tweets.

The computed or analyzed data will be represented in various diagram such as Pie- chart, Bar graph and Word cloud.

1.5 Problem definition

The algorithm proposed works on Twitter Data, primarily it collects the tweets and then study it with the help of different statistical computing procedures. Twitter account once registered and logged in, needs registering the application name on Twitter API to create our application which provide us the four legal credentials (API_key, API_secret, access_token, access_token_secret) required for connection establishment.

CHAPTER : -2
TECHNOLOGY AND LITERATURE REVIEW

2.1 Brief History of Work Done

Sentiment analysis has been handled as a Natural Language Processing task at many levels of granularity. Starting from being a document level classification, it has been handled at the sentence level and more recently at the phrase level (Wilson et al., 2005; Agarwal et al., 2009). Microblog data like Twitter, on which users post real time reactions to and opinions about "everything", poses newer and different challenges. Some of the early and recent results on sentiment analysis of Twitter data are by Go et al. (2009), (Bermingham and Smeaton, 2010) and Pak and Paroubek (2010) . Go et al. (2009) use distant learning to acquire sentiment data. They use tweet sending in positive emotions like ":)" ":-)" as positive and negative emoticons like ":(" " :-(" as negative. They build models using Naive Bayes, Max Ent and Support Vector Machines (SVM), and they report SVM outperforms other classifiers. In terms of feature space, they try a Unigram, Bigram model in conjunction with parts-of-speech (POS) features. They note that the unigram model outperforms all other models. Specifically, bigrams and POS features do not help. Pak and Paroubek (2010) collect data following a similar distant learning paradigm. They perform a different classification task though: subjective versus objective. For subjective data they collect the tweets ending with emoticons in the same manner as Go et al. (2009). For objective data they crawl Twitter accounts of popular newspapers like "New York Times", "Washington Posts" etc. They report that POS and bigrams both help (contrary to results presented by Go et al. (2009)). Both these approaches, however, are primarily based on ngram models. Moreover, the data they use for training and testing is collected by search queries and is therefore biased. In contrast, we present features that achieve a significant gain over a unigram baseline. In addition we explore a different method of data representation and report significant improvement over the unigram models. Another contribution of this paper is that we report results on manually annotated data that does not suffer from any known biases. Our data will be a random sample of streaming tweets unlike data collected by using specific queries. The size of our hand-labeled data will allow us to perform cross validation experiments and check forth variance in performance of the classifier across folds. Another significant effort for sentiment classification on Twitter data is by Barbosa and Feng (2010). They use polarity predictions from three websites as noisy labels to train a model and use 1000 manually labeled tweets for tuning and another 1000 manually labeled tweets for testing. They however do not mention how they collect their test data. They propose the use of syntax features of tweets like retweet, hash tags, link, punctuation and exclamation marks in conjunction with features like prior polarity of words and POS of words. We extend their approach by using real valued prior polarity, and by combining prior polarity with POS. Our results show that the features that enhance the performance of our classifiers the most are features that combine prior polarity of words with their parts of speech. The tweet syntax features help but only marginally. Gamon (2004) perform sentiment analysis on feedback data from Global Support Services survey. One aim of their paper is to analyze the role of linguistic features like POS tags. They perform extensive feature analysis and feature selection and demonstrate that

abstract linguistic analysis features contributes to the classifier accuracy. In this paper we perform extensive feature analysis and show that the use of only 100 abstract linguistic features performs as well as a hard unigram baseline. One fundamental problem in sentiment analysis is categorization of sentiment polarity .Given a piece of written text, the problem is to categorize the text into one specific sentiment polarity, positive or negative (or neutral). Based on the scope of the text, there are three levels of sentiment polarity categorization, namely the document level, the sentence level, and the entity and aspect level. The document level concerns whether a document, as a whole, expresses negative or positive sentiment, while the sentence level deals with each sentence's sentiment categorization. The entity and aspect level then targets on what exactly people like or dislike from their opinions. For feature selection, Pang and Lee suggested to remove objective sentences by extracting subjective ones. They proposed a text-categorization technique that is able to identify subjective content using minimum cut. Gann et al. Selected 6799 tokens based on Twitter data, where each token is assigned a sentiment score, namely TSI (Total Sentiment Index), featuring itself as a positive token or a negative token. Specifically, a TSI for a certain token is computed as: $TSI = p - tp\ tn * n\ p + tp\ tn * n$ where p is the number of times a token appears in positive tweets and n is the number of times a token appears in negative tweets. $tp\ tn$ is the ratio of total number of positive tweets over total number of negative tweets. Moreover, showed that using the well-known "geo-tagged" feature in Twitter to identify the polarity of political candidates in the US could be done by employing the sentiment analysis algorithms to predict the future events such as the presidential elections results. Comparing to previous approaches in sentiment topics, additional findings by showed that adding the semantic feature produces better Recall (retrieved documents) to compute the score) in negative sentiment classification.

CHAPTER : - 3

SYSTEM REQUIREMENTS STUDY

3.1 User Characteristics

Sentiment analysis can be defined as a process that automates mining of attitudes, opinions, views and emotions from text, speech, tweets and database sources through Natural Language Processing (NLP).

Sentiment analysis involves classifying opinions in text into categories like "positive" or "negative" or "neutral". It's also referred as subjectivity analysis, opinion mining, and appraisal extraction.

The words opinion, sentiment, view and belief are used interchangeably but there are differences between them.

- ***Opinion****:* A conclusion open to dispute (because different experts have different opinions)
- ***View:*** subjective opinion
- ***Belief:*** deliberate acceptance and intellectual assent

Sentiment: opinion representing one's feelings

Sentiment Analysis is a term that include many tasks such as sentiment extraction, sentiment classification, subjectivity classification, summarization of opinions or opinion spam detection, among others.

It aims to analyze people's sentiments, attitudes, opinions emotions, etc. towards elements such as, products, individuals, topics ,organizations, and services.

3.2 SOFTWARE AND HARDWARE REQUIREMENTS:

SOFTWARE REQUIREMENTS:-

Operating System: Windows 7/8/8.1/10

Microsoft Visio (2016)

Microsoft Word (2016)

Atom (Text Editor)

R Studio

HARDWARE SPECIFICATIONS:-

Processor	:	Intel i5 or more
Motherboard	:	Intel® Chipset Motherboard.
Ram	:	8GB or more
Cache	:	512 KB
Hard disk	:	16 GB hard disk recommended
Disk Drive	:	1.44MB Floppy Disk Drive
Monitor	:	1024 x 720 Display
Speed	:	2.7GHZ and more

CHAPTER :- 4
SYSTEM ANALYSIS

4.1 Feasibility Study:

A feasibility study is a preliminary study which investigates the information of prospective users and determines the resources requirements, costs, benefits and feasibility of proposed system. A feasibility study takes into account various constraints within which the system should be implemented and operated. In this stage, the resource needed for the implementation such as computing equipment, manpower and costs are estimated. The estimated are compared with available resources and a cost benefit analysis of the system is made. The feasibility analysis activity involves the analysis of the problem and collection of all relevant information relating to the project. The main objectives of the feasibility study are to determine whether the project would be feasible in terms of economic feasibility, technical feasibility and operational feasibility and schedule feasibility or not. It is to make sure that the input data which are required for the project are available. Thus we evaluated the feasibility of the system in terms of the following categories:

- Technical feasibility
- Operational feasibility
- Economic feasibility
- Schedule feasibility

4.1.1 Technical Feasibility

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time there is no any detailed designed of the system, making it difficult to access issues like performance, costs (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis; understand the different technologies involved in the proposed system. Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system. Is the required technology available? Our system is technically feasible since all the required tools are easily available. R Studio and Shiny makes the system more user and developer friendly and although all tools seems to be easily available there are challenges too.

4.1.2 Operational Feasibility

Proposed project is beneficial only if it can be turned into information systems that will meet the operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? The proposed was to make a simplified web application. It is simpler to operate and can be used in any webpages. It is free and not costly to operate.

4.1.3 Economic Feasibility

Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This

feasibility study gives the top management the economic justification for the new system. A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could increase improvement in product quality, better decision making, and timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information. This is a web based application. Creation of application is not costly.

4.1.4 Schedule Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given period of time using some methods like payback period. Schedule feasibility is a measure how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some project is initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.
A minor deviation can be encountered in the original schedule decided at the beginning of the project. The application development is feasible in terms of schedule.

4.2 Requirement Definition:

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 and non-functional requirements. These requirements are listed below:

4.2.1 Functional Requirements

Functional requirement 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 that the system must require are as follows:
-System should be able to process new tweets stored in database after retrieval
-System should be able to analyze data and classify each tweet polarity

4.2.2 Non-Functional Requirements

Non-functional requirements is a description of features, characteristics and attribute 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.
Based on these the non-functional requirements are as follows:

 -User friendly
 -System should provide better accuracy
 -To perform with efficient throughput and response time

4.3 Study of Current System

There are primarily two types of approaches for sentiment classification of opinionated texts:

Using a Machine learning based text classifier such as Naive Bayes

- Using Natural Language Processing

We will be using those machine learning and natural language processing for sentiment analysis of tweets.

Machine Learning

The machine learning based text classifiers are a kind of supervised machine learning paradigm, where the classifier needs to be trained on some labeled training data before it can be applied to actual classification task. The training data is usually an extracted portion of the original data hand labelled manually. After suitable training they can be used on the actual test data. The Naive Bayes is a statistical classifier whereas Support Vector Machine is a kind of vector space classifier. The statistical text classifier scheme of Naive Bayes (NB) can be adapted to be used for sentiment classification problem as it can be visualized as a 2-class text classification problem: in positive and negative classes. Support Vector machine (SVM) is a kind of vector space model based classifier which requires that the text documents should be transformed to feature vectors before they are used for classification. Usually the text documents are transformed to multidimensional vectors. The entire problem of classification is then classifying every text document represented as a vector into a particular class. It is a type of large margin classifier. Here the goal is to find a decision boundary between two classes that is maximally far from any document in the training data.

This approach needs

- A good classifier such as Naive Bayes. A training set for each class

There are various training sets available on Internet such as Movie Reviews data set, twitter dataset, etc. Class can be Positive, negative. For both the classes we need training data sets.

R LANGUAGE

R is a coding language and software system utilized for the analysis of statistical data, representation of charts, graphs and reporting. R language was developed by Ross Ihaka and Robert Gentleman at Auckland University New Zealand. R is available freely under public license. Name of programming language R was derived from the first letter of first name of the two R developers (Robert Gentleman and Ross Ihaka).

Features of R Language

- R language is a well-developed, straight forward and efficient programming language. It includes loops,conditionals, recursive functions, and input and output facilities.
- R has an excellent storage and data handling facility.
- R provides a set of operators for vectors, arrays and matrices.

• R provides a list of wide collection of tools for data analysis.
• R is accessed through interpreter based on command line; it supports arithmetic operations which are matrix based. Data structure of R involve vectors, arrays, matrices, lists and data frames. Extendable object scheme of R contains objects for regression models, time-series and geo-spatial coordinates. The scalar data type is not a data structure of R. As an alternative, a scalar is expressed as a vector which is of length one.
• Procedural programing is supported by R language with functions, and object-oriented programming with generic functions for particular functions. It is mainly utilized by statisticians and mathematicians, needs an atmosphere for analysis of statistical data and development of software, R language is also utilized as a tool box for common matrix operations with performance standards similar to MATLAB or GNU octave.

Package :-

Performance of R language can be enhanced through a package which is created by user generally developed in C, C++ and java. For specific statistical method, graphical plots (ggplots), Import/ Export abilities, reporting tools (knitr, sweave) etc. R has a core group of packages; it is provided through the installation, with more than 7,801 extra packages, these include Comprehensive R Archive Network (CRAN), Bio conductor, Omega hat, GitHub, etc. The "Task Views" page on the website of CRAN provide a great variety of jobs (such as Finance, Genetics, Computing with good performance, Machine Learning, Medical Imaging, Social Sciences and Spatial Statistics) to which R has been utilized and for which packages are provided. R is also used by the Food and Drug Administration (FDA) asright for analyzing data from medical research. Some R package resources comprise Crantastic, which is an open site for rating and studying all CRAN packages, and R- Forge, a central platform for the collective enhancement of R packages, software associated to R, and projects. R-Forge also hosts various unpublished beta packages, and development of CRAN package. For the analysis of genomic data, the Bio-conductor project provides many R packages like Affymetrix and cDNA microarray object-oriented data-handling, and has begun to offer tools for examination of next generation data high throughput sequencing technique.

R Studio :-

RStudio is an IDE, integrated development environment. It offers management of workspace, it involves syntax highlighting editor, console and debugging. RStudio is an open supply software system although business versions are also provided with some improved features and it supports desktop computers which operates on windows, mac and Linux as well as on browser connected to RStudio.
Two versions available are:

a. Rstudio desktop: Software runs in the same way as desktop application.
b. Rstudio server: In this Rstudio is used to access web browser.

The proposed work was carried out using Rstudio Desktop. Features utilized were:

1) IDE was created specifically for R language.
   • Syntax is highlighted, completion of code and the smart indention
   • From the source editor R program can be executed directly
   • Rapidly switch to function definitions
2) Workflow is taken together

- Integrated R support and documentation
- using projects multiple working directories can be easily managed
- Data viewer and workplace browser

3) Influential authoring and fixing
- Quickly detect and fix errors.
- Tools Extensive package development.
- Authoring with Sweave and R Markdown

Shiny :-

Graphic User Interface for the proposed work 'Sentiment Analysis' was developed utilizing shiny package of RStudio. It is one of the strongest software supports provided by RStudio. Shiny is equipped with a lot of prominent interface enhancing features. It's an interactive and user-friendly app developing package.

Naïve Bayes Classifier (NB) :-

The Naïve Bayes classifier is the simplest and most commonly used classifier. Naïve Bayes classification model computes the posterior probability of a class, based on the distribution of the words in the document. The model works with the BOWs feature extraction which ignores the position of the word in the document. It uses Bayes Theorem to predict the probability that a given feature set belongs to a particular label.

P(label|features) = P(label)∗P(features|label) P(features)

where ,

P(label) :- is the prior probability of a label or the likelihood that a random feature set the label.

P(features|label) :- is the prior probability that a given feature set is being classified as a label.

P(features) :- is the prior probability that a given feature set is occurred. Given the Naïve assumption which states that all features are independent, the equation could be rewritten as follows:

P(label|features) = P(label)∗P(f1|label)∗……∗P(fn|label) P(features)

Multinomial Naïve Bayes Classifier

Accuracy – around 75%

Algorithm :

i. Dictionary generation

Count occurrence of all word in our whole data set and make a dictionary of some most frequent words.

ii. Feature set generation

All document is represented as a feature vector over the space of dictionary words.

For each document, keep track of dictionary words along with their number of occurrence in that document.

Formula used for algorithms:

$$\phi_{k|label=y} = P(x_j = k \mid label = y)$$

$$\phi_{k|label=y} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\{x_j^{(i)} = k \text{ and } label^{(i)} = y\} + 1}{(\sum_{i=1}^{m} 1\{label^{(i)} = y\} n_i) + |V|}$$

$\phi_{k|label=y}$ = probability that a particular word in document of label(neg/pos) = y will be the k^{th} word in the dictionary.

m = Number of words in i^{th} document.

n_i = Total Number of documents.

Training:-

In this phase We have to generate training data(words with probability of occurrence in positive/negative train data files).

Calculate k|label for each label and dictionary words and store the result.(Here: label will be negative and positive.).

Now we have, words and corresponding probability for each of the defined label.

Testing Goal :-

- Finding the sentiment of given test data file.
- Generate Feature set(x) for test data file.
-For each document is test set find

Decision1=log P(x| label= pos)+log P(label=pos)

Similarly calculate,

Decision2=log P(x| label= neg)+log P(label=neg)

Compare decision 1&2 to compute whether it has Negative or Positive sentiment.

Natural Language Processing

Natural language processing (NLP) is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages. This approach utilizes the publicly available library of Opinion Lexicon, which provides a sentiment polarity values for every term occurring in the document. In this lexical resource each term t occurring in Opinion Lexicon is associated to three numerical scores obj(t), pos(t) and neg(t), describing the objective, positive and negative polarities of the term, respectively. These three scores are computed by combining the results produced by eight ternary classifiers. Opinion Lexicon is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, each expressing a distinct concept. Opinion Lexicon is also freely and publicly available for download. Opinion Lexicon's structure makes it a useful tool for computational linguistics and natural language processing.
It groups words together based on their meanings. It is nothing but a set of one or more Synonyms. This approach uses Semantics to understand the language.

Major tasks in NLP that helps in extracting sentiment from a sentence:

- Extracting part of the sentence that reflects the sentiment

- Understanding the structure of the sentence

Different tools which help process the textual data Basically, Positive and Negative scores got from Opinion Lexicon according to its part-of- speech tag and then by counting the total positive and negative scores we determine the sentiment polarity based on which class (i.e. either positive or negative) has received the highest score.

4.4 CHALLENGES IN SENTIMENT ANALYSIS:

Sentiment Analysis is a very challenging task. Following are some of the challenges faced in Sentiment Analysis of Twitter.

1. Identifying subjective parts of text:
Subjective parts represent sentiment-bearing content. The same word can be treated as subjective in one case, or an objective in some other. This makes it difficult to identify the subjective portions of text.
Example*:*

1. The language of the Mr Dennis was very crude.

2. Crude oil is obtained by extraction from the sea beds.

The word „crude“ is used as an opinion in first example, while it is completely objective in the second example.

2. Domain dependence:
The same sentence or phrase can have different meanings in different domains.

***Example**:*
The word “unpredictable” is positive in the domain of movies, dramas, etc, but if the same word is used in the context of a vehicle's steering, then it has a negative opinion.

3. Sarcasm Detection:
Sarcastic sentences express negative opinion about a target using positive words in unique way.
***Example**:*
“Nice perfume. You must shower in it.”
The sentence contains only positive words but actually it expresses a negative sentiment.
4. Thwarted expressions:
There are some sentences in which only some part of text determines the overall polarity of the document.
Example:
“This Movie should be amazing. It sounds like a great plot, the popular actors, and the supporting cast is talented as well. “
In this case, a simple bag-of-words approaches will term it as positive sentiment, but the ultimate sentiment is negative.

5. Explicit Negation of sentiment:
Sentiment can be negated in many ways as opposed to using simple no, not, never, etc. It is difficult to identify such negations.
Example:
“It avoids all suspense and predictability found in Hollywood movies.”
Here the words suspense and predictable bear a negative sentiment, the usage of „avoids“ negates their respective sentiments.

6. Order dependence:
Discourse Structure analysis is essential for Sentiment Analysis/Opinion Mining.
Example:
A is better than B, conveys the exact opposite opinion from, B is better than A.

7. Entity Recognition:
There is a need to separate out the text about a specific entity and then analyze sentiment towards it.
***Example*:**
“I hate Microsoft, but I like Linux”.
A simple bag-of-words approach will label it as neutral, however, it carries a specific sentiment for both the entities present in the statement.

8. Building a classifier for subjective vs. objective tweets.
Current research work focuses mostly on classifying positive vs. negative correctly. There is need to look at classifying tweets with sentiment vs. no sentiment closely.

9. Handling comparisons.
Bag of words model doesn't handle comparisons very well.
Example*:*
"IIT‟s are better than most of the private colleges", the tweet would be considered positive for both IIT‟s and private colleges using bag of words model because it doesn't take into account the relation towards "better".

10. Applying sentiment analysis to Facebook messages.
There has been less work on sentiment analysis on Facebook data mainly due to various restrictions by Facebook graph api and security policies in accessing data.

11. Internationalization
Current Research work focus mainly on English content, but Twitter has many varied users from across.

4.5 APPLICATIONS OF SENTIMENT ANALYSIS:

Sentiment Analysis has many applications in various Fields.

Applications that use Reviews from Websites*:*
Today Internet has a large collection of reviews and feedbacks on almost everything. This includes product reviews, feedbacks on political issues, comments about services, etc. Thus there is a need for a sentiment analysis system that can extract sentiments about a particular product or services. It will help us to automate in provision of feedback or rating for the given product, item, etc. This would serve the needs of both the users and the vendors.

Applications as a Sub-component Technology
A sentiment predictor system can be helpful in recommender systems as well. The recommender system will not recommend items that receive a lot of negative feedback or fewer ratings.
In online communication, we come across abusive language and other negative elements. These can be detected simply by identifying a highly negative sentiment and correspondingly taking action against it.

Applications in Business Intelligence

It has been observed that people nowadays tend to look upon reviews of products which are available online before they buy them. And for many businesses, the online opinion decides the success or failure of their product. Thus, Sentiment Analysis plays an important role in businesses. Businesses also wish to extract sentiment from the online reviews in order to improve their products and in turn their reputation and help in customer satisfaction.

Applications across Domains:
Recent researches in sociology and other fields like medical, sports have also been benefitted by Sentiment Analysis that show trends in human emotions especially on social media.

Applications in Smart Homes:
Smart homes are supposed to be the technology of the future. In future entire homes would be networked and people would be able to control any part of the home using a tablet device. Recently there has been lot of research going on Internet of Things (IoT). Sentiment Analysis would also find its way in IoT. Like for example, based on the current sentiment or emotion of the user, the home could alter its ambiance to create a soothing and peaceful environment.
Sentiment Analysis can also be used in trend prediction. By tracking public views, important data regarding sales trends and customer satisfaction can be extracted.

CHAPTER :- 5
SYSTEM DESIGN AND ARCHITECTURE

5.1 Use Case Diagram

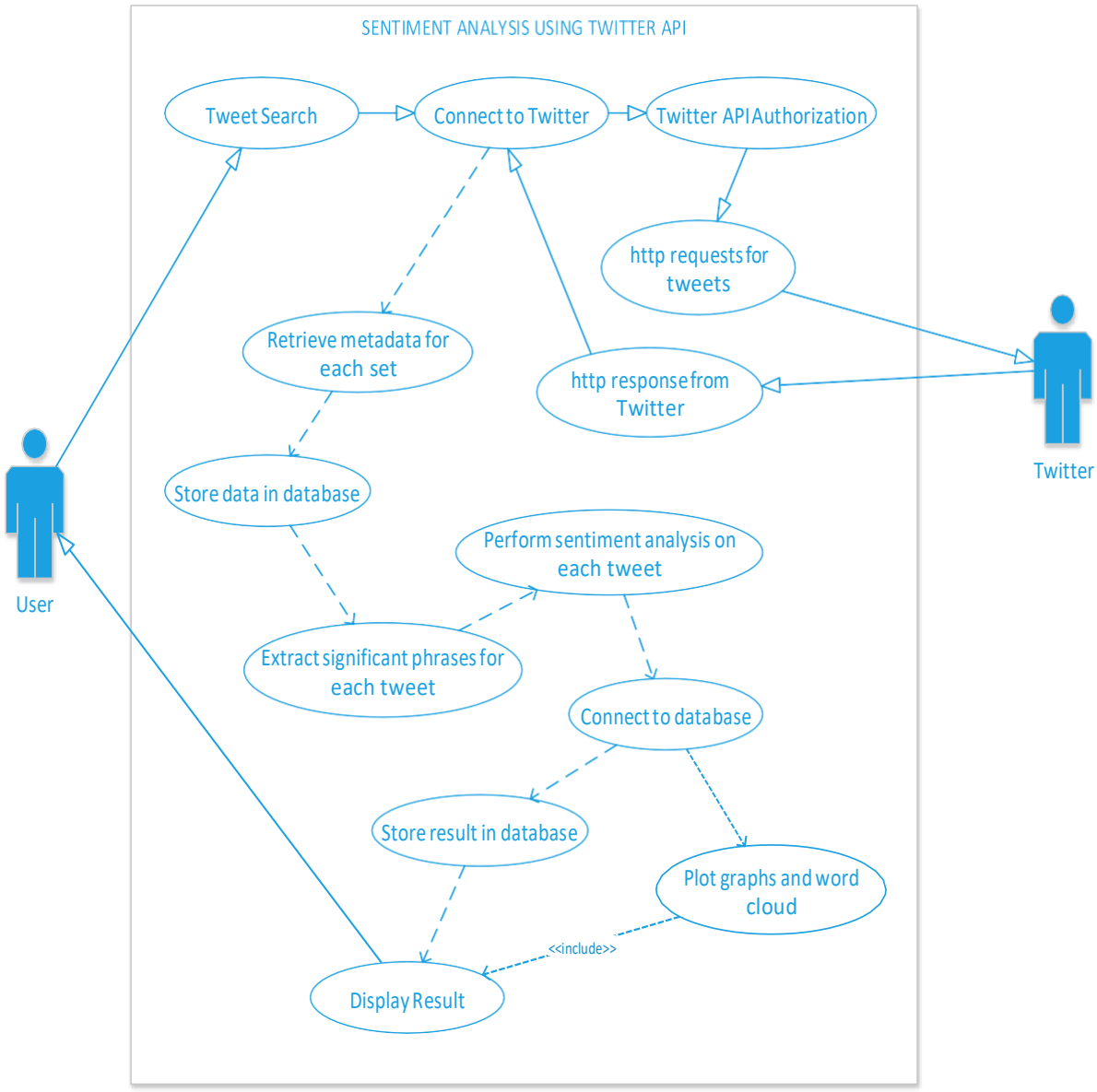

Fig 5.1 Use Case Diagram for Sentiment Analysis using Twitter API

5.2 System Flow Diagram

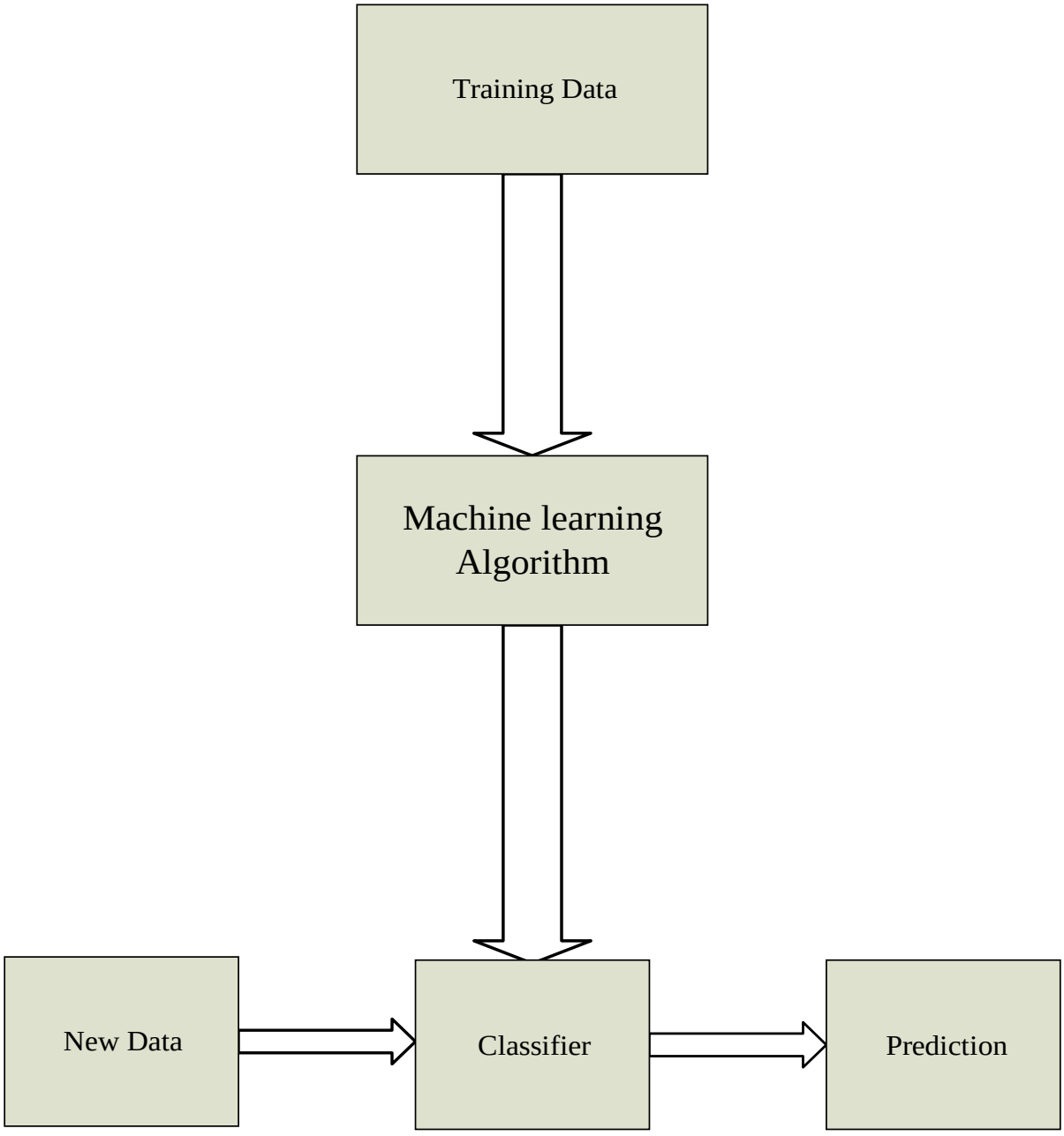

Fig 5.2(a) System Flow Diagram for Sentiment Analysis using Twitter API

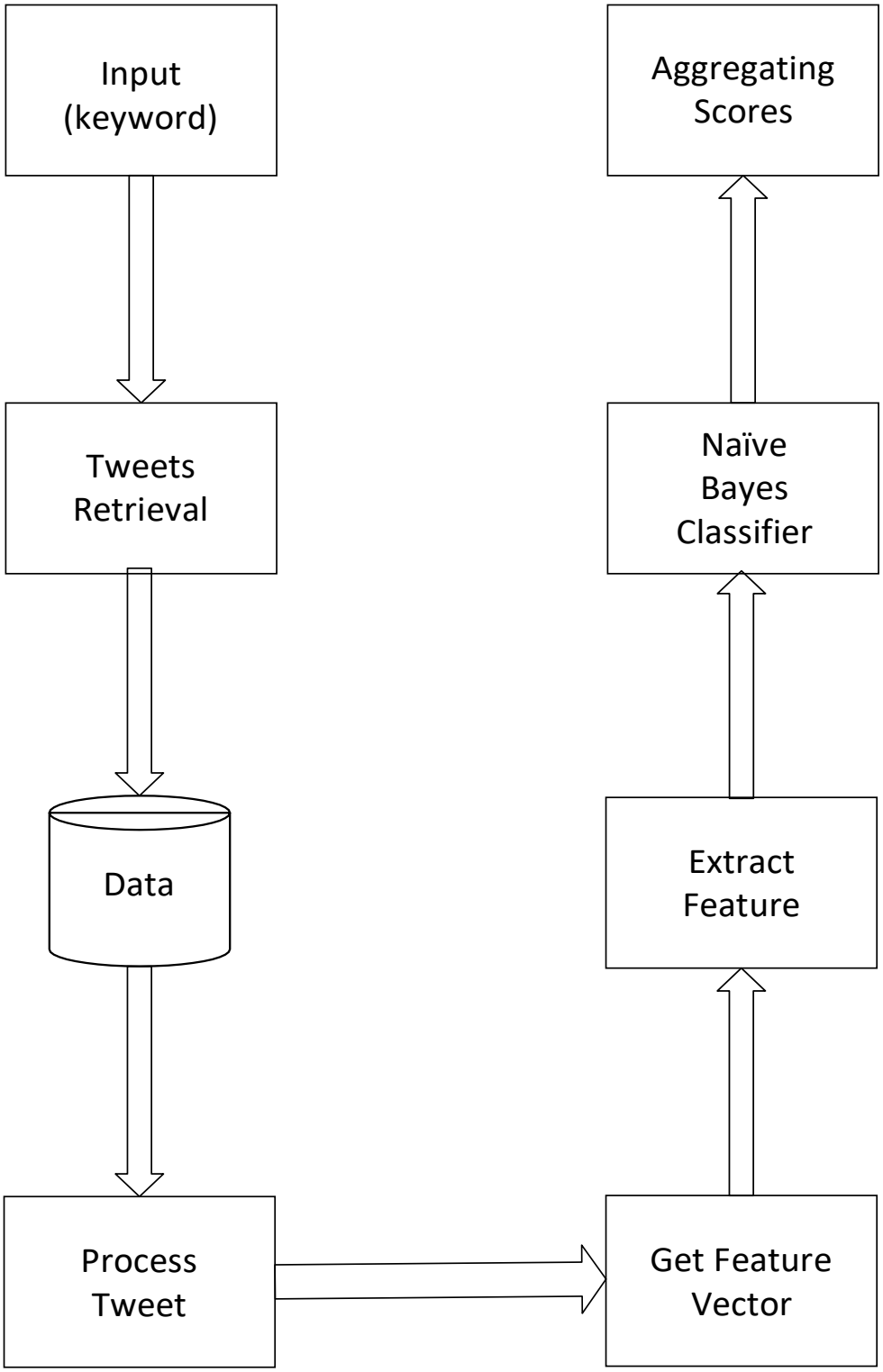

Fig 5.2(b) System Flow Diagram for Sentiment Analysis using Twitter API

5.3 Entity Relationship (ER) Diagram

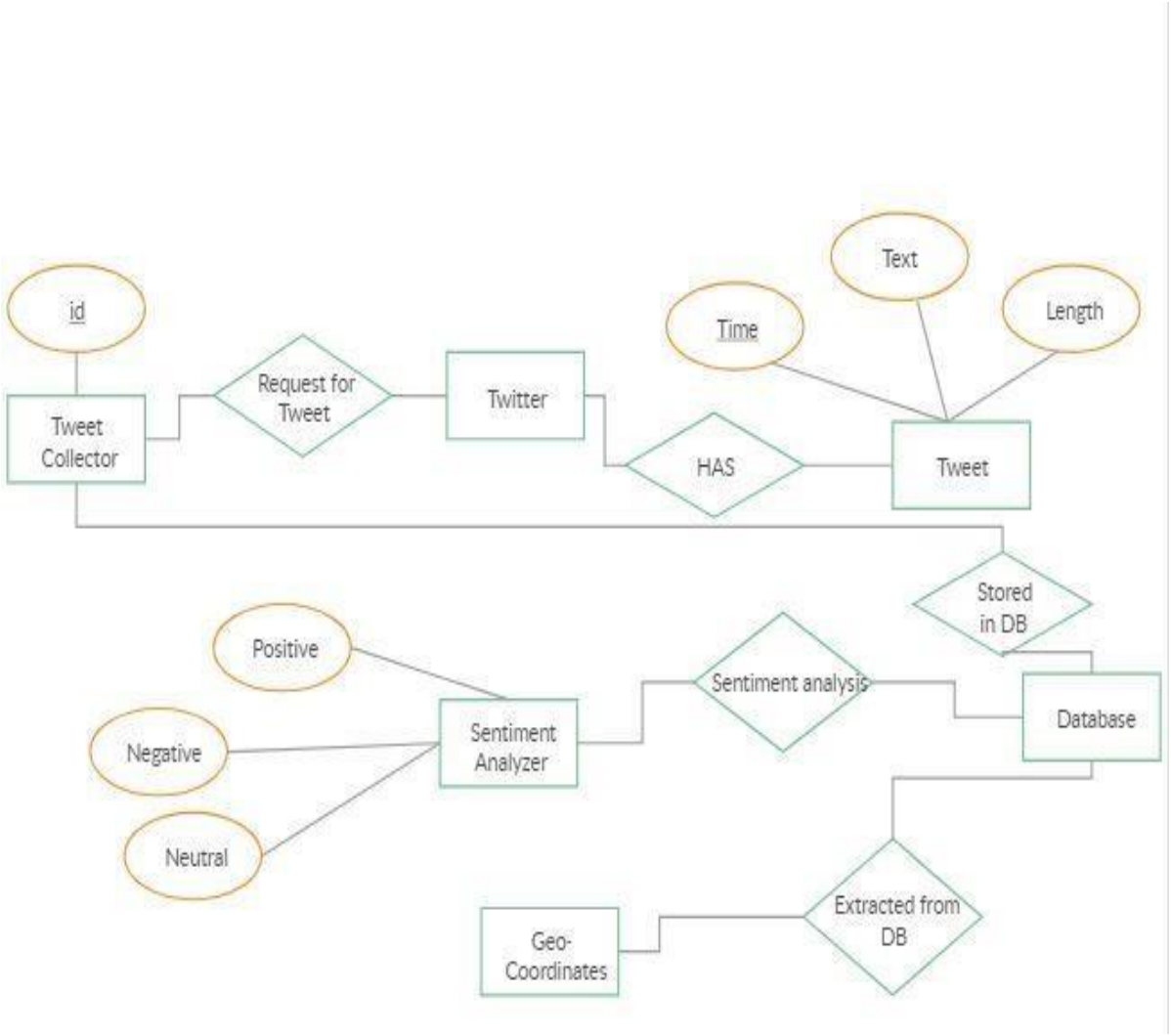

Fig 5.3 ER Diagram for Sentiment Analysis using Twitter API

5.4 Class Diagram

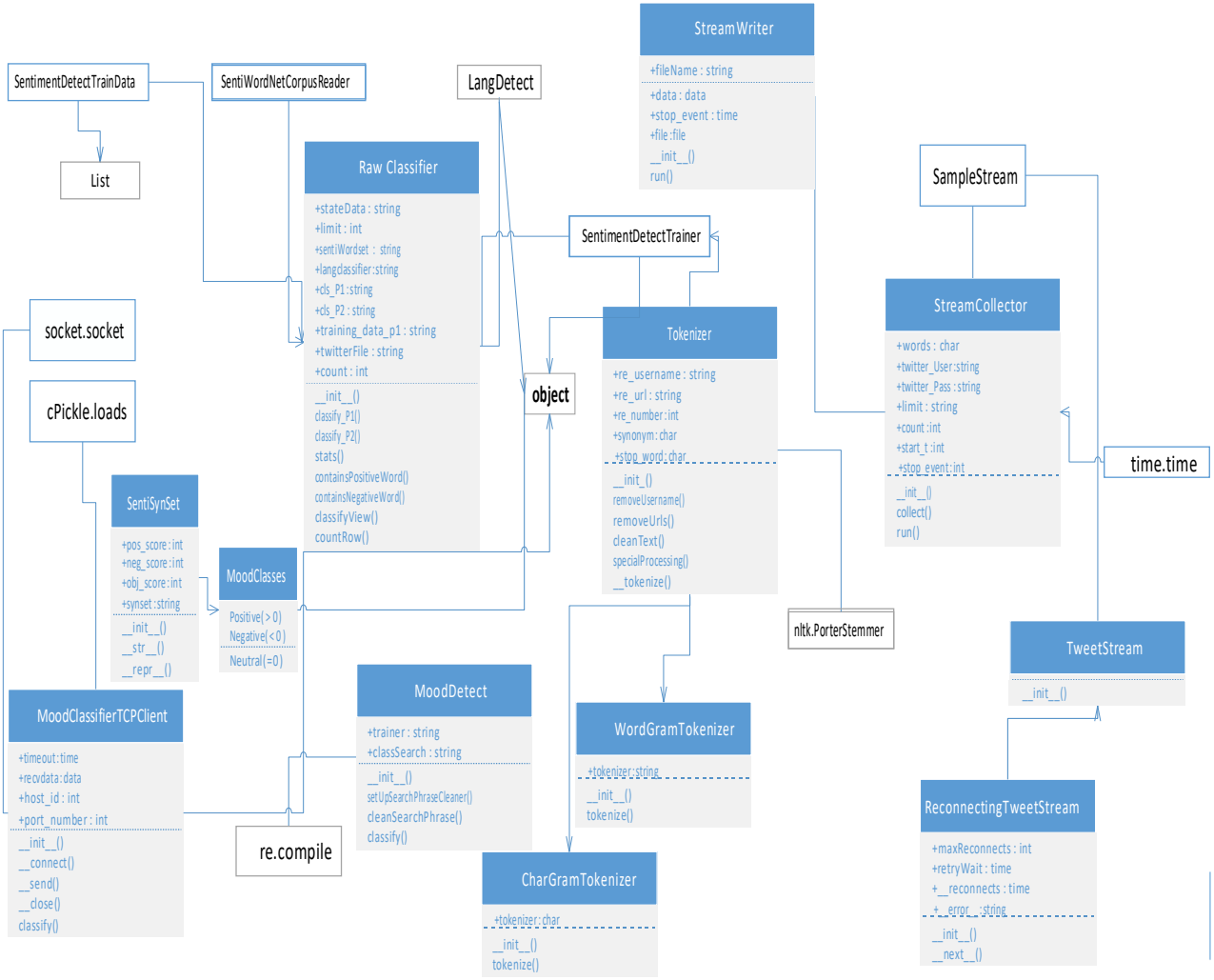

Fig 5.4 Class Diagram for Sentiment Analysis using Twitter API

5.5 Activity Diagram

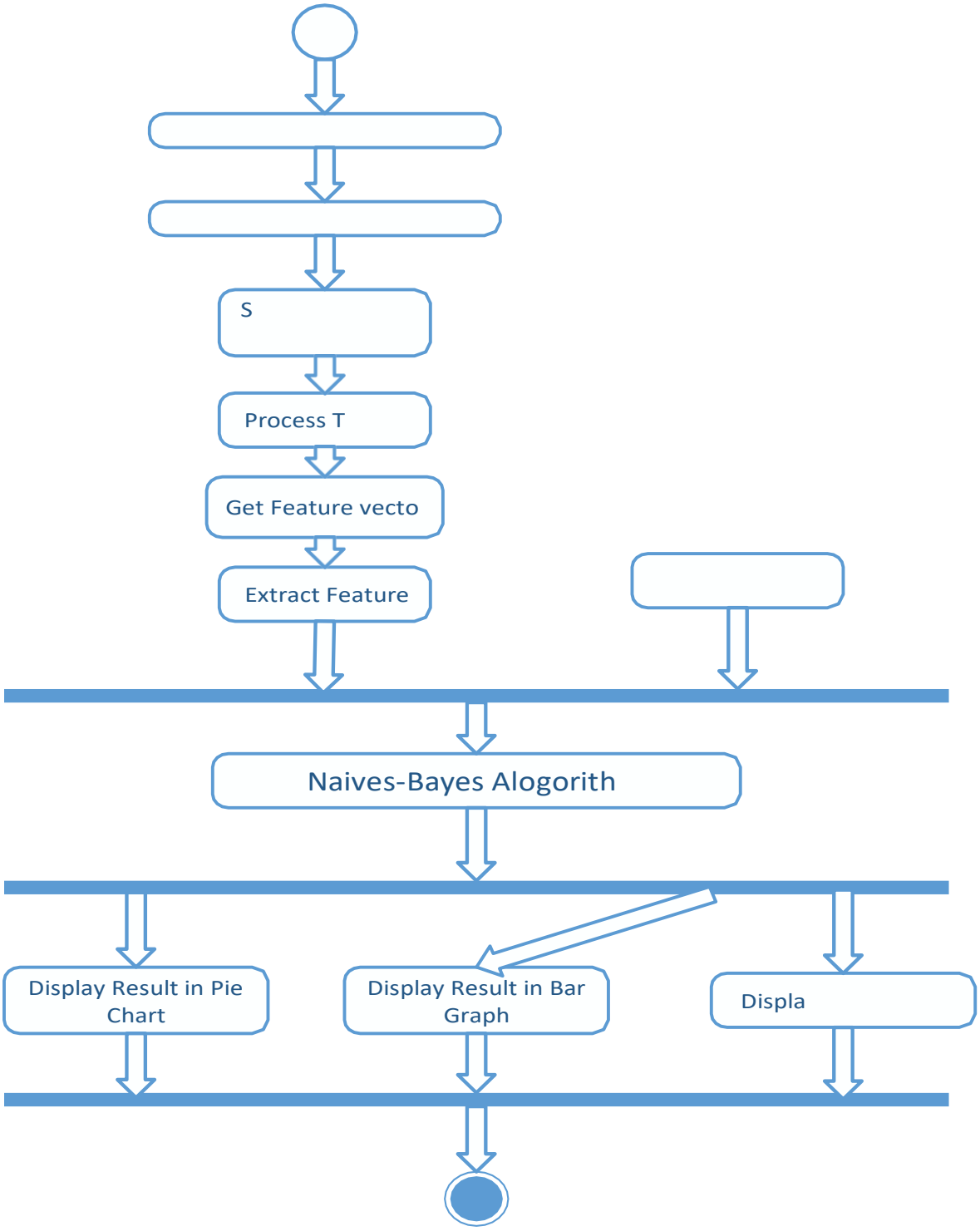

Fig 5.5 Activity Diagram for Sentiment Analysis using Twitter API

5.6 Data Flow Diagram

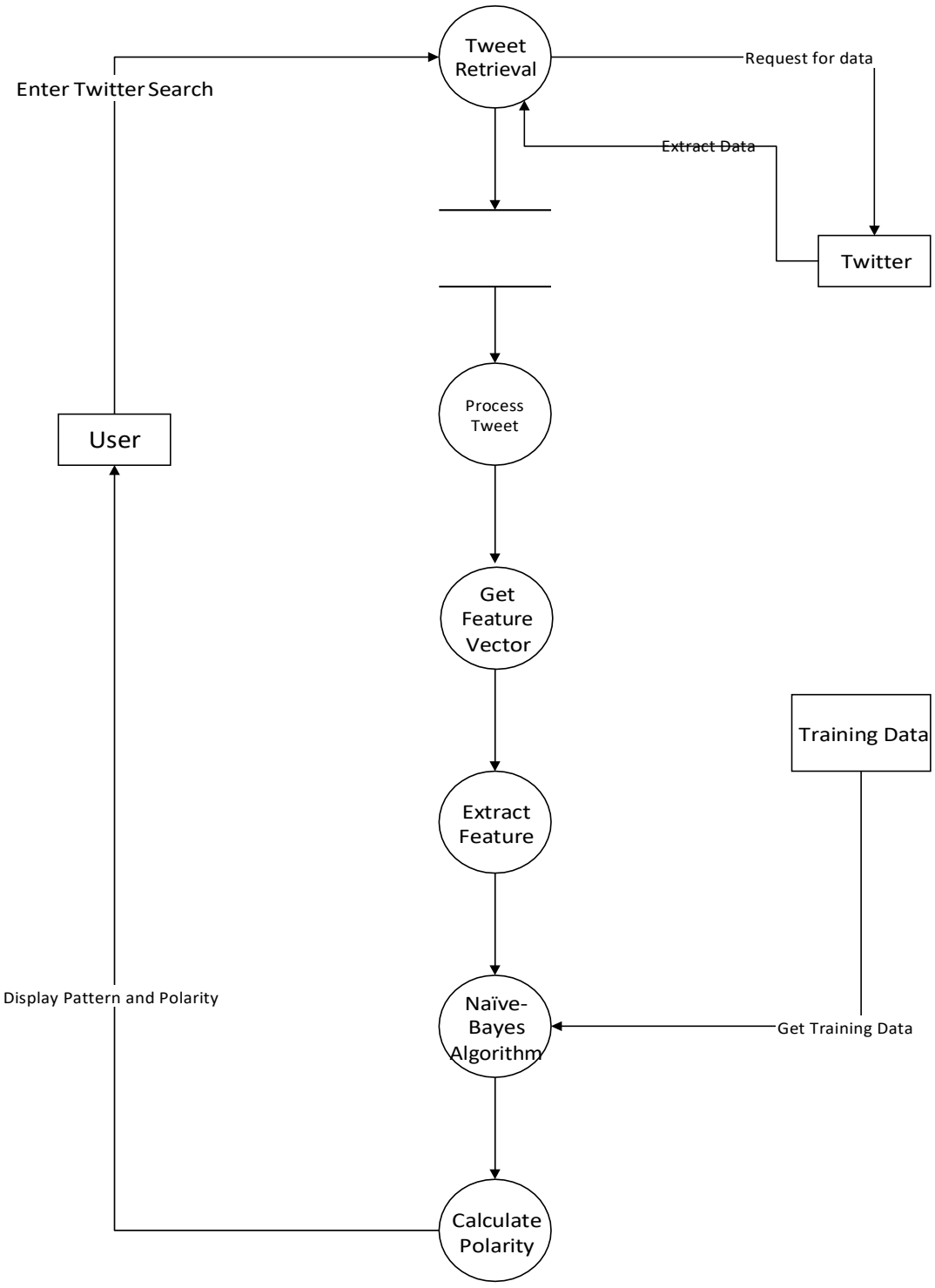

Fig 5.6 Data Flow Diagram for Sentiment Analysis using Twitter API

5.7 Sequence Diagram

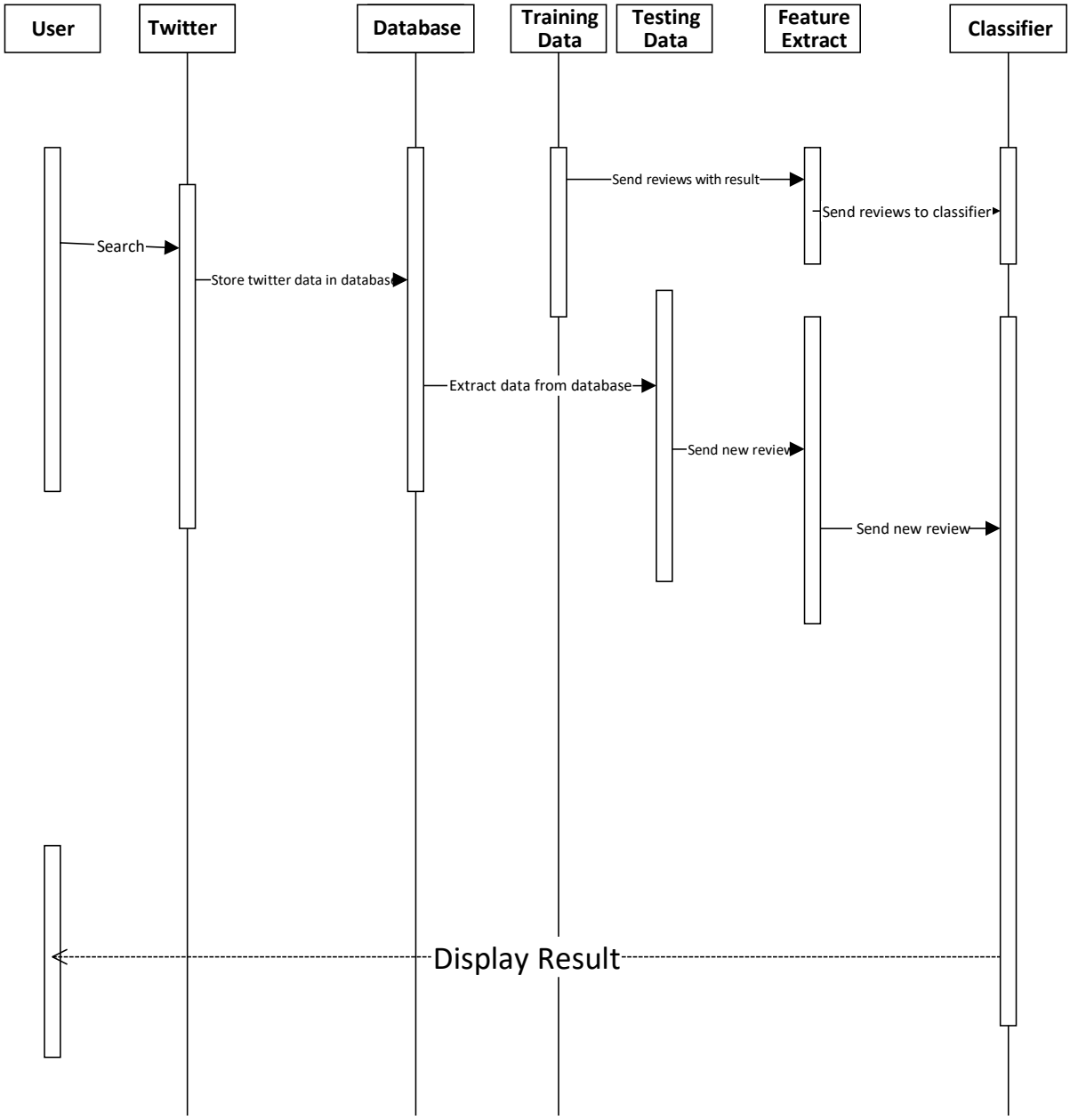

Fig 5.7 Sequence Diagram for Sentiment Analysis using Twitter API

5.8 Flow Chart

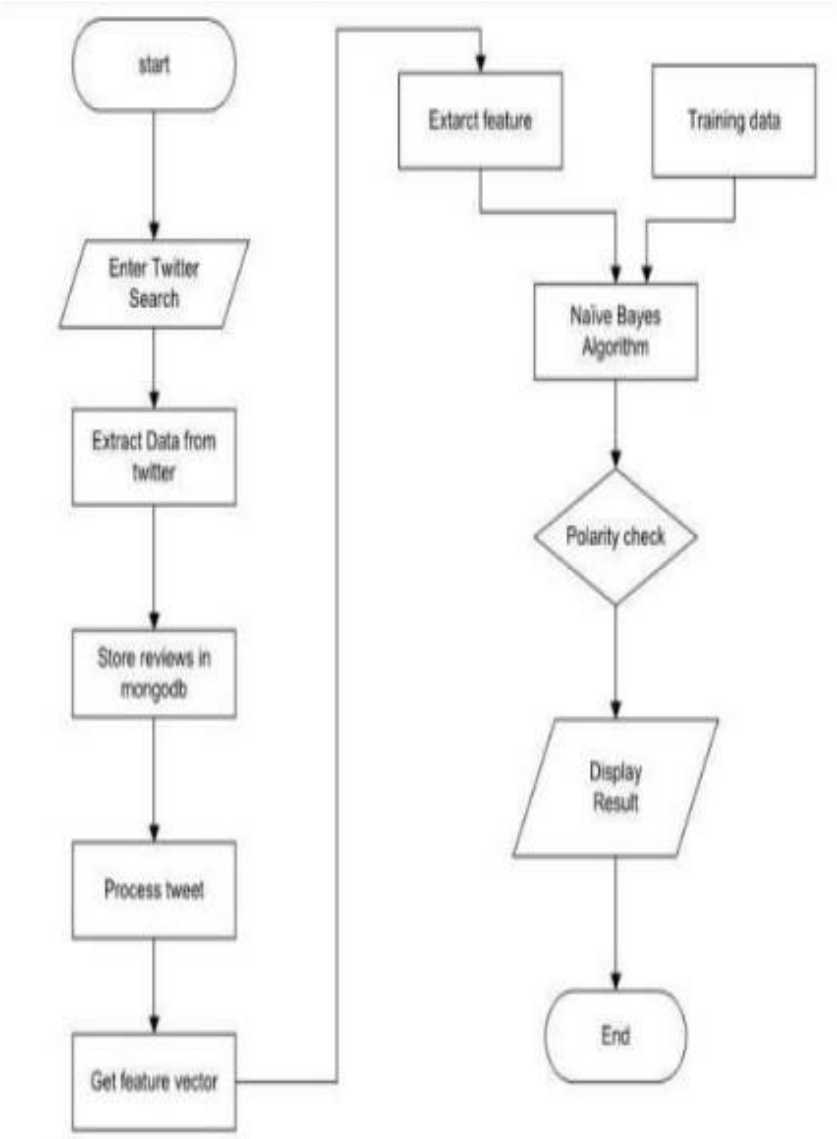

Fig 5.8 Flow Chart for Sentiment Analysis using Twitter API

CHAPTER :- 6
SYSTEM TESTING

Testing is the process of evaluating a system or its component's with the intent to find that whether it satisfies the specified requirements or not .This activity results in the actual, expected and difference between their results i.e testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements.

Testing Strategies

In order to make sure that system does not have any errors, the different levels of testing strategies that are applied at different phases of software development are

6.1. Unit Testing:

Unit testing is performed for testing modules against detailed design. Inputs to the process are usually compiled modules from the coding process. Each modules are assembled into a larger unit during the unit testing process. Testing has been performed on each phase of project design and coding. We carry out the testing of module interface to ensure the proper flow of information into and out of the program unit while testing. We make sure that the temporarily stored data maintains its integrity throughout the algorithm's execution by examining the local data structure. Finally, all error-handling paths are also tested.

6.2. Integration Testing:

The testing of combined parts of an application to determine if they function correctly together is Integration testing .This testing can be done by using two different methods

6.2.1 Top Down Integration testing

In Top-Down integration testing, the highest-level modules are tested first and then progressively lower-level modules are tested.

6.2.2 Bottom-up Integration testing

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time .When bottom level modules are tested attention turns to those on the next level that use the lower level ones they are tested individually and then linked with the previously examined lower level modules. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

6.3. System Testing:

We usually perform system testing to find errors resulting from unanticipated interaction between the sub-system and system components. Software must be tested to detect and rectify all possible errors once the source code is generated before

delivering it to the customers. For finding errors, series of test cases must be developed which ultimately uncover all the possibly existing errors. Different software techniques can be used for this process. These techniques provide systematic guidance for designing test that Exercise the internal logic of the software components, Exercise the input and output domains of a program to uncover errors in program function, behavior and performance. We test the software using two methods: White Box testing: Internal program logic is exercised using this test case design techniques. Black Box testing: Software requirements are exercised using this test case design techniques. Both techniques help in finding maximum number of errors with minimal effort and time.

6.4 Performance Testing:

It is done to test the run-time performance of the software within the context of integrated system. These tests are carried out throughout the testing process. For example, the performance of individual module is accessed during white box testing under unit testing.

6.5 Acceptance Testing

The main purpose of this Testing is to find whether application meets the intended specifications and satisfies the client's requirements. We will follow two different methods in this testing.

6.5.1 Alpha Testing

*This test is the first stage of testing and will be performed amongst the teams. Unit testing, integration testing and system testing when combined are known as alpha testing. During this phase, the following will be tested in the application:

- Spelling Mistakes.
- Broken Links.

The Application will be tested on machines with the lowest specification to test loading times and any latency problems.

6.5.2 Beta Testing

In beta testing, a sample of the intended audience tests the application and send their feedback to the project team .Getting the feedback, the project team can fix the problems before releasing the software to the actual users.

Testing Methods:

1 White Box Testing

White box testing is the detailed investigation of internal logic and structure of the Code. To perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code .The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

2 Black Box Testing

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing .The tester is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

6.6 Verification and Validation:

The testing process is a part of broader subject referring to verification and validation. We have to acknowledge the system specifications and try to meet the customer's requirements and for this sole purpose, we have to verify and validate the product to make sure everything is in place. Verification and validation are two different things. One is performed to ensure that the software correctly implements a specific functionality and other is done to ensure if the customer requirements are properly met or not by the end product. Verification is more like 'are we building the product right?' and validation is more like 'are we building the right product?'

CHAPTER :- 7

CONCLUSION & FUTURE WORK

7.1 Conclusion:

The experimental studies performed through the chapters, successfully show that hybridizing the existing machine learning analysis and lexical analysis techniques for sentiment classification yield comparatively outperforming accurate results. For all the datasets used, we recorded consistent accuracy of almost 90%.

The first method that we approached for our problem is Naïve Bayes. It is mainly based on the independence assumption .Training is very easy and fast In this approach each attribute in each class is considered separately. Testing is straightforward, calculating the conditional probabilities from the data available. One of the major task is to find the sentiment polarities which is very important in this approach to obtain desired output. In this Naïve Bayes approach we only considered the words that are available in our dataset and calculated their conditional probabilities. We have obtained successful results after applying this approach to our problem.

Clearly from the success of Hybrid Naive Bayes, it can positively be applied over other related sentiment analysis applications like financial sentiment analysis (stock market, opinion mining), customer feedback services, and etc.

7.2 Future Work

Substantial amount of work is left to be carried on, here we provide a beam of light in direction of possible future avenues of research.

- **Interpreting Sarcasm**: The proposed approach is currently incapable of interpreting sarcasm. In general sarcasm is the use of irony to mock or convey contempt, in the context of current work sarcasm transforms the polarity of an apparently positive or negative utterance into its opposite.

  This limitation Conclusion and Future Work can be overcome by exhaustive study of fundamentals in \discourse-driven sentiment analysis".

  The main goal of this approach is to empirically identify lexical and pragmatic factors that distinguish sarcastic, positive and negative usage of words.

- **Multi-lingual support**: Due to the lack of multi-lingual lexical dictionary, it iscurrent not feasible to develop a multi-language based sentiment analyser.

  Further research can be carried out in making the classifiers language independent. The authors have proposed a sentiment analysis system with support vector machines, similar approach can be applied for our system to make it language independent.

- Analysing sentiments on emoji/smileys

- Determining neutrality

- Potential improvement can be made to our data collection and analysis method

- Future research can be done with possible improvement such as more refined data and more accurate algorithm.

REFERENCES :-

- "**TWITTER SENTIMENT ANALYSIS**"
  BY :- NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

- "**SENTIMENTS ANALYSIS OF DATA COLLECTED FROM SOCIAL MEDIA FOR IMPROVING HEALTH CARE** "
  BY :- TEXAS A&M UNIVERSITY

- "**SENTIMENT ANALYSIS USECASES**"
  BY :- DR. ADAM BERMINGHAM
  INSIGHTS@DCU