VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY
Department of Computer Engineering

Project Report on

ANOMALY BASED FORENSIC ANALYSIS OF SOCIAL MEDIA

In partial fulfillment of the Fourth Year, Bachelor of Engineering (B.E.) Degree in Computer Engineering at the University of Mumbai Academic Year 2017-2018

Submitted by
Mohit Ahuja (D17A , Roll no 02)
Dipika Jiandani (D17A , Roll no 35)
Riddhi Karkera (D17A, Roll no 37)
Megha Manglani (D17A, Roll no 45)

Project Mentor
Mrs. Abha Tewari

(2017-18)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Department of Computer Engineering

Certificate

This is to certify that ***Mohit Ahuja, Dipika Jiandani, Riddhi Karkera, Megha Manglani*** of Fourth Year Computer Engineering studying under the University of Mumbai have satisfactorily completed the project on "***Anomaly based forensic analysis of social media***" as a part of their coursework of PROJECT-II for Semester-VIII under the guidance of their mentor ***Prof. Abha Tewari*** in the year 2017-2018 .

This thesis/dissertation/project report entitled ***Anomaly based forensic analysis of social media*** by ***Mohit Ahuja, Dipika Jiandani, Riddhi Karkera, Megha Manglani*** is approved for the degree of ***Computer Engineering.***

Programme Outcomes	Grade
PO1,PO2,PO3,PO4,PO5,PO6,PO7, PO8, PO9, PO10, PO11, PO12 PSO1, PSO2	

Date:

Project Guide: Mrs. Abha Tewari

Project Report Approval
For
B. E (Computer Engineering)

This thesis/dissertation/project report entitled ***Anomaly based forensic analysis of social media*** by ***Mohit Ahuja, Dipika Jiandani, Riddhi Karkera, Megha Manglani*** is approved for the degree of ***Computer Engineering.***

Internal Examiner

External Examiner

Head of the Department

Principal

Date:
Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

--	--
(Signature)	(Signature)
--	--
(Mohit Ahuja D17A 02)	(Dipika Jiandani D17A 35)
--	--
(Signature)	(Signature)
--	--
(Riddhi Karkera D17A 37)	(Megha Manglani D17A 45)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Mrs. Abha Tewari** (Project Guide) for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair ,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement at several times.

Computer Engineering Department
COURSE OUTCOMES FOR B.E PROJECT

Learners will be to,

Course Outcome	Description of the Course Outcome
CO 1	Able to apply the relevant engineering concepts, knowledge and skills towards the project.
CO2	Able to identify, formulate and interpret the various relevant research papers and to determine the problem.
CO 3	Able to apply the engineering concepts towards designing solution for the problem.
CO 4	Able to interpret the data and datasets to be utilized.
CO 5	Able to create, select and apply appropriate technologies, techniques, resources and tools for the project.
CO 6	Able to apply ethical, professional policies and principles towards societal, environmental, safety and cultural benefit.
CO 7	Able to function effectively as an individual, and as a member of a team, allocating roles with clear lines of responsibility and accountability.
CO 8	Able to write effective reports, design documents and make effective presentations.
CO 9	Able to apply engineering and management principles to the project as a team member.
CO 10	Able to apply the project domain knowledge to sharpen one's competency.
CO 11	Able to develop professional, presentational, balanced and structured approach towards project development.
CO 12	Able to adopt skills, languages, environment and platforms for creating innovative solutions for the project.

Abstract

Offensive language on social media has unfortunately become a common occurrence among users. When offensive language is detected in a user message, a problem arises about how the offensive language should be removed, i.e. the offensive language filtering problem. To solve this problem, manual filtering approach is known to produce the best filtering result. However, manual filtering is costly in time and labour thus cannot be widely applied. The motive is to detect offensive language in a user message, post or comment and take necessary actions for the same. This is called as offensive language filtering.

In this project we provide a comparison of different algorithms to build a solution through which Facebook users can find their cyber bullies and report them. The entire process consists of six stages: data collection, pre-processing, sessionization, ground truth, feature extraction and classification. Using machine learning algorithms for pre-processing and classification of the data and tools like Facepager and PyCharm, we have evaluated the processing, usage and accuracy of three major classification algorithms which are Naive Bayes, Support Vector Machine and Neural Networks.

Index

List of Figures

List of tables

CHAPTER 1: INTRODUCTION

1.1 Motivation

Cyberbullying is defined as an aggressive, intentional act carried out by a group or individual, using electronic forms of contact repeatedly and over time against a victim who cannot easily defend him or herself. One of the most common forms is the posting of hateful comments about someone in social net-works. To the online community, the spread of offensive language undermines its reputation, drives users away, and even directly affects its growth. To users, viewing offensive language brings negative influence to their mental health, especially for children and youth. When offensive language is detected in a user message, a problem arises about how the offensive language should be removed, i.e. the offensive language filtering problem. To solve this problem, manual filtering approach is known to produce the best filtering result. However, manual filtering is costly in time and labor thus cannot be widely applied. This project focuses on conducting forensic analyses on some of the widely used social networking applications like Facebook, Instagram to name a few. The offensive language has been analyzed in text messages posted in online communities, and a new automatic sentence-level filtering approach has been proposed that is able to semantically remove the offensive language by utilizing the grammatical relations among words.

1.2 Problem definition

This project focuses on conducting forensic analyses on some of the widely used social networking applications like Facebook Instagram to name a few. This analysis will be aimed at analyzing offensive comments with the motivation of cyber-bully in posted on these applications and backtracking them to the offender. The extent, significance, and intention of the data that could be found and retrieved. If so, the suspect will be found guilty of a cybercrime since there will be a solid evidence to prove the activity was performed by him. This application includes pre-processing and analysis of data via various models of Machine Learning like Naive Bayes, Support Vector Machine and Artificial Neural Networks. The development of such an application has the ability to stop and reduce the rate of subjugating that has been happening online at a full-fledged rate. The goal is to compare three classification models used in the development of a web application that will proactively detect and report cases of cyber-bullying and personal security intrusion on social media platforms like Facebook, Twitter, etc. using the concepts of behavioural analysis and machine learning. Further, a block action or report will be generated on the basis of the supporting evidence found through Forensic Analysis of social media sites.

1.3 Relevance of the Project

A study estimated that the number of users of online social networks worldwide was about 2.2 billion monthly active users (International Telecommunications Union, 2016).With hundreds of millions social network users worldwide, forensic data extraction on social networks has become an important research problem. Academia and law enforcement alike have shown a strong demand for data that is collected from online social networks. There is a lot of identity theft, theft of personal data, public defamation, cyber stalking, bullying and other criminal activities on social network sites. In many cases, the privacy settings of these apps may allow anybody to post offensive content on any individuals portal, regardless of his relationship to the user. But due to the immense advancement in technology and data science, potential evidence can be held on these devices and recovered with the right tools and examination methods. This project focuses on conducting forensic analyses on one of the widely used social networking application-Facebook.

1.4 Methodology used:

Our approach to detect aggressive and bullying behavior on Facebook involves the following steps: (1) data collection, (2) preprocessing of tweets, (3) sessionization, (4) ground truth building, (5) extracting user-, text-, and network-level features, (6) user modeling and characterization, and (7) classification.

First, a tool called “Facepager” is used to get the data from Facebook. It gives access to different posts, pictures, comments and emoticons of various public profiles on Facebook using which the training and testing data are formed. With the help of preprocessing algorithms, the raw data is converted into executable form. The pre-processed data is then classified using the machine learning algorithms- naive Bayes, support vector machines and neural networks for sentiment analysis. A comparison of these three classification algorithms is done on the basis of the processing, performance and the accuracy of each of the them. The polarity and subjectivity of each algorithm is found and plotted on a graph to compare. Also, based on the frequency of the bad words, a word cloud is generated. The bad word which has the highest frequency will have a bigger size compared to the words whose frequency is less.

CHAPTER 2:
LITERATURE SURVEY

2.1 Papers

2.1.1 Forensic investigation of social networking applications.

Social networking applications such as Facebook, LinkedIn, MySpace and Twitter provide facilities including email, blogging, instant messaging and photo sharing for social and commercial exchange. There has been a rapid growth in the use of social networking applications by both individuals and organisations. And an increasing number of organisations use Facebook and Twitter as part of their marketing campaigns.

2.1.2 Using Naïve Bayes Algorithm in detection of Hate Tweets.

Social Media has become a very powerful tool for information exchange as it allows users to not only consume information but also share and discuss various aspects of their interest. Nevertheless, online social platforms are beset with hateful speech - content that expresses hatred for a person or group of people. Such content can frighten, intimidate, or silence platform users, and some of it can incite other users to commit violence. The main goal of this study is to develop a reliable tool for detection of hate tweets. This paper develops an approach for detecting and classifying hateful speech that uses content produced by self-identifying hateful communities from Twitter.

2.1.3 Sentiment analysis for hate speech detection on social media

Monitoring hate content in traditional mainstream media such as radio and television, is much easier than monitoring online hate speech content such as social media and microblogging sites. This is largely due to the fact that social media consists of a large amount of user generated content that would need to be monitored.

2.1.4 Analysis of Various Sentiment Classification Techniques

Sentiment analysis is an ongoing research area in the field of text mining. People post their review in form of unstructured data so opinion extraction provides overall opinion of reviews so it does best job for customer, people, organization etc. The main aim of this paper is to find out approaches that generate output with good accuracy. This paper presents recent updates on papers related to classification of sentiment analysis of implemented various approaches and algorithms.

The main contribution of this paper is to give idea about that careful feature selection and existing classification approaches can give better accuracy.

2.1.5 Sentiment Classification using Machine Learning Techniques

The discussion forum, review sites, blogs are some of the opinion rich resources where review or posted articles is their sentiment, or overall opinion towards the subject matter. In this study, sentiment classification techniques were applied to movie reviews. Specifically, we compared two supervised machine learning approaches SVM, Naive Bayes for Sentiment Classification of Reviews. Results states that Naïve Bayes approach outperformed the svm. If the training dataset had a large number of reviews, Naive Bayes approach reached high accuracies as compare to other.

2.1.6 Using Machine Learning Techniques for Sentiment Analysis

The Natural language processing is the discipline that studies how to make the machines read and interpret the language that the people use, the natural language. But in the machines world, the words not exist, and they are represented by sequences of numbers that the machine represents with a character when displaying them on screen. The Sentiment Analysis is the name of the problem that with a sentence or text the machine gets capable to analyze and predict with the maximum precision possible the sentiment that will be obtained by a person when reads it or the contextual opinion related to something. This document wants to show what we can obtain using the most used machine learning tools.

2.1.7 Social Media Sentiment Analysis using Machine Learning Classifiers

Comparison of different machine learning techniques applied to the case of sentiment analysis in social media. Several machine learning methods were used during experimentation session: Maximum Entropy, Naive Bayes and Support Vector Machines we tried to compare different techniques for preprocessing Social media data and find those ones which impact on the building accurate classifiers. We use Twitter, an online social networking and micro-blogging tweets facility, which user can update related post in the form of content type is text. We develop an automated system which extracts the sentiments from the online posts from twitter. Our system shows sentiment identification, which expresses opinion associated with each entity Finally, we have leveraged Naive Bayes, Support vector machine and Maximum Entropy Classifiers Algorithms to do the sentiment analysis on this myriad of data.

CHAPTER 3: REQUIREMENTS

3.1 Functional Requirements:

- It would serve as an anti-bullying app for Facebook.
- Easy detection of cybercrimes/bullying
- Provide a platform for the user to analyze and generate reports of all of his social media accounts
- It would help in easy detection of spammers, stalkers and other cyber bullies.
- The comments, direct messages, likes, followers- all of these would be tapped to generate a report every month/whenever required.
- The report would give information about the time at which there is maximum interaction, growth/depreciation in the followers, likes, etc.
- Prevent identity thefts
- Behavioral analysis

3.2 Non-Functional Requirements

- Distinct well-defined sentiment classification.
- Opinionated word extraction from reviews should be retrieved as quickly as possible.
- Trust score should be secure.
- Good user interface for enhanced user experience.
- System should be as automated as possible.

3.3 Constraints:

- If the harassment or bullying is via personal messages, our tool will not be able to reveal the identity of that person because that data keeps on changing dynamically and our tool doesn't support that feature as of now.
- Also, crowdsourcing of data will be a cumbersome task to deal at this moment.

3.4 Hardware & Software Requirements

3.4.1 Hardware Requirements:

- The minimum hardware topology for a small dataset scenario is a single–server deployment that contains the following three tiers: SQL Server, Application server, Front-end web server
- Front-end web and Application server hardware recommendations: Processor:64-bit, four-core, 2.5 GHz minimum per core
- RAM:8 GB for developer or evaluation use, 16 GB for production use
- Hard disk: 80 GB

3.4.1 Software Requirements:

- PyCharm (Python 3.6.5)
- Internet connection must be established (preferably at least 10 Mbps).
- SQLite environment as DBMS.
- XAMPP support
- Facepager

CHAPTER 4:

PROPOSED DESIGN

4.1 System Block Diagram:

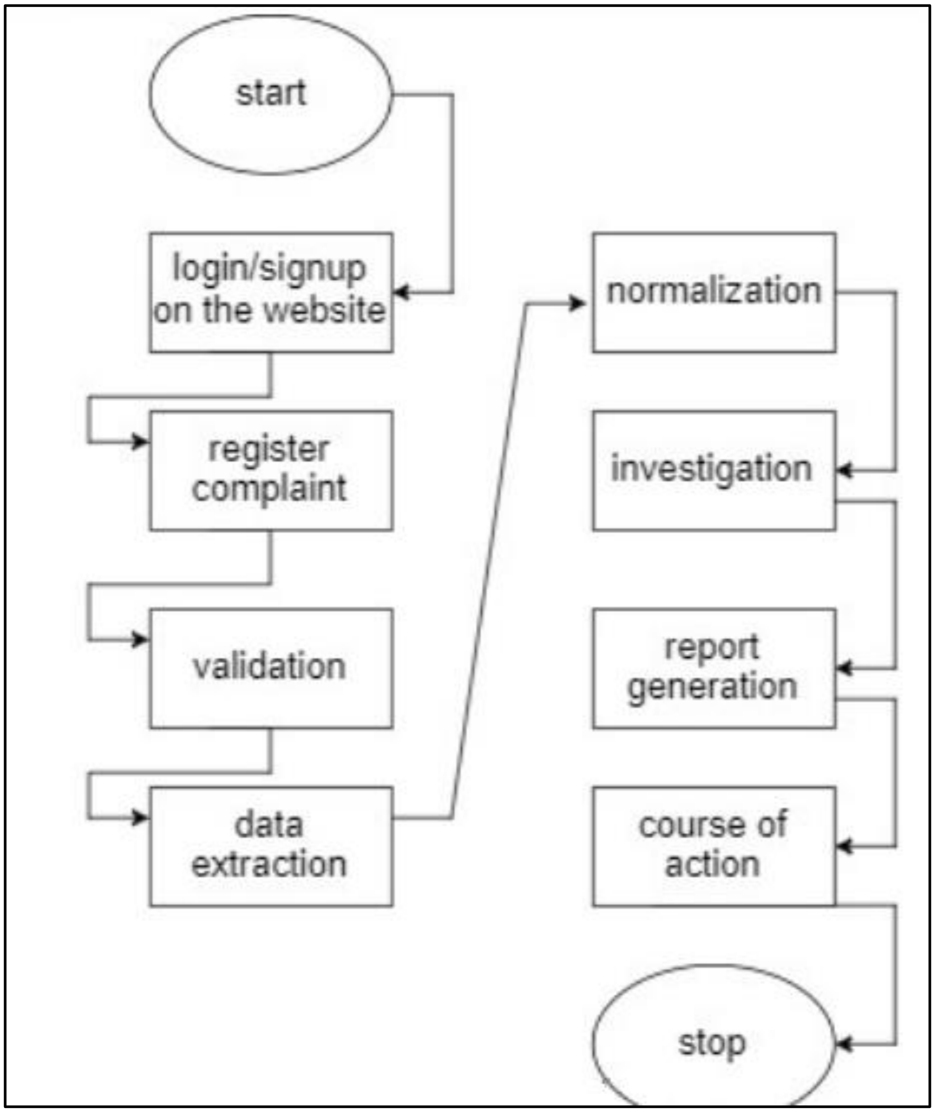


Figure 4.1 - System block diagram

Figure 4.1 shows the block diagram of our proposed system.

Data Collection. Our 1st step is to collect posts and, naturally, there are a few possible ways to do so. In this paper, we rely on Facebook's Streaming API, which provides free access to about 10%.

Preprocessing. Next, we remove stop words, URLs, and punctuation marks from the tweet text and perform normalization – i.e., we eliminate repetitive characters; e.g., the word "yessss" is converted to "yes".

Sessionization. Since analyzing single comments does not provide enough context to discern if a user is behaving in an aggressive or bullying way, we group comments from the same user, based on time clusters, into sessions and analyze them instead of single comments.

Ground Truth. We build ground truth (needed for machine learning classification, explained next) using human annotators. For this we use a crowdsourced approach, by recruiting workers who are provided with a set of comments from a user, and are asked to classify them according to predefined labels. If such an annotated dataset is already available, this step can be omitted.

Feature Extraction. We extract features from both tweets and user profiles. More specifically, features can be user-, text-, or network-based; e.g., the number of followers, tweets, hashtags, etc.

Classification. The nal step is to perform classification using the (extracted) features and the ground truth. Naturally, different machine learning techniques can be used for this task, including probabilistic classifiers (e.g., Naïve Bayes), decision trees (e.g., J48), ensembles (e.g., Random Forests), or neural networks.

Scalability and Online Updates. An important challenge to address is supporting scalable analysis of large tweet corpora

4.2 System Design / Conceptual Design (Architectural)

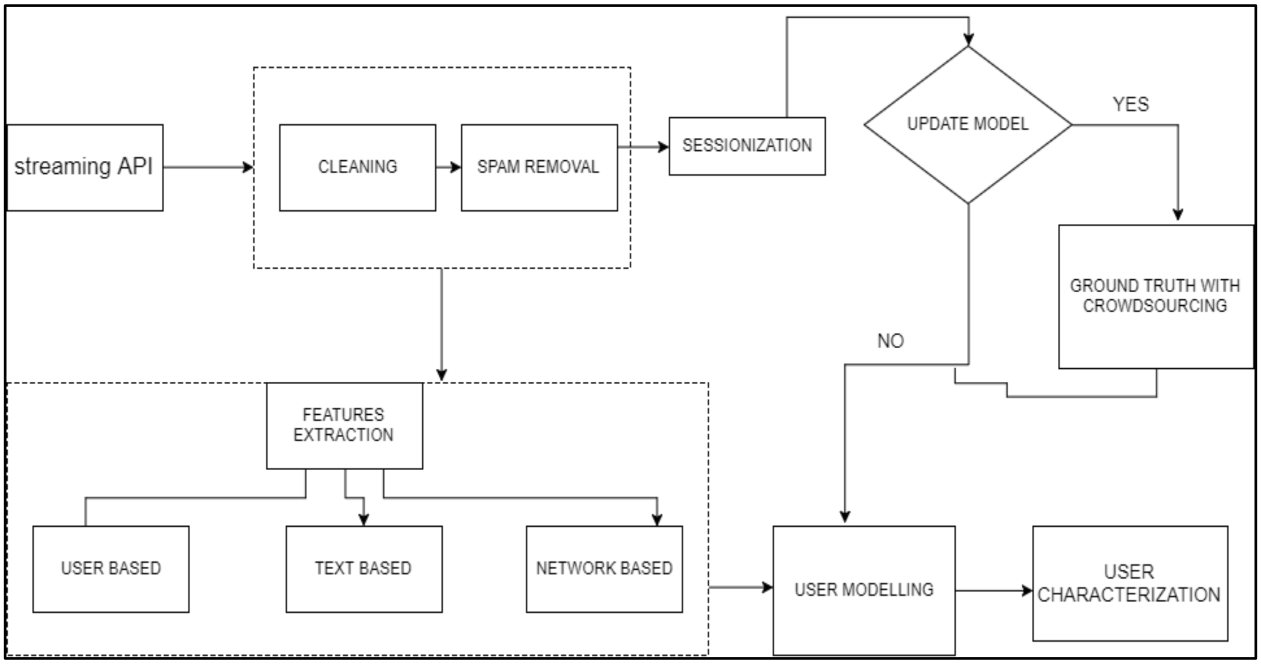


Figure 4.2: conceptual diagram

4.3 Detailed Design (DFD, Flowchart, State Transition Diagram, ER Diagram)

4.3.1 DFD Level 0:

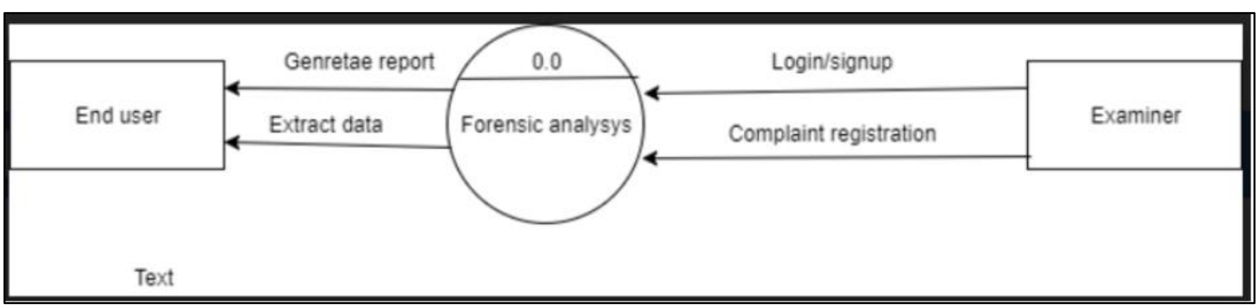


figure 4.3.1 - Level 0 DFD

4.3.2 DFD Level 1:

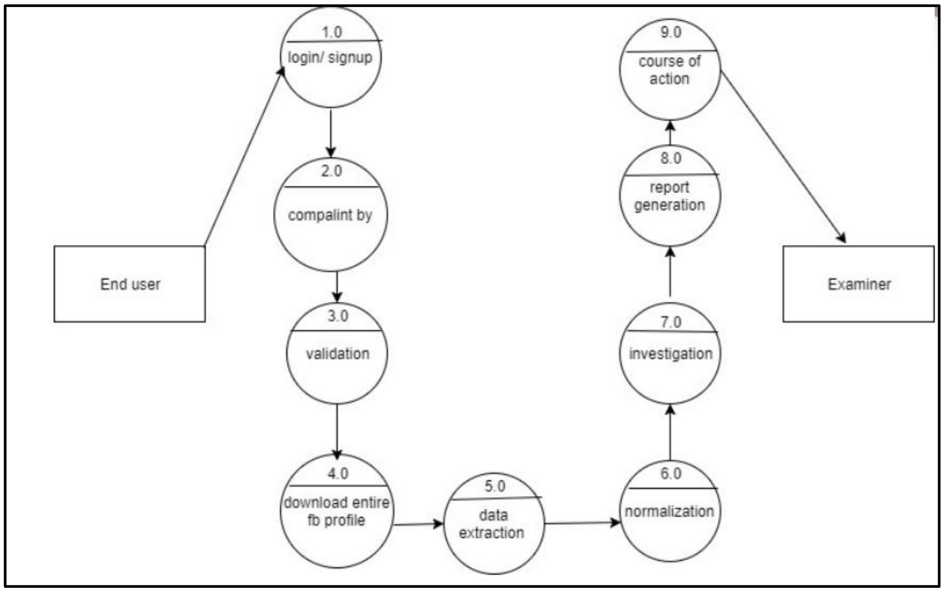


figure 4.3.2: Login/Signup Level 1 DFD

4.3.3 Authentication and authorization

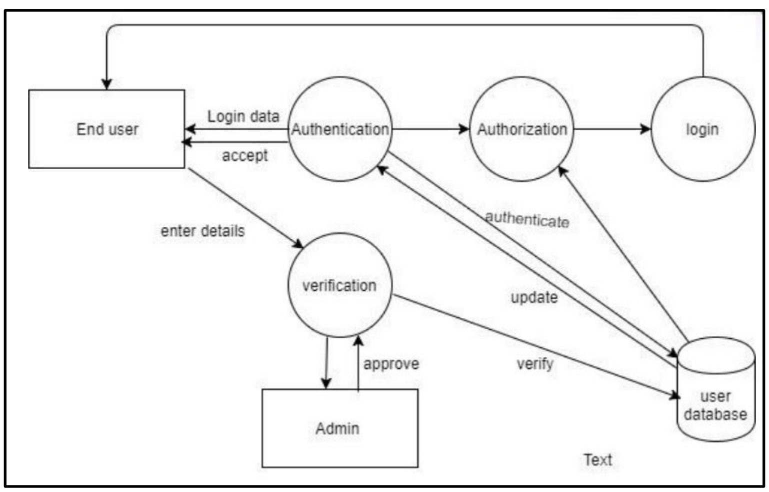


figure 4.3.3 - Authentication and authorization Level 1 DFD

4.3.4 Complaint By:

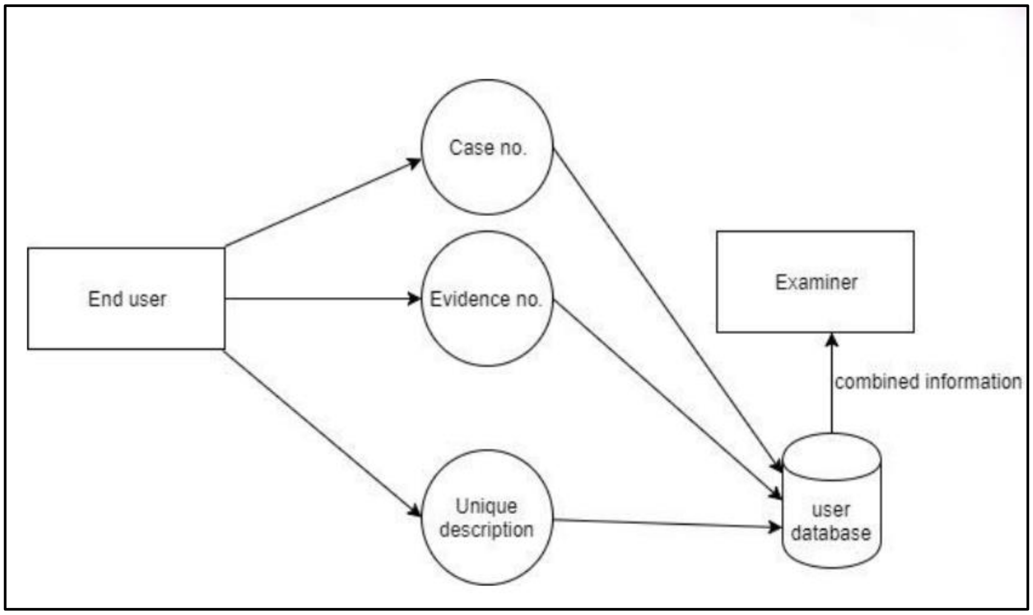


figure 4.3.4- Complaint by Level 1 DFD

4.3.5 Validation:

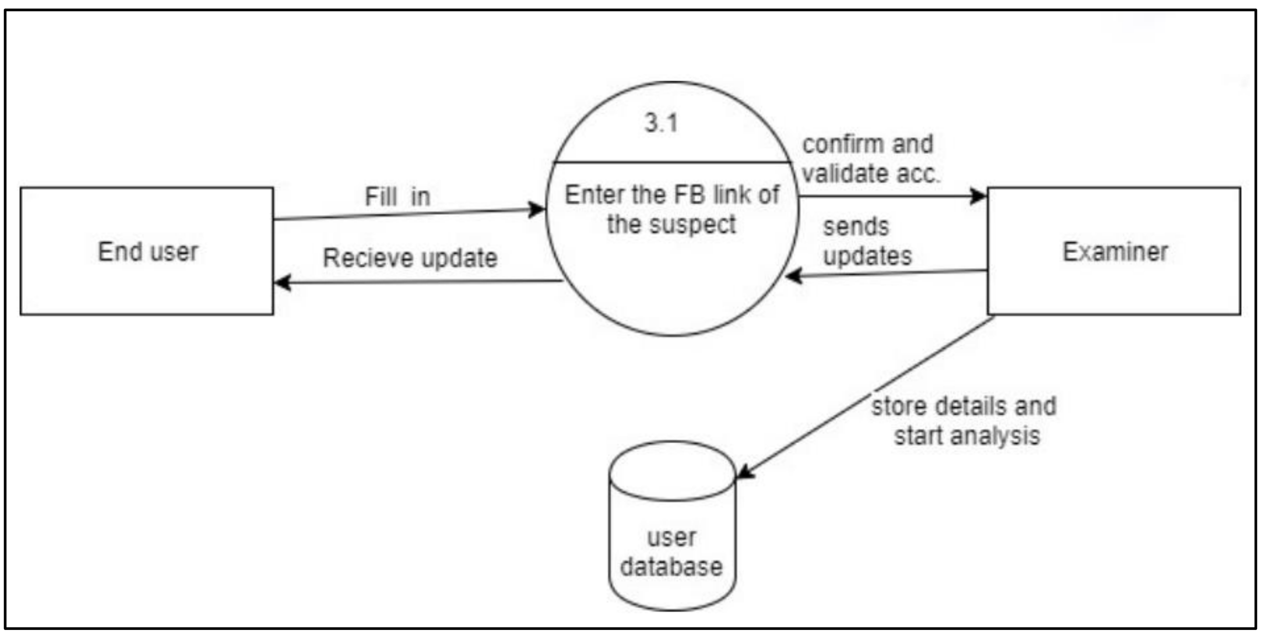


figure 4.3.5 - Validation Level 1 DFD

4.3.6 Data Extraction:

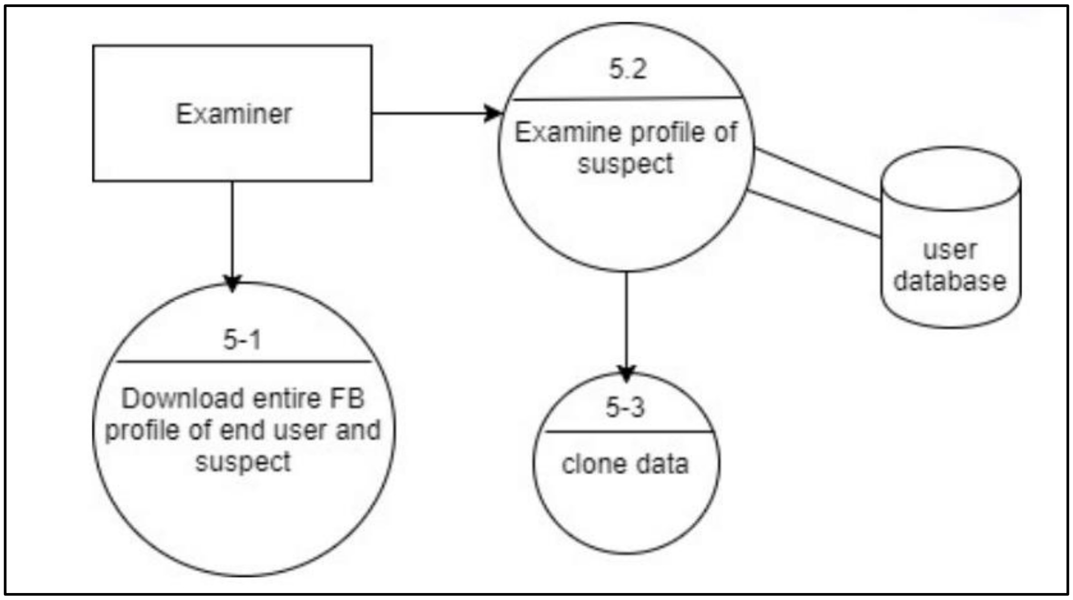


Figure 4.3.6 - Data Extraction Level 1 DFD

4.3.7 Normalization:

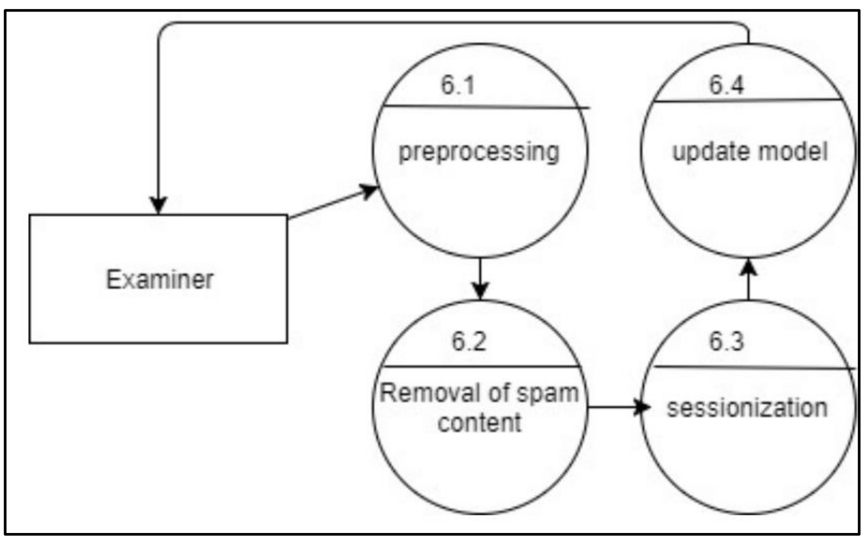


figure 4.3.7 - Normalization Level 1 DFD

4.3.8 Investigation:

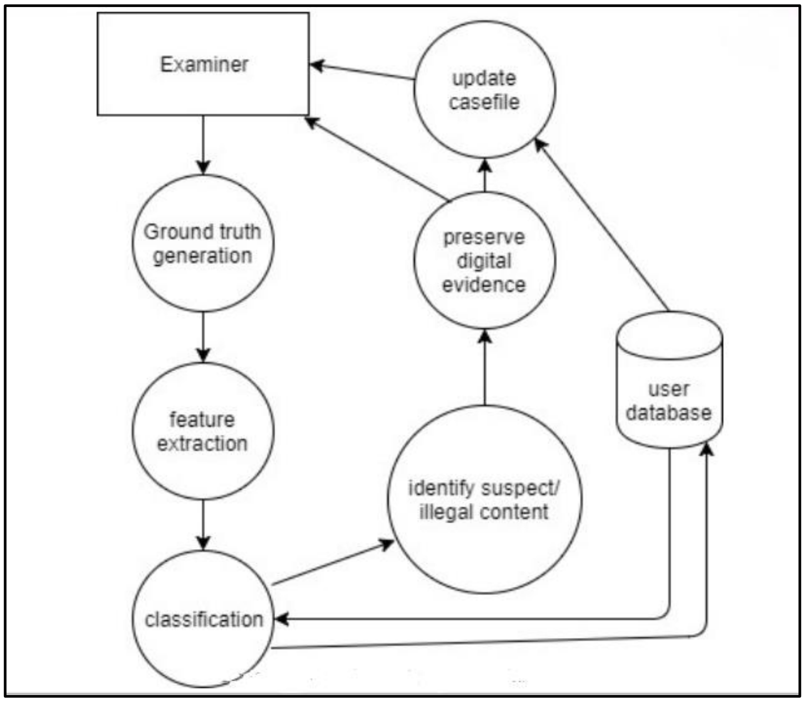


figure 4.3.8- Investigation Level 1 DFD

4.3.9 Report generation:

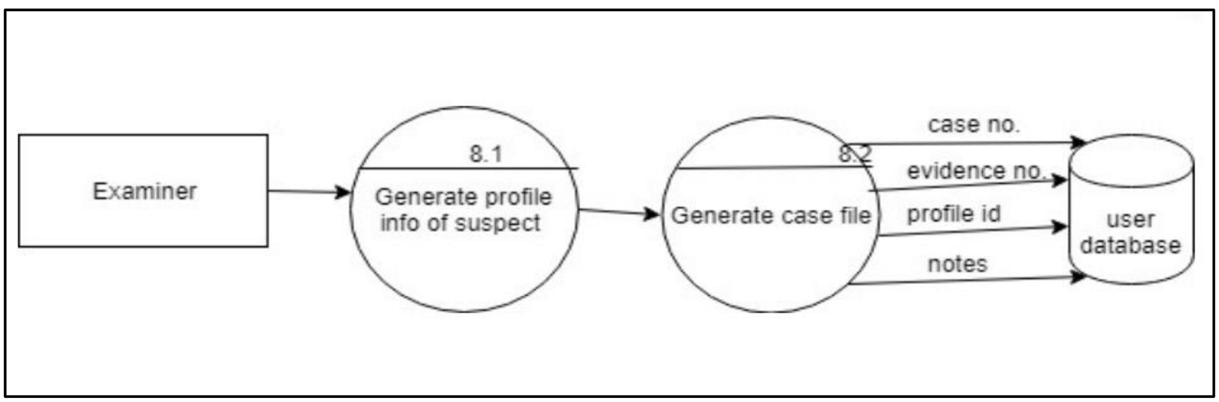


figure 4.3.9- Report Generation Level 1 DFD

4.3.10 Course of action:

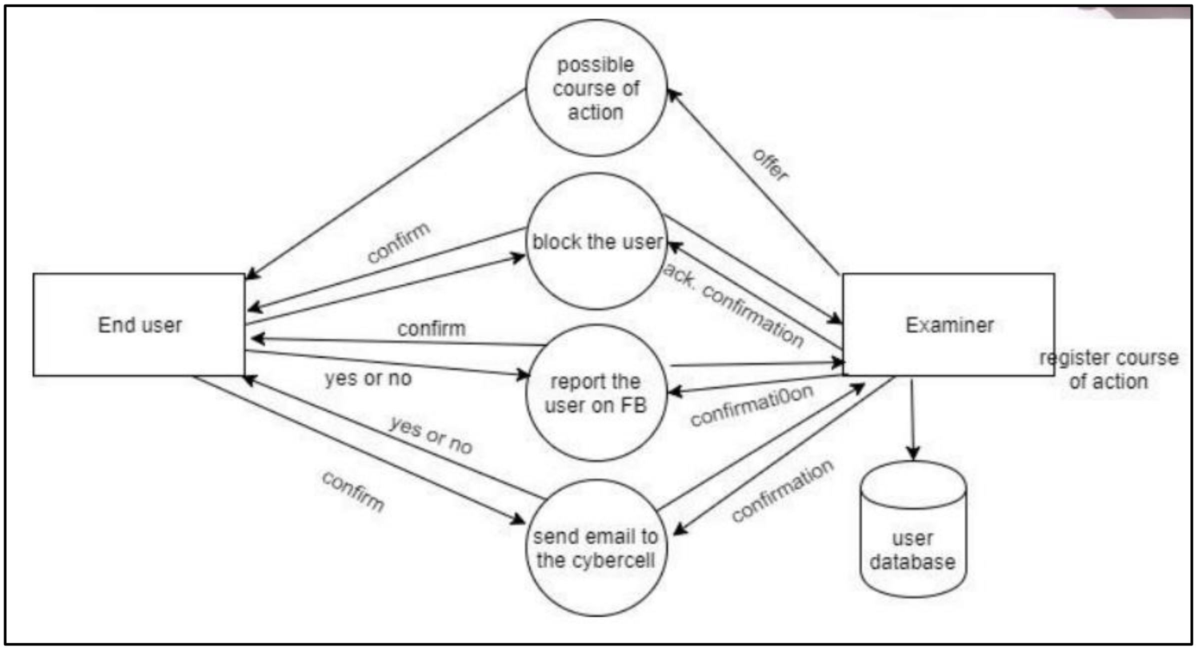


figure 4.3.10- Normalization Level Course of Action 1 DFD

4.4 Project Scheduling & Tracking using Time line / Gnatt Chart

4.4.1 Project scheduling

	At Risk	Task Name	Start Date	End Date	Assigned To	Durat...	% Complete	Prede...	Notes
1		Watch how-to video on this template (4:16)							
2									
3		**Section 1 - Research and documentation**	**09/01/17**	**09/12/17**		**7.5d**	**100%**		
4		Sub-task 1 - Collect IEEE papers and make summary	09/01/17	09/05/17	Dipika, Moh	3d	100%		
5		Sub-task 2- Project report	09/06/17	09/07/17	Megha	2d	100%	4	
6		Sub task 3- Project PPT	09/08/17	09/12/17	Riddhi	2.5d	100%	5	
7		**Section 2- Data Collection**	**10/12/17**	**10/25/17**	**Megha,Dipi**	**10d**	**20%**		
8		**Section 3- Implementation**	**01/04/18**	**02/26/18**		**38d**			
9		Preprocessing	01/04/18	01/10/18		5d			
10		Sessionization	01/11/18	01/12/18		2d		9	
11		Feature extraction	01/15/18	01/17/18		3d			
12		Classification	01/19/18	01/24/18		4d			
13		Report generation	01/30/18	02/07/18		7d			
14		Final action	02/09/18	02/16/18		6d			
15		IEEE paper 2	02/20/18	02/26/18		5d			

figure 4.4.1: Project scheduling

4.4.2 Section 1: Research and Documentation

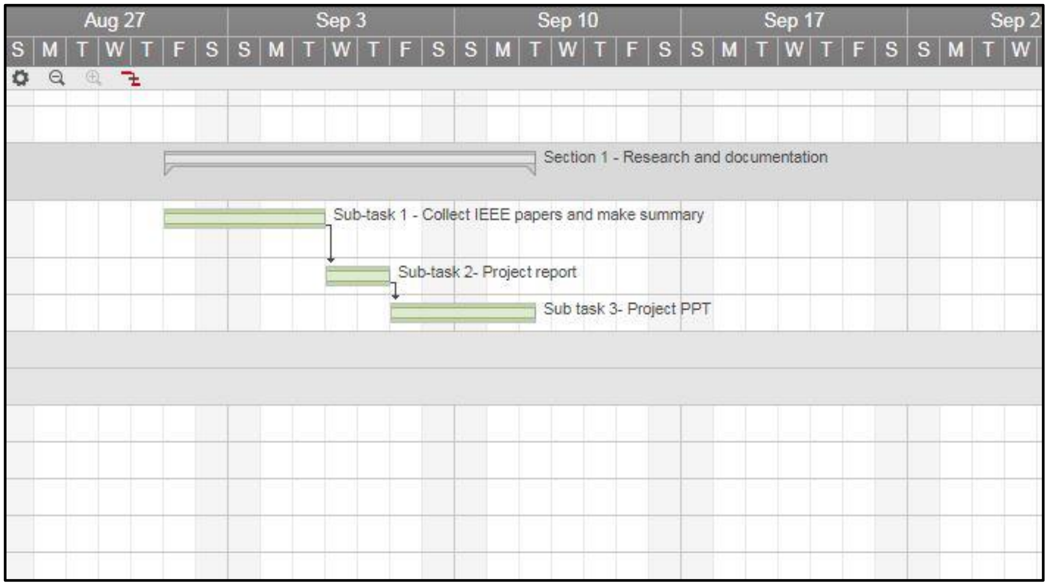


figure 4.4.2: Gantt chart: research and documentation

4.4.3 Section 2: Data Collection

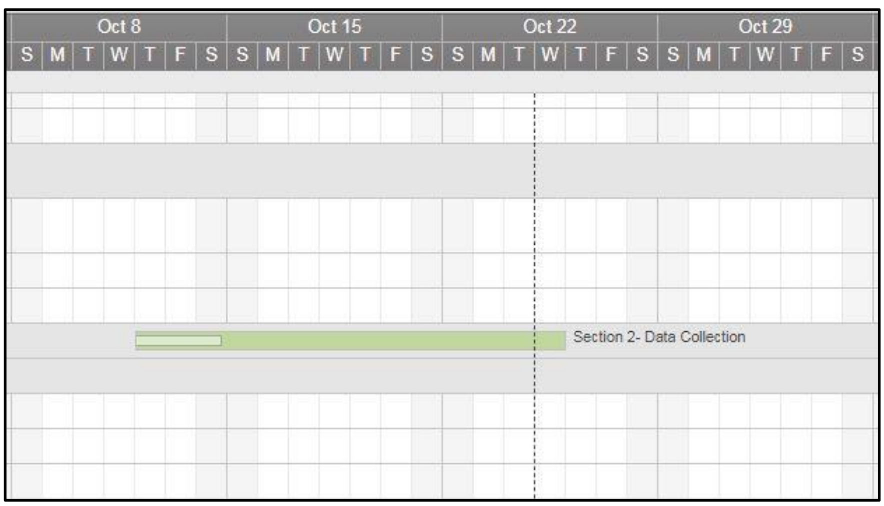


figure 4.4.3: Gantt chart: data collection

4.4.4 Section 3 part 1

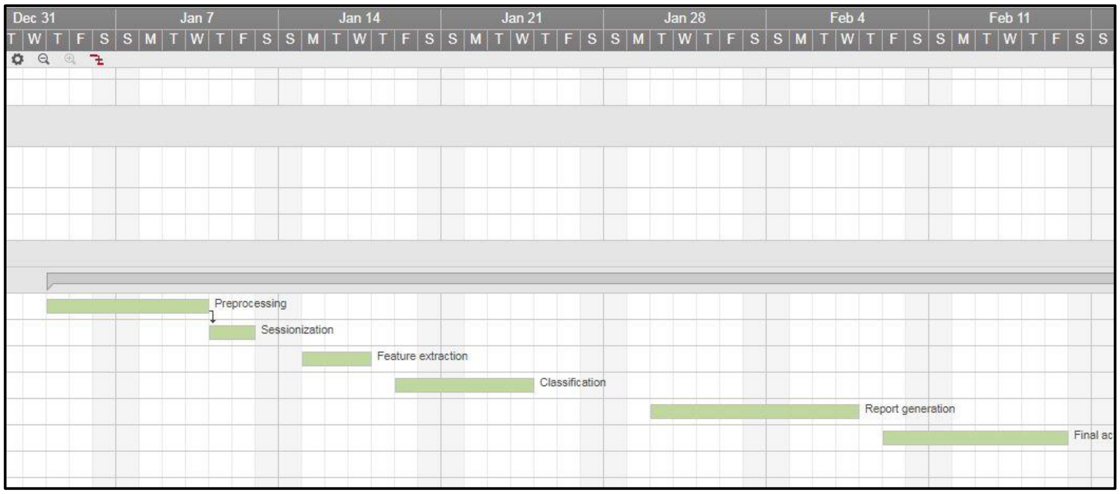


figure 4.4.4: Gantt chart: section 3 part 1

4.4.5 Section 3 part 2

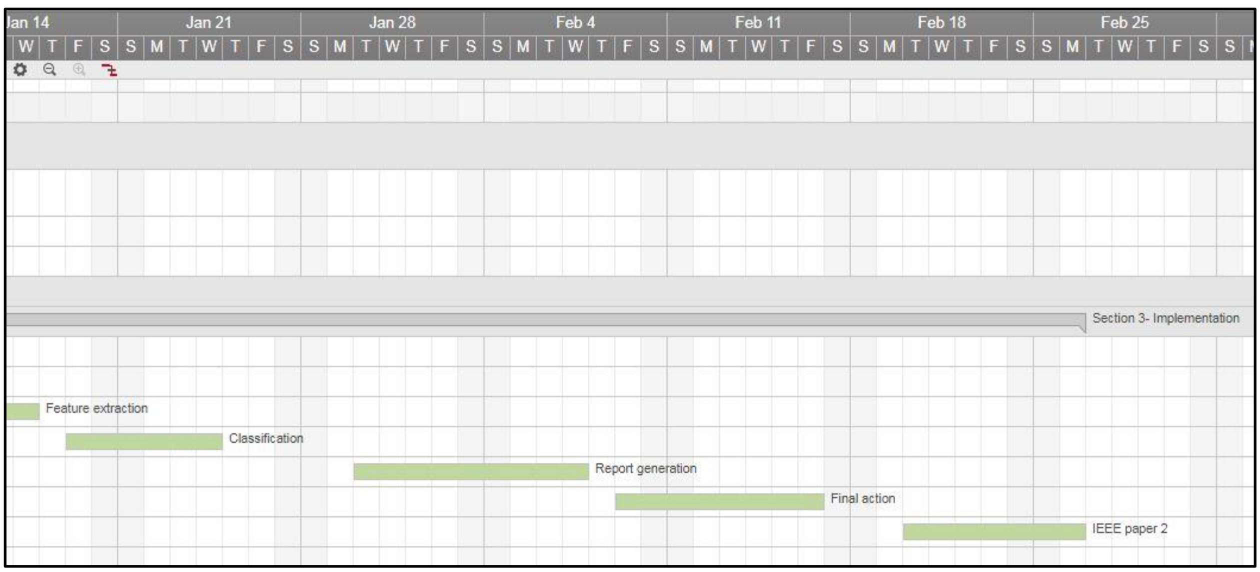


figure 4.4.5 Gantt chart: section 3 part 2

CHAPTER 5: IMPLEMENTATION DETAILS

5.1 Algorithms

5.1.1 Word sense disambiguation

In English as well as in any other natural language, different words can mean different things according to the context and reference. For example, the word “bow” can mean the archery equipment “bow” or can mean the act of bending your head or body forward. To gauge the exact sense of the word, it is important to do disambiguate or clear the ambiguity of the meaning of the word in that context. This is done using word sense disambiguation.

An example of this method can be given as follows:

Consider,

company_list= {Apple, Microsoft, Google, Samsung}

Fruit_keywords_list= {Orange, Mango, Cherry, Apple}

If any item in the company keyword list is present in the given sentence, then Sentence_score +=1

But If the item in the fruit keyword list is in the given sentence, then

Sentence score -=1

If the sentence score > 0, the sentence has a company sense.

And, if the sentence score<0, then the sentence is used in the fruit sense.

Otherwise, the sense of sentence is undecided.

Example 2:

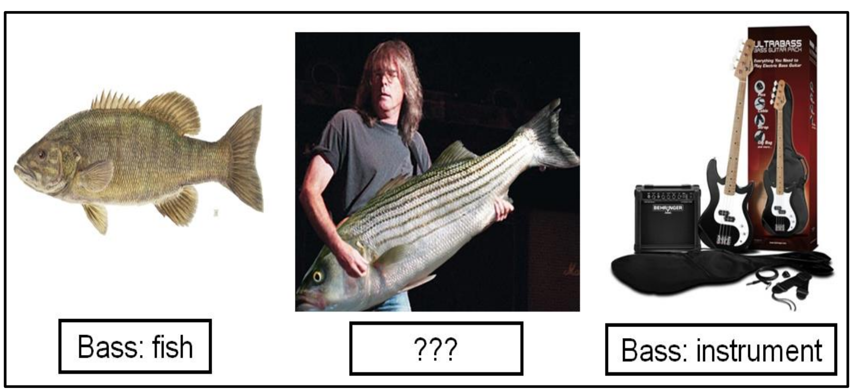


Figure 5.1.1 Word sense disambiguation

5.1.2 Tokenization:

Tokenization is a process of slicing the data into the smallest possible unit. The Facebook comments are divided into single word units and stored as a csv file. Tokenization has two steps. First, the text is tokenized into sentences. And then the sentences are tokenized into words. Tokenization is followed by stop words removal and lemmatization. For example, consider the sentence, “John Doe is an architect.”. This will produce the following tokens: ‘John’, ‘Doe’, ‘is’, ‘an’, ‘architect’.

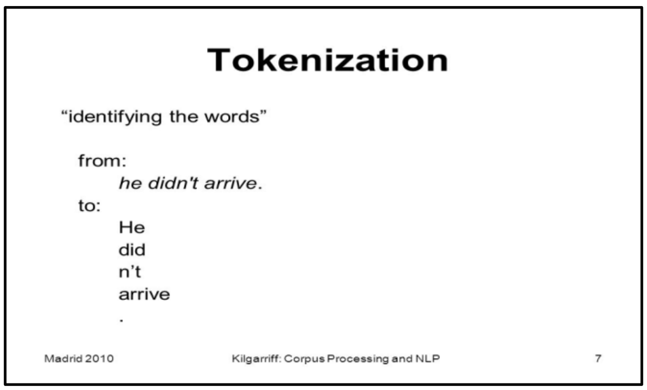


Figure 5.1.2(a) Tokenization example

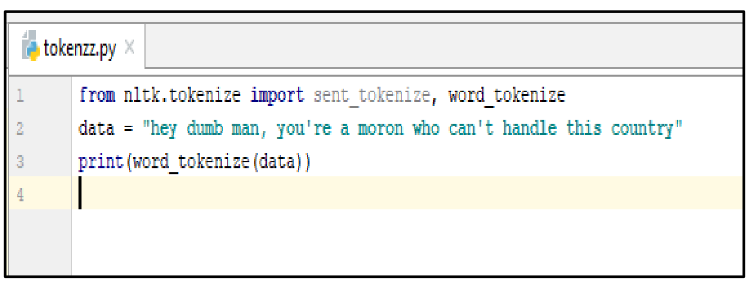


Figure 5.1.2(b) code

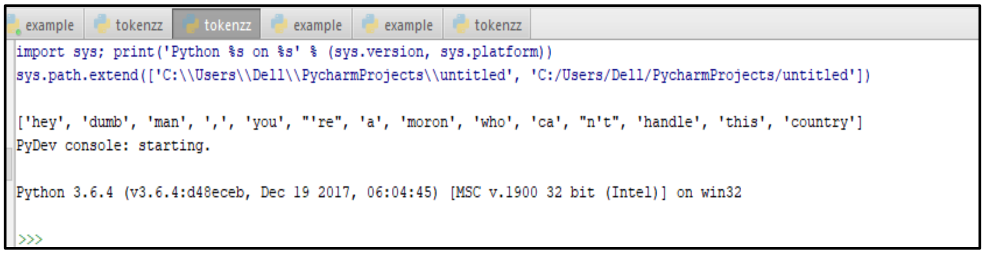


Figure 5.1.2(c) output

5.1.3 Stop-words removal:

A stop word is a commonly used word like in, the, an, for, of, etc. We certainly do not want these words to take up space in our database or utilize the processor. Hence, we eliminate these words along with the punctuation marks. For this purpose, we maintain a database of characters and words that we consider as stop words based on the document frequency of each word.

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

Figure 5.1.3 stop-word removal

5.1.4 Suffix stripping:

The problem of removing morphological suffixes from a word to get the stem is referred to as suffix stripping algorithm. This process of stemming is used to unite the different forms of a word into a common representation called the ‘stem’. The process basically involves removal of the suffixes from the words. For example, the words: dancer, danced, danced, dancing could be reduced to the common root ‘dance’.

Figure 5.1.4 Suffix stripping

5.2 Comparative Analysis with the existing algorithms

Subjectivity - Subjectivity is a measure which tells us whether the data is subjective or objective. A subjective may or may not express feelings and emotions. For example, "i like monsoons" is a subjective statement and does express some feelings. But the sentence " I want to go home" does not depict any kind of emotions and is still considered to be subjective data.

Polarity- Polarity describes the type of emotions expressed in the data. It tells us if the data is positive, negative or neutral. Generally, the intensity of emotions determines the strength of a sentiment, For example, "1+ series are the best phones available in the mobile market" depicts a positive emotion, whereas "The services provided by Toyota are horrible" showcase negative emotions.

5.3. Evaluation of the developed system

5.3.1 Naive Bayes Algorithm:

Our main goal is the classification of comments into offensive and clean i.e. it is a binary classification and partly subjective classification with respect to their sentiment and subject matter. We used the powerful scikit-learn library in Python for this purpose. This library is better than nltk because where nltk only supports Gaussian based Naïve Bayes, scikit-learn supports its multinomial distribution. This library can be downloaded like any other library in Python by simply using the "pip install" command. In addition to scikit-learn, other libraries imported for smooth processing were nltk, csv, numpy, pandas, genism, etc. Pycharm SDK is the platform used for coding.

The Naïve Bayes method is previously known to be an effective machine learning algorithm pertaining to the classification of spam content. It classifies both numerical and textual data. One of its major features is that it believes in independence between any pair of feature points. Some of its main advantages over other classification methods like SVM, Decision trees are higher training efficiency, quicker convergence to solution, comparatively easier implementation and large vocabulary-oriented data handling. In one pass of the testing data, it first computes the conditional probability of individual features with reference to the test dataset. Following that, it applies the Bayes theorem to obtain the posterior probability.

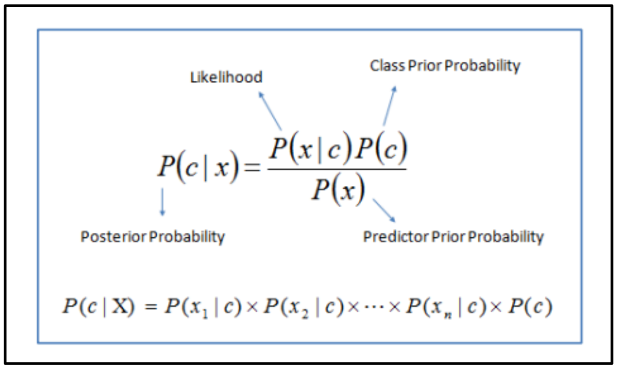


Figure 5.3.1(a) Naïve Bayes equation

Using the Naïve Based Classification method, we could obtain about 79% accuracy. However, it largely depends on the quality and quantity of training and testing data. Once the classification is done, the results can be stored back into a csv file, to be backtracked to the original comment. This method did show some inefficiencies. The Naïve Based classifier will not be as effective for features that are highly dependent like short texts. Further, conditional independence assumption cannot be wholly relied on in real world data.

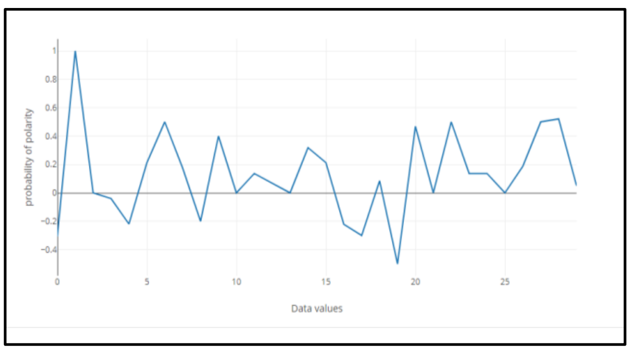


Figure 5.3.1(b) Polarity using Naïve Bayes model

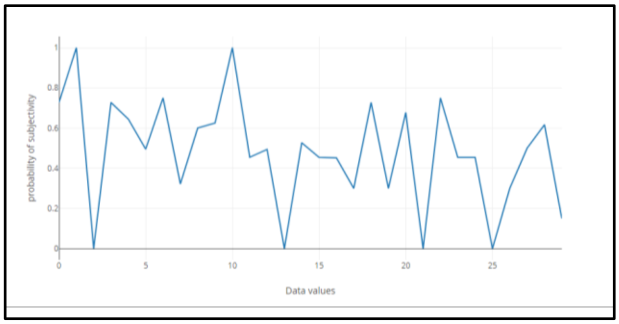


Figure 5.3.1(c) Subjectivity using Naïve Bayes model

5.3.2 Support Vector Machine Algorithm

Support Vector Machine, a machine learning algorithm is used for categorization of huge amount of text. The main motive is to find a hyperplane that separates vectors in one class from vectors in other classes. The technique which is used here is called as the kernel trick. It is used to transform our training data and then based on these transformations, an optimal boundary is found out between the possible outputs. We used support vector machine algorithm to categorize and estimate the strength of positive and negative sentiments of the comments. The experiments are performed on the dataset containing comments of a Facebook page. Naïve Bayes classifier is better than SVM in sentiment classification. Because SVM doesn't perform well when the dataset is too large or if it contains noise. The preliminary results don't seem to be promising as it gives an accuracy of 58% for our training dataset which less than the accuracy percentage of Naïve Bayes Algorithm.

		Correct labels	
		Positive	Negative
Classified labels	Positive	True Positive(TP)	False Positive(FP)
	Negative	False Negative(FN)	True Negative(TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 * Precision * Recall}{Precision + Recall}$$

Figure 5.3.2(a) Performance parameters

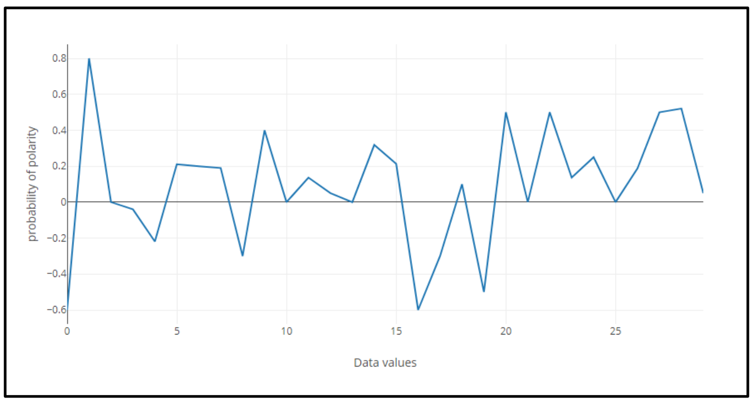


Figure 5.3.2(b) Polarity using SVM model

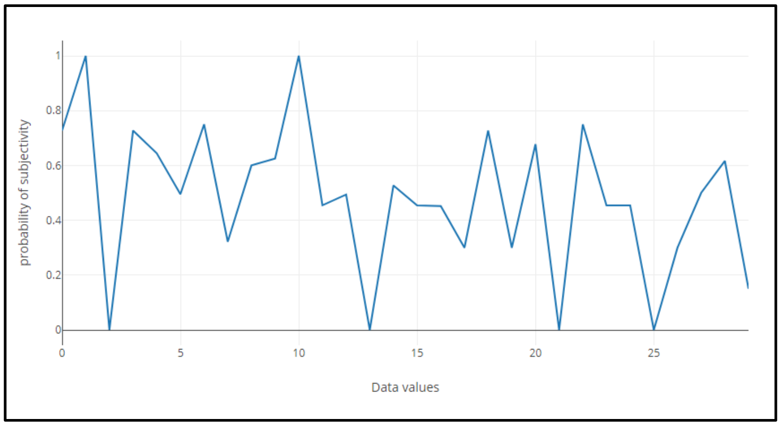


Figure 5.3.2(c) Subjectivity using SVM model

The performance after performing sentiment analysis on the training data uses the following formula:

Text	Polarity
@Kelsie_love really, my boyfriend isn't going to get any siblings? whatevas, you're totally going to have another one, bitch. FRODO! haha	Negative
@littledotty nope not my job, not heard anything from that nor my assignment results im expecting. just about life really its poo	Negative
@tessdejong too bad its not MY bday party! hehe	Negative
Is working and then wedding at 2 yippy skippy	Neutral
Oh. Wow. dont look at people or scream conor oberst	Neutral
@xia_hime i'm going to aya! got a room onsite though so cant hotel share	Neutral
Finished garden for today. Hey fever broke out finally	Negative
@sebby_peek I kind of have to, for myself you don't like me though sweet dreams love xoxoxoxoxox	Positive
@drewryanscott nope but i want one to	Neutral
@UNO_OUT what about me jay I want a keychain	Neutral
@thepistol good luck with ur concert in san diego! wish i could be there but i live all the way in buffalo i know u'll do great tho! duh.	Positive

Table 5.3.2 Sentiment labeling using SVM

5.3.3 Neural Networks

Neural network is one of the many techniques used in machine learning and is also known as Artificial Neural Networks (ANN). The main element of ANN is the novel structure of the information processing system, which is influenced by the way information is processed by biological nervous systems and the brain. It is made of an enormous number of highly interconnected units known as ‘neurons’, processing in coordination to execute a particular task. Artificial neural networks learn through examples, just like people. For eg, pattern recognition and data classification.

There are various deep neural network models available for data classification.

Unsupervised mode: The main benefit of using the unsupervised mode is word2vec, that enables us to create a low dimension distributed representation of data. These words work on a simple logic that words with similar meaning must have similar environment. Skip-gram and continuous bag of words are the two popular models used in this mode. We decided to go to with skip-gram as it has a very simple and straightforward design.

The primary logic behind skip gram is that it considers each word in a huge corpus and simultaneously also takes one word which surrounds it within a specific defined ‘window’. It then trains the neural network in such manner that it will predict the probability of each word to occur in the window around the focus word. To make things convenient, we create a vocabulary of various words available in our data set. We then encode this data as a vector that has the same dimensions as our vocabulary. For eg, if we have a vocabulary made out of the words “the”, “big”, “black”, “bear”, “eats”, “the” “white”, “goat”, the word “bear” is represented by this vector: [0, 0, 1, 0, 0, 0 ,0 ,0].

Based on predicting the nearby training words, a model depiction of current word is generated in the training process. After training, the word embedding is created which is the vector of weights from the hidden layer.

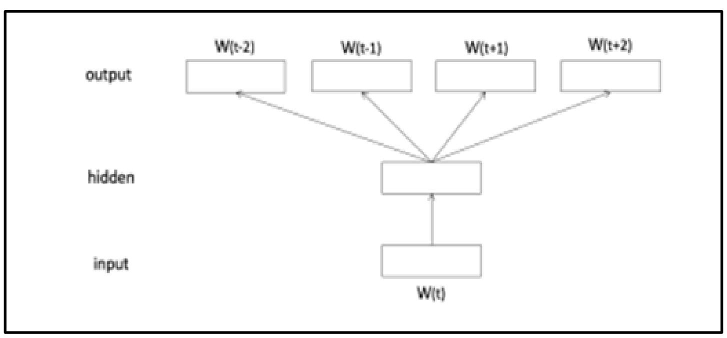


Figure 5.3.3(a) Primary logic behind neural Networks

W represents the set of vectors containing the words and the output layer predicts the next word in the scenario. Since the input is now ready, we use it to enter into the 2-layer neural network model. The probability for each text in the vocabulary will be displayed in the second layer.

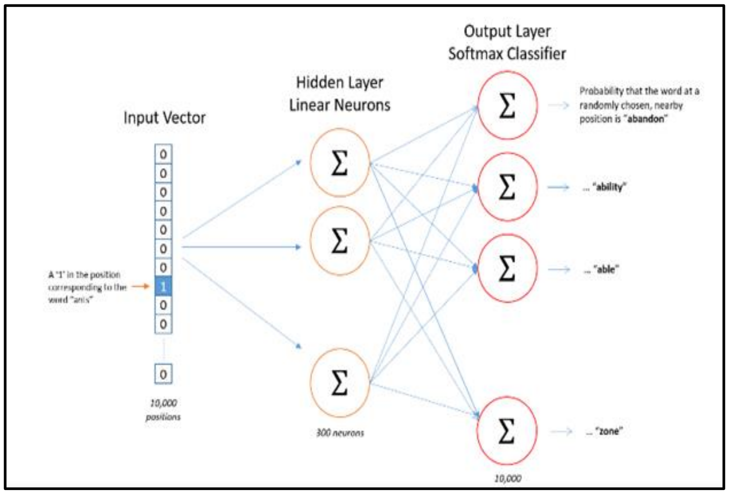


Figure 5.3.3(b) Functionality of NN using skip gram model

The unsupervised mode gives us approximately an accuracy of 58% when we use the skip gram model. The Skip gram model reduces 39% of the errors that occur during the implementation Continuous Bag of Words Model.

Supervised Mode: Supervised mode is used at times because of the fast execution feature that it offers. A data with a set of 'N' features are embedded and then averaged to middle layer. However, the performance of algorithm in supervised method is very elementary and primitive.

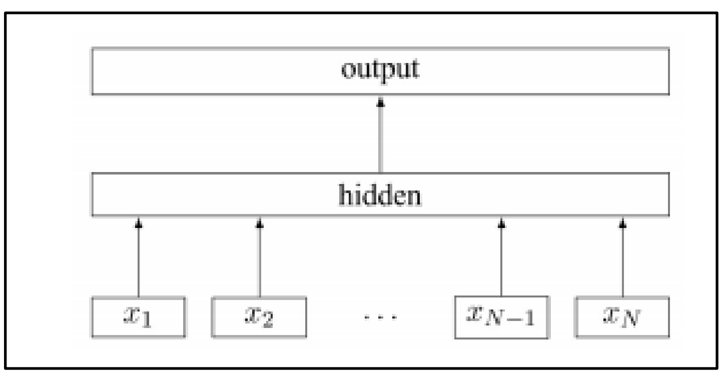


Figure 5.3.3(c) Supervised mode

Convolution Neural Network: CNN was used in natural language processing and generated remarkable results for a number of text classification tasks. It uses multiple layers with convolving filters whose target is to obtain local features.

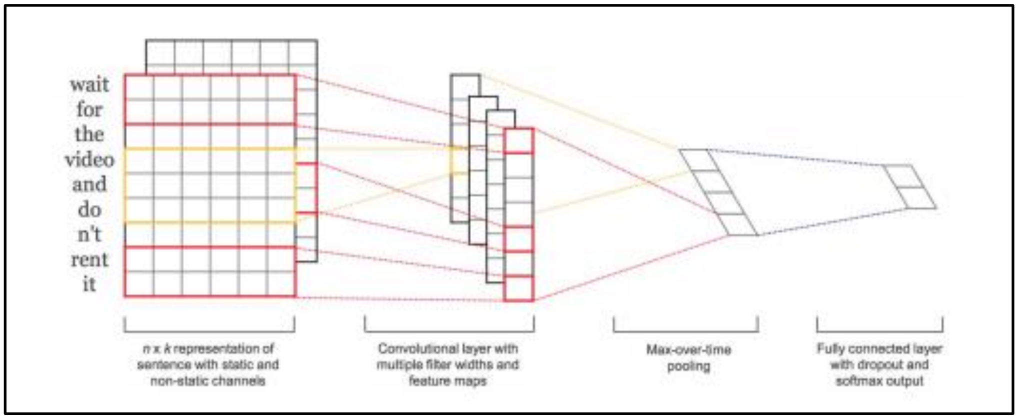


Figure 5.3.3(d) Convolution NN

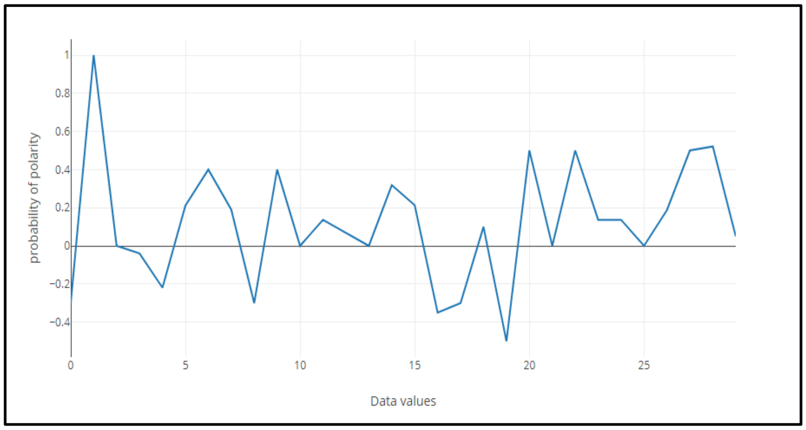


Figure 5.3.3(e) Polarity using NN

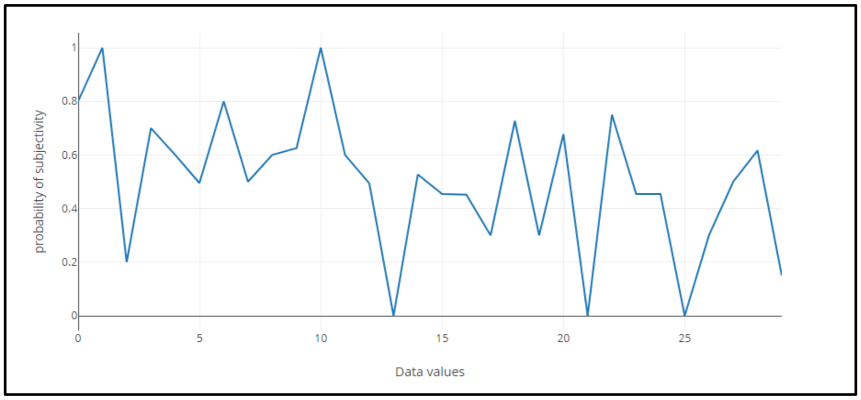


Figure 5.3.3(f) Subjectivity using NN

CHAPTER 6: TESTING

6.1 Data Extraction:

The facepager application helps us get data in csv format. Using an access token of the facebook profile, we can extract posts, comments, likes, etc.

Figure 6.1.1 Facebook page of Mr. Narendra Modi

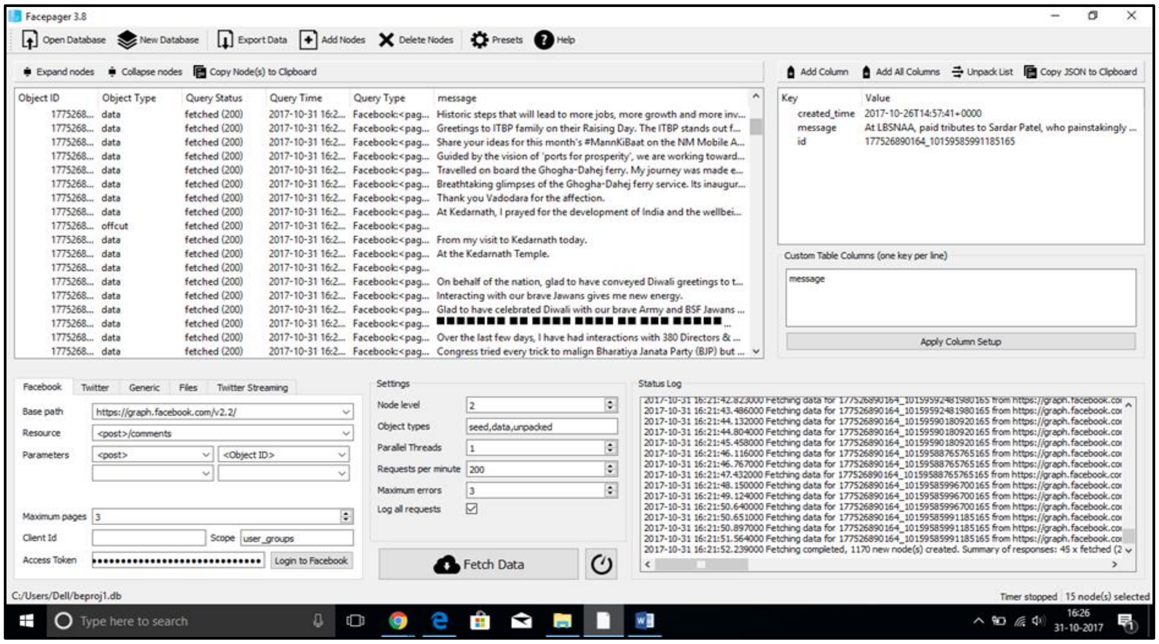
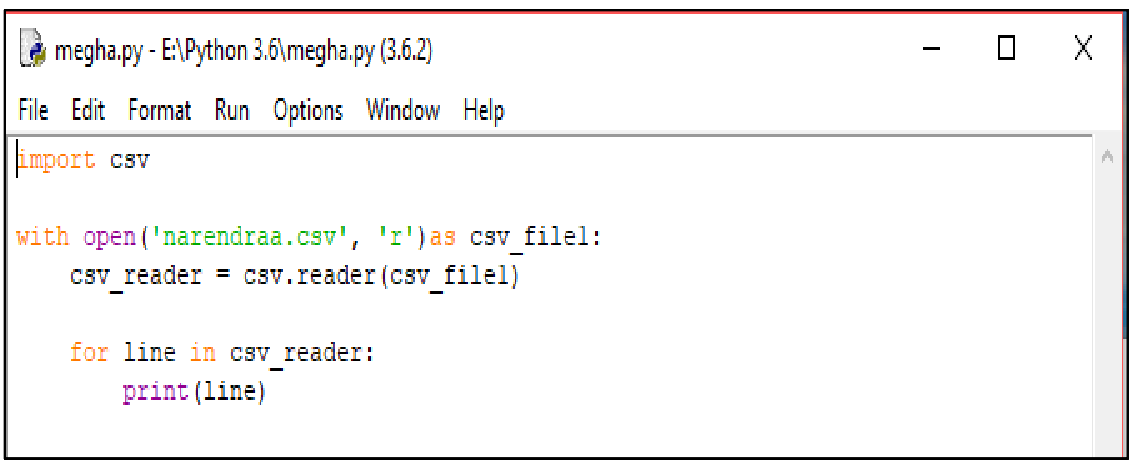


Figure 6.1.2 Facepager

6.2 Importing data in Python:




```
import csv

with open('narendraa.csv', 'r')as csv_file1:
    csv_reader = csv.reader(csv_file1)

    for line in csv_reader:
        print(line)
```

Figure 6.2.1 Importing data in Python

```
Python 3.6.2 (v3.6.2:5fd33b5, Jul  8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===================== RESTART: E:\Python 3.6\megha.py =====================
['id', 'message']
['457', 'This person works 18 hrs a day to keep the country together and keep the enemies at an arms distance.']
['12', 'Sir, what have you done so far ... we have voted you and u have only increased corruption u stupid dumbo']
['365', "Pm can't announce demonetization , only Governor of RBI can do that you are brainless"]
['789', 'Modiji,so far what you did for the country,India became worse than before,very scared to travel through,more crime,poverty and more rape,no peace,people are 
ecoming more tired,frustrated and upset']
['14', 'dear ModI I salute your speech always, this is the sign of a good leader']
['13', 'Excellent speech.I admire your stamina as you are totally dedicated to the nation.Keep it up PM Sir.']
['78', 'You r the worst leader ever please grow some balls…']
['2222', 'The Biggest problem with Our PM is that he can only talk. Useless PM']
['69', 'I hate myself that I voted for this person.']
['32', 'Shameless party which has put Indias neck deep in corruption ']
['1', 'you are a piece of shit Modi. Grow some balls!']
['9', "Modi Only knows how to give speeches.  Shouting ,playing with words and dramatizing . Politics at it's height"]
['99', 'best leader ever may god bless u']
['140', 'no sense of how to govern people']
['7', 'totally regretting my decision to vote for this foolish man']
['63', 'I m ur biggest fan and admirer. Proud to be an Indian!!!!']
['45', 'modi you are the most hardworking leader keep it up…']
['37', 'always fooling the public and just speaking and making false promises']
['764', 'nothing is fine in the country. Petrol prices are increasing, farmer suicides are increasing.. Change the PM']
['355', 'Indians always criticize..modi u r fantastic']
['76', 'Mr.Prime Minister,you are doing a commendable job.']
['1036', 'idiot old man just roaming the world and doing nothing for indians']
['911', 'Those who want to change something in life u can follow Narendra Modi.. Awesome speech.']
['87', 'this person is the most fake I have ever seen. I hope the next PM is genuine']
['4321', 'awesome work modi india is now soon going to become a developed country']
['697', 'worst leader ever who sold the country']
['856', 'you are a dumbfuck who just wants votes and nothing else… shame on u and ur party… rot in hell']
['44', 'after seeing modis speech I became a fan of politics']
['398', 'modi is scared of america and wants all indians to become slaves of americans…ban modi']
>>>
```

Figure 6.2.2 Output

6.3 Python Code

6.3.1 Lemmatization

```
from __future__ import print_function
from nltk.stem import *
from nltk.stem.porter import *


stemmer = PorterStemmer()
plurals = ['bitching', 'hater', 'running', 'madam', 'denied']
singles = [stemmer.stem(plural) for plural in plurals]
print(' '.join(singles))  # doctest: +NORMALIZE_WHITESPACE
```

```python
from __future__ import print_function
from nltk.stem import *
from nltk.stem.porter import *


stemmer = PorterStemmer()
plurals = ['bitching', 'hater', 'running', 'madam', 'denied']
singles = [stemmer.stem(plural) for plural in plurals]
print(' '.join(singles))  # doctest: +NORMALIZE_WHITESPACE
```

Figure 6.3.1 Lemmatization

6.3.2 Tokenization:

```python
from nltk.tokenize import sent_tokenize, word_tokenize
data = "hey dumb man, you're a moron who can't handle this country"
print(word_tokenize(data))
```

Figure 6.3.2(a) Tokenization

```
import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['C:\\Users\\Dell\\PycharmProjects\\untitled', 'C:/Users/Dell/PycharmProjects/untitled'])

['hey', 'dumb', 'man', ',', 'you', "'re", 'a', 'moron', 'who', 'ca', "n't", 'handle', 'this', 'country']
PyDev console: starting.

Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)] on win32

>>>
```

Figure 6.3.2(b) Tokenization

6.3.3 Sentiment Classification

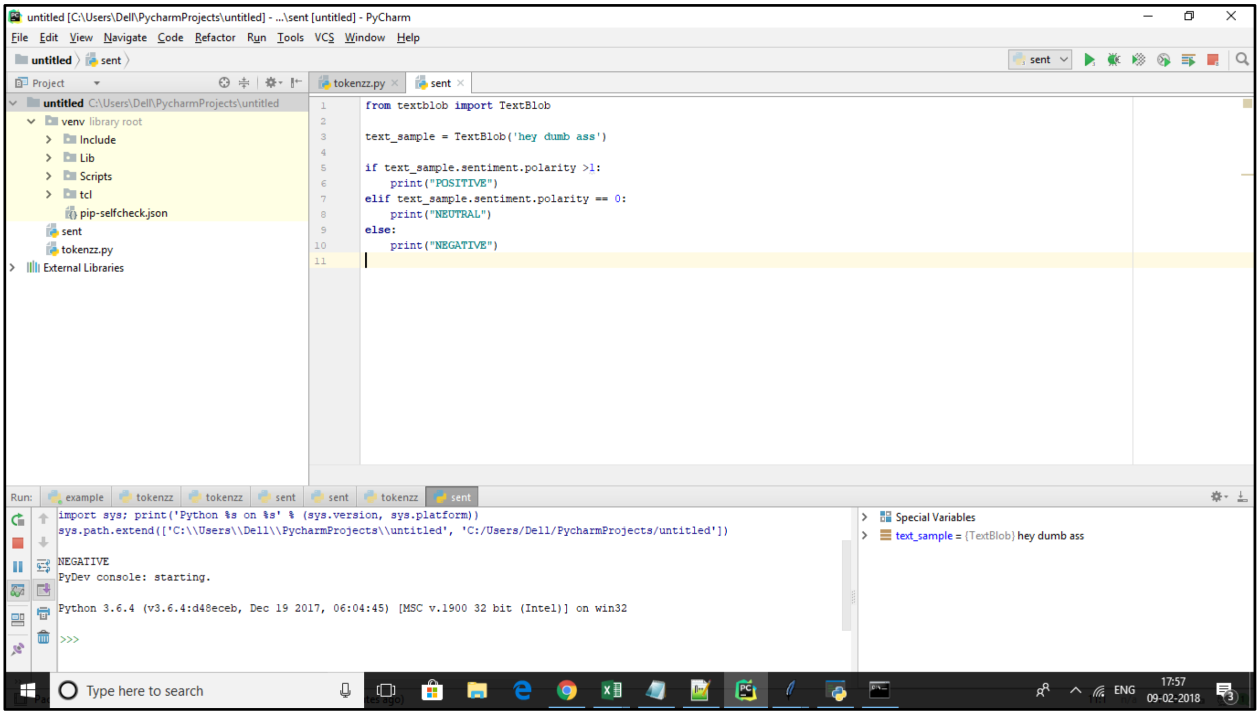


Figure 6.3.3 Sentiment Classification

6.3.4 Computing the subjectivity and polarity:

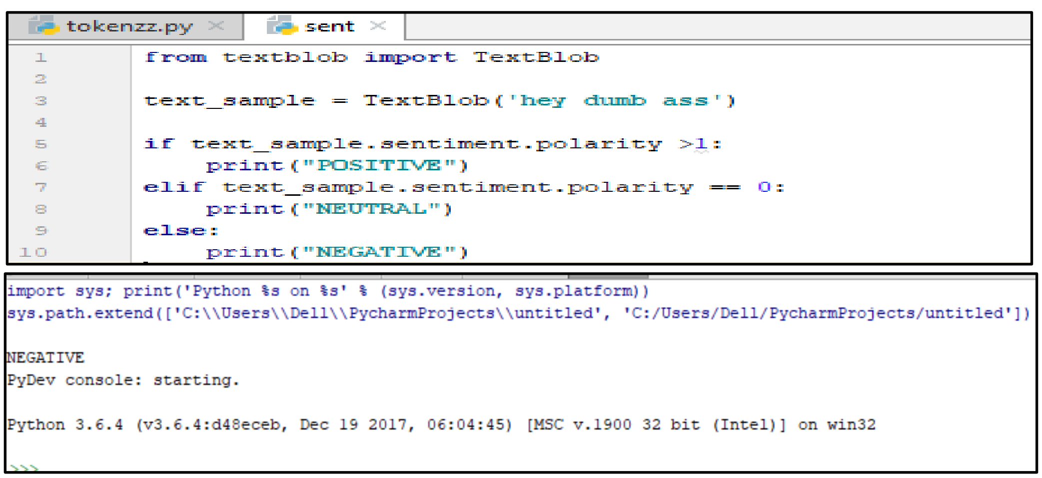


Figure 6.3.4(a) Computing polarity and subjectivity

```
*megha1.py - E:\Python 3.6\megha1.py (3.6.2)*
File  Edit  Format  Run  Options  Window  Help

import csv
from textblob import TextBlob

infile = 'narendraa.csv'

with open(infile, 'r') as csvfile:
    rows = csv.reader(csvfile)
    for row in rows:
        sentence = row[0]
        blob = TextBlob(sentence)
        print (blob.sentiment)
```

Figure 6.3.4(b) Reading a .csv file

```
Python 3.6.2 Shell
File  Edit  Shell  Debug  Options  Window  Help

Python 3.6.2 (v3.6.2:5fd33b5, Jul  8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
 on win32
Type "copyright", "credits" or "license()" for more information.
>>>
====================== RESTART: E:\Python 3.6\meghal.py ======================
Sentiment(polarity=-0.30681818181818, subjectivity=0.7272727272727273)
Sentiment(polarity=1.0, subjectivity=1.0)
Sentiment(polarity=0.0, subjectivity=0.0)
Sentiment(polarity=-0.03977272727272728, subjectivity=0.7272727272727273)
Sentiment(polarity=-0.21939393939393934, subjectivity=0.6442424242424243)
Sentiment(polarity=0.2110389610389610З, subjectivity=0.4951298701298701)
Sentiment(polarity=0.5, subjectivity=0.75)
Sentiment(polarity=0.17454545454545453, subjectivity=0.32181818181818184)
Sentiment(polarity=-0.2, subjectivity=0.6)
Sentiment(polarity=0.4, subjectivity=0.625)
Sentiment(polarity=0.0, subjectivity=1.0)
Sentiment(polarity=0.13636363636363635, subjectivity=0.45454545454545453)
Sentiment(polarity=0.06818181818181818, subjectivity=0.49393939393939396)
Sentiment(polarity=0.0, subjectivity=0.0)
Sentiment(polarity=0.3181818181818182, subjectivity=0.5272727272727272)
Sentiment(polarity=0.2130681818181818, subjectivity=0.4545454545454545З)
Sentiment(polarity=-0.22121212121212122, subjectivity=0.4515151515151515)
Sentiment(polarity=-0.3, subjectivity=0.3)
Sentiment(polarity=0.08522727272727272, subjectivity=0.7272727272727273)
Sentiment(polarity=-0.5, subjectivity=0.29999999999999993)
Sentiment(polarity=0.4681818181818182, subjectivity=0.6772727272727272)
Sentiment(polarity=0.0, subjectivity=0.0)
Sentiment(polarity=0.5, subjectivity=0.75)
Sentiment(polarity=0.13636363636363635, subjectivity=0.45454545454545453)
Sentiment(polarity=0.13636363636363635, subjectivity=0.45454545454545453)
Sentiment(polarity=0.0, subjectivity=0.0)
Sentiment(polarity=0.1875, subjectivity=0.3)
Sentiment(polarity=0.5, subjectivity=0.5)
Sentiment(polarity=0.5208333333333334, subjectivity=0.6166666666666667)
Sentiment(polarity=0.05, subjectivity=0.15)
```

Figure 6.3.4(c) Polarity and subjectivity output

CHAPTER 7:

RESULT ANALYSIS

7.1 Screenshots of User-Interface

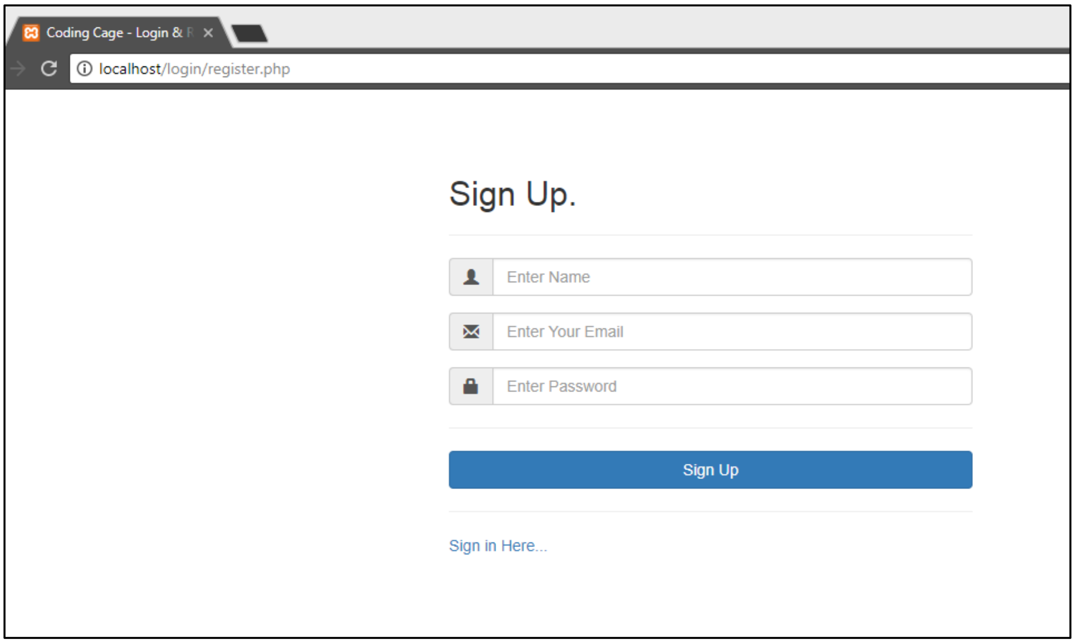


Figure 7.1.1 Sign up page for the user

The person who has to get the analysis done will have to first signup on our portal.

Figure 7.1.2 Introduction page

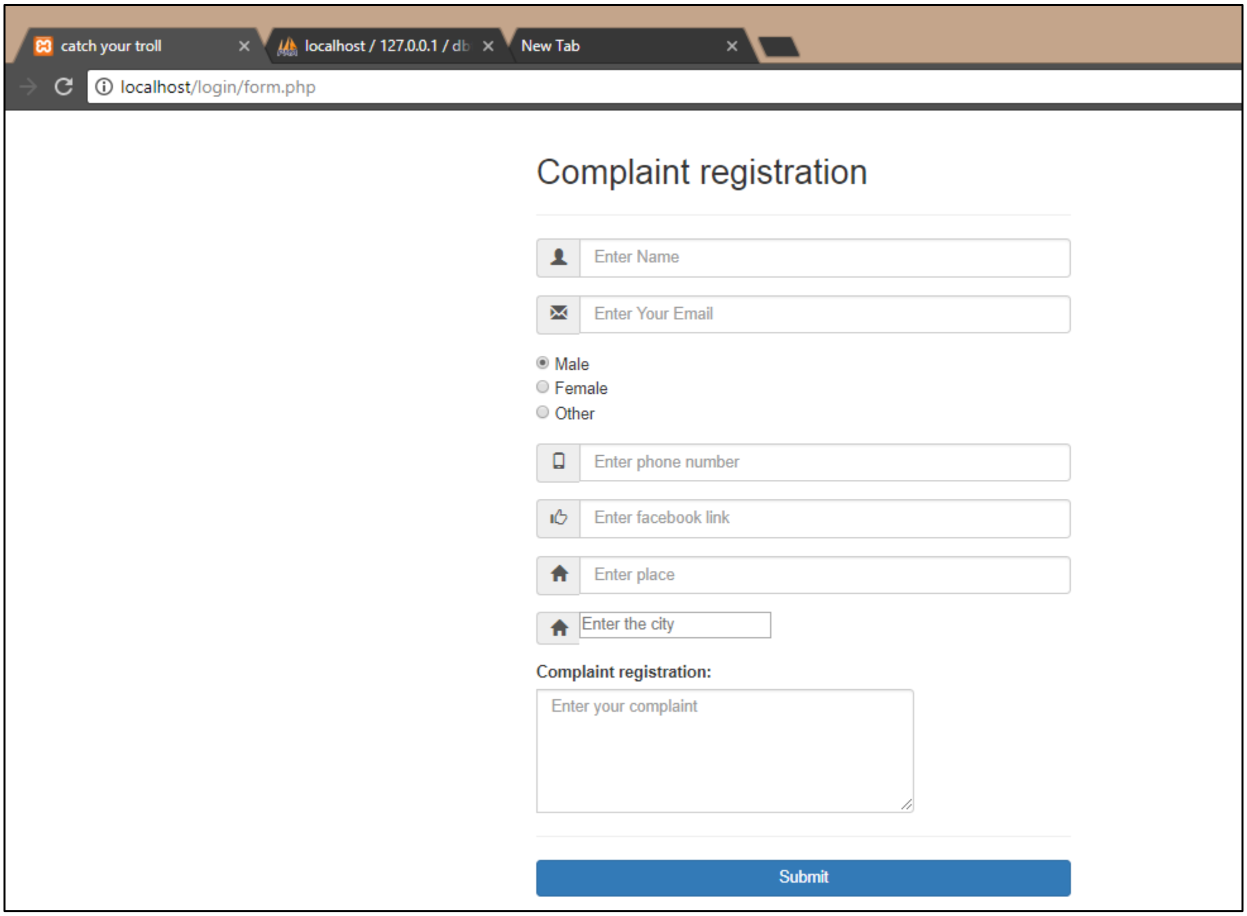


Figure 7.1.3 Form to register a complaint

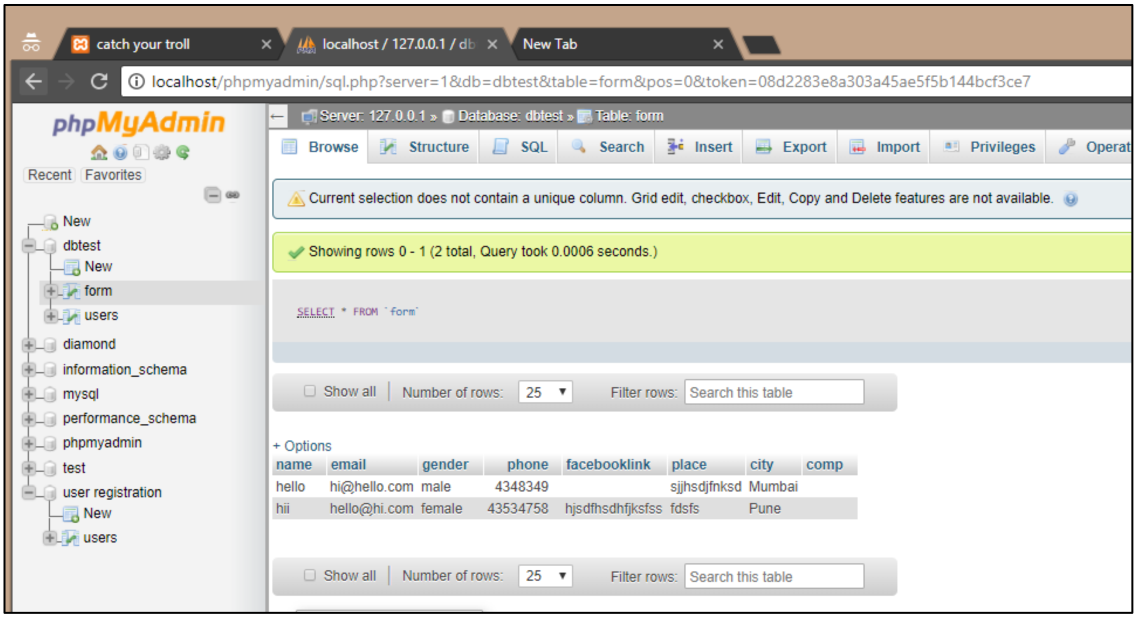


Figure 7.1.4 Database of complaints registered

By using this database we will get the information of the user and the Facebook page which needs to be analyzed.

7.2 Wordcloud

Figure 7.2 wordcloud

The fundamental idea of creating a word cloud is to represent a set words on a canvas, where its size depends on its importance. In our case, greater the size indicates higher degree of profanity usage.

CHAPTER 8:
CONCLUSION

8.1 Limitations

If the harassment or bullying is via personal messages, our tool will not be able to reveal the identity of that person because that data keeps on changing dynamically and our tool doesn't support that feature as of now. Also, crowdsourcing of data will be a cumbersome task to deal at moment.

8.2 Conclusion

The emotional consequences that the preys of cyberbullying suffer can be disastrous and mortifying. It is even worse than face-to-face bullying, as the victim has no idea of who the bully is. On examining the computer forensic process of obtaining digital evidence from social media, and the legal aspects of such cases of cyberbullying, three models were used on the training dataset i.e. Naive Bayes, Support Vector Machine and Neural Networks. The accuracy obtained by Naïve Bayes Classification method was 79%, whereas SVM offered an accuracy of 55% and 58% of accuracy was achieved by the Artificial Neural Network Model.

Thus, it is clear that the Naïve Bayes approach is the most efficient one and is therefore the best classifier for sentiment analysis, out of the three models chosen. The evidence provided by the application can be used by the users i.e., bloggers or any individual or organisation to report the crime to the cyber-crime department for further legal actions.

8.3 Future scope

As reflected through this project and report, bullying behavior is a serious public health issue with significant negative consequences, in both the short and long term, for the people who are bullied, the people who perpetrate bullying behavior, and people who are both perpetrators and targets of bullying. This application will aim to curb these menaces faced by bullying and help to reduce the number of cyber bullying cases that are occurring at a full-fledged rate. This application will provide the users with a variety of controls which include blocking the accused person, filing an online report against the accused on our portal, or registering a complaint to the cyber security team. If strict and imperative measures are taken against cyber bullying and the issue is addressed sternly, this social evil, cyber bullying, will not persist for long.

References

Papers:

1)Forensic investigation of social networking applications

2) Using Naïve Bayes Algorithm in detection of Hate Tweets.

3) Sentiment analysis for hate speech detection on social media

4) Analysis of Various Sentiment Classification Techniques

5) Sentiment Classification using Machine Learning Techniques

6) Using Machine Learning Techniques for Sentiment Analysis

7) Social Media Sentiment Analysis using Machine Learning Classifiers

Project Progress review sheets

Progress report 1:

Project Evaluation Sheet 2017 - 18 GROUP NO.: 48

of Project: ANOMALY BASED FORENSIC ANALYSIS OF SOCIAL MEDIA

ıp Members: Mohit Ahuja - D17A, 02; Dipika Jiandani - D17A, 35; Riddhi Karkera - D17A, 37; Megha Manglani - D17A, 45

	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (3)	Applied Engg & Mgmt principles (3)	Life - long learning (3)	Professional Skills (5)	Innovative Approach (5)	Total Marks (50)
Review Project Stage 1	3	3	3	3	4	2	2	2	1	2	2	2	4	4	37

Comments: Very Basic Approach. No comparative analysis. All cases not considered

Richard Sequeira
Name & Signature Reviewer1

	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (3)	Applied Engg & Mgmt principles (3)	Life - long learning (3)	Professional Skills (5)	Innovative Approach (5)	Total Marks (50)
Review of Project Stage 1	3	3	3	3	4	2	2	2	1	2	2	2	4	4	37

Comments: Comparative Analysis needed

Date: 26/02/18

Swati Sharma
Name & Signature Reviewer2

Progress report 2:

Inhouse/ Industry:

Class: D17A/B/C
Group No.: 48

Project Evaluation Sheet 2017 - 18

Title of Project: ANOMALY BASED FORENSIC ANALYSIS OF SOCIAL MEDIA

Group Members: Mohit Ahuja (02), Dipika Jiandani (35), Riddhi Karkera (37), Megha Manglani (45)

	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation on Skills (3)	Applied Engg &Mgmt principles (3)	Life - long learning (3)	Professional Skills (5)	Innovative Approach (5)	Total Marks (50)
Review of Project Stage 1	4	4	3	2	4	2	2	2	2	3	3	2	4	4	41

Comments:

Pooja Nagdev
Name & Signature Reviewer1

	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation on Skills (3)	Applied Engg &Mgmt principles (3)	Life - long learning (3)	Professional Skills (5)	Innovative Approach (5)	Total Marks (50)
Review of Project Stage 1	4	4	3	2	4	2	2	2	2	3	3	2	3	4	40

Comments: GUI should be included

Date: 15th March,2018

Yugchhaya Galphade
Name & Signature Reviewer2

Appendix:

Certificates:

Semester 7: We participated in a project competition 'Basic`18'. We were shortlisted in top 6 teams from a group of 25 teams. We presented our project to delegates from Spain in VESIM.

Semester 8: We published our paper "Comparative analysis of different machine learning algorithms to detect cyber bullying on facebook" in the reputed publication IJRASET.

Comparative Analysis of Different Machine Learning Algorithms to Detect Cyber-bullying on Facebook

Dipika Jiandani[1], Riddhi Karkera[2], Megha Manglani[3], Mohit Ahuja[4], Mrs. Abha Tewari[5]
Department of Computer Engineering,
VESIT
Mumbai, India
dipika.jiandani@ves.ac.in
riddhi.karkera@ves.ac.in
megha.manglani@ves.ac.in
mohit.ahuja16@ves.ac.in
abha.tewari@ves.ac.in

***Abstract*— Offensive language on social media has unfortunately become a common occurrence among users. The motive is to detect offensive language in a user message, post or comment and take necessary actions for the same. This is called as offensive language filtering. In this paper, we provide a comparison of different algorithms to build a solution through which Facebook users can find their cyber bullies and report them. The entire process consists of six stages: data collection, pre-processing, sessionization, ground truth, feature extraction and classification. Using machine learning algorithms for pre-processing and classification of the data and tools like Facepager and Pycharm, we have evaluated the processing, usage and accuracy of three major classification algorithms which are Naive Bayes, Support Vector Machine and Neural Networks.**

***Keywords*— Naïve Bayes, Support Vector Machine, Neural Networks, Facebook, Facepager, tokenization, word-sense disambiguation, sentiment analysis, subjectivity, polarity, cyber-bullying.**

I. INTRODUCTION

Anomaly based forensic analysis of social media refers to analysis on Facebook data using machine learning algorithms. The aim is to build a web application that will proactively detect and report cases of cyber-bullying and personal security intrusion on social media platforms (here, Facebook) using machine learning algorithms and behavioural analysis. The sub goals of our project are data collection, pre-processing, sessionization and crowd sourced labelling. The application can be used for identifying theft, theft of public data, public defamation, cyber stalking, bullying and other criminal activities on such sites. The anonymous nature of social networking applications can be leveraged by malicious users. Our focus is on conduction of forensic analysis on one of the most popular social media applications in the recent times, i.e. Facebook. The application has the ability to stop cybercrimes happening at a full-fledged rate.

II. PROBLEM DEFINITION

This project focuses on conducting forensic analyses on some of the widely used social networking applications like Facebook, Instagram to name a few. This analysis will be aimed at analysing offensive comments with the motivation of cyber-bullying posted on these applications and backtracking them to the offender. The extent, significance, and intention of the data that could be found and retrieved. If so, the suspect will be found guilty of a cybercrime since there will be a solid evidence to prove the activity was performed by him. This application includes pre-processing and analysis of data via various models of Machine Learning like Naive Bayes, Support Vector Machine and Artificial Neural Networks. The development of such an application has the ability to stop and reduce the rate of subjugating that has been happening online at a full-fledged rate. Our goal is to compare three classification models used in the development of a web application that will proactively detect and report cases of cyber-bullying and personal security intrusion on social media platforms like Facebook, Twitter, etc. using the concepts of behavioural analysis and machine learning. Further, a block action or report will be generated on the basis of the supporting evidence found through Forensic Analysis of social media sites

First, a tool called "Facepager" is used to get the data from Facebook. It gives access to different posts, pictures, comments and emoticons of various public profiles on Facebook using which the training and testing data are formed. With the help of pre-processing algorithms, the raw data is converted into executable form. The pre-processed data is then classified using the machine learning algorithms- naive Bayes, support vector machines and neural networks for sentiment analysis. A comparison of these three classification algorithms is done on the basis of the processing, performance and the accuracy of each of the them. The polarity and subjectivity of each algorithm is found and plotted on a graph to compare. Also, based on the frequency of the bad words, a word cloud is generated. The bad word which has the highest frequency will have a bigger size compared to the words whose frequency is less.

III. METHODOLOGY

A. Dataset

The classification of comments plays a crucial role in the process of filtering the comments. Such a classification would help in the categorization of numerous online content into offensive and clean comments reducing the pressure on human monitoring. We have developed and made use of two datasets namely the trained dataset and testing dataset for the purpose of effectively classifying the comments. Training a dataset prior to running the codes on the testing dataset helps to find the potential negative comments for further processing of dynamically obtained comments.

The training dataset is hatebase.csv, which is the static dataset obtained from hatebase.org, a Canadian website that provides a crowdsourced, multilingual corpus consisting of a repository of words and phrases indicating hate speech. This site is dynamically updated with new additions every single day. In addition to the data, the hatebase.csv file also consists of a column that contains the values indicating their degree of offensiveness. The trained dataset is coded with threshold being a fixed value indicating positive and negative comments on either side.

The test dataset is fetched using a data crawler tool called Facepager. It is a Facebook Graph API dependent tool, that can be used to extract Facebook data in the form of posts, photos, videos, group activity and the most import element in our case, comments. Once the comments are extracted from post(s) from the particular user's profile, it is exported to csv. This now becomes the testing dataset.

The comments that are thus obtained could be in the form of structured, semi-structured or completely unstructured text. It is necessary to handle all three kinds of data. For this purpose, pre-processing is important.

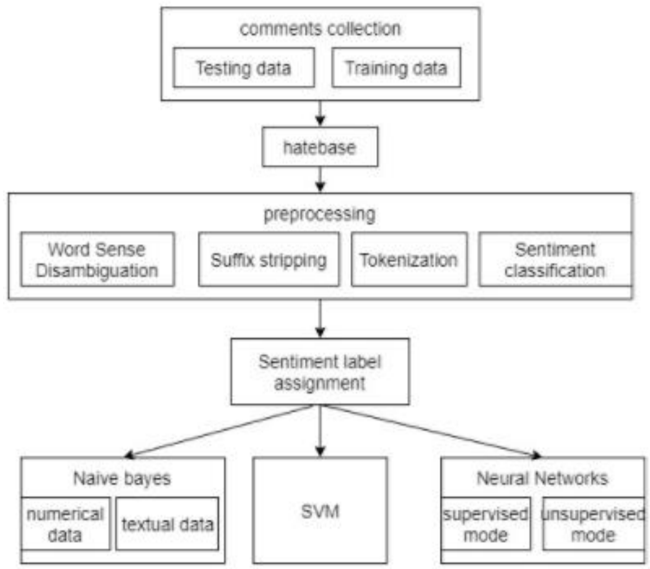
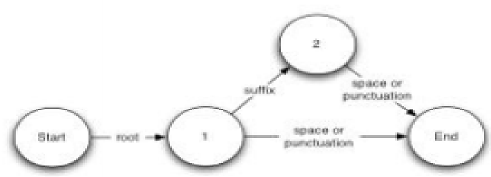


Fig. 1 Methodology

B. Pre-processing algorithms

1. Word Sense Disambiguation: In any language, same words can mean different things with respect to a particular context or reference. For example, the word 'bow' can mean the act of bending forward and it also refers to an archery equipment. To know the exact meaning of the sentence, we need to eliminate the ambiguity. For this purpose, we used the Naive Bayes classifier along with a unique keyword identifier.

2.



3. Fig. 2 Word Sense Disambiguation

4. Tokenization: Tokenization is a process of slicing the data into the smallest possible unit. The Facebook comments are divided into single word units and stored as a csv file. Tokenization has two steps. First, the text is tokenized into sentences. And then the sentences are tokenized into words. Tokenization is followed by stop words removal and lemmatization. For example, consider the sentence, “John Doe is an architect.”. This will produce the following tokens: ‘John’, ‘Doe’, ‘is’, ‘an’, ‘architect’.
5. Stop-words removal: A stop word is a commonly used word like in, the, an, for, of, etc. We certainly do not want these words to take up space in our database or utilize the processor. Hence, we eliminate these words along with the punctuation marks. For this purpose, we maintain a database of characters and words that we consider as stop words based on the document frequency of each word.
6. Sentiment Analysis: Sentiment analysis, also known as opinion mining, is used to find out the opinions and sentiments about some topic. People use microblogging websites like Facebook, Twitter, Instagram to express their opinions. We used Facebook to get our training data using the Facepager application. We extracted the comments and classified them into positive, negative and neutral comments. Using

SVM, Naive Bayes and Neural networks, a comparison of the processing and the accuracy of each algorithm is evaluated.

C) Offensive comments classification

Subjectivity- Subjectivity is a measure which tells us whether the data is subjective or objective. A subjective may or may not express feelings and emotions. For example, “i like monsoons” is a subjective statement and does express some feelings. But the sentence “I want to go home” does not depict any kind of emotions and is still considered to be subjective data.
Polarity- Polarity describes the type of emotions expressed in the data. It tells us if the data is positive, negative or neutral. Generally, the intensity of emotions determines the strength of a sentiment, for example, “1+ series are the best phones available in the mobile market” depicts a positive emotion, whereas “The services provided by Toyota are horrible” showcase negative emotions.

1) Naive Bayes Algorithm:
Our main goal is the classification of comments into offensive and clean i.e. it is a binary classification and partly subjective classification with respect to their sentiment and subject matter. We used the powerful scikit-learn library in Python for this purpose. This library is better than nltk because where nltk only supports Gaussian based Naïve Bayes, scikit-learn supports its multinomial distribution. This library can be downloaded like any other library in Python by simply using the “pip install” command. In addition to scikit-learn, other libraries imported for smooth processing were nltk, csv, numpy, pandas, genism, etc. Pycharm SDK is the platform used for coding.
The Naïve Bayes method is previously known to be an effective machine learning algorithm pertaining to the classification of spam content. It classifies both numerical and textual data. One of its major features is that it believes in independence between any pair of feature points. Some of its main advantages over other classification methods like SVM, Decision trees are higher training efficiency, quicker convergence to solution, comparatively easier implementation and large vocabulary-oriented data handling. In one pass of the testing data, it first computes the conditional probability of individual features with reference to the test dataset. Following that, it applies the Bayes theorem to obtain the posterior probability.

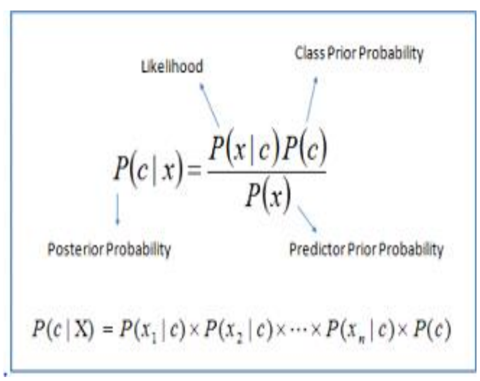
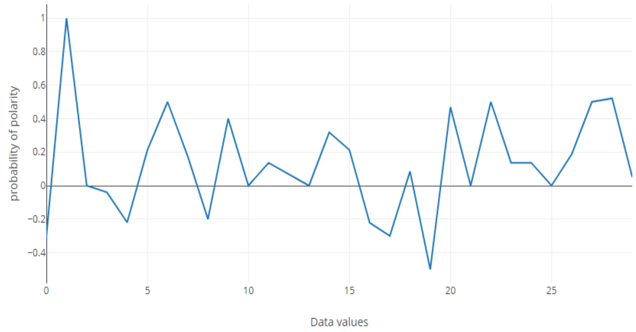


Fig. 3 Naive Bayes equation

Using the Naïve Based Classification method, we could obtain about 79% accuracy. However, it largely depends on the quality and quantity of training and testing data. Once the classification is done, the results can be stored back into a csv file, to be backtracked to the original comment. This method did show some inefficiencies. The Naïve Based classifier will not be as effective for features that are highly dependent like short texts. Further, conditional independence assumption cannot be wholly relied on in real world data.



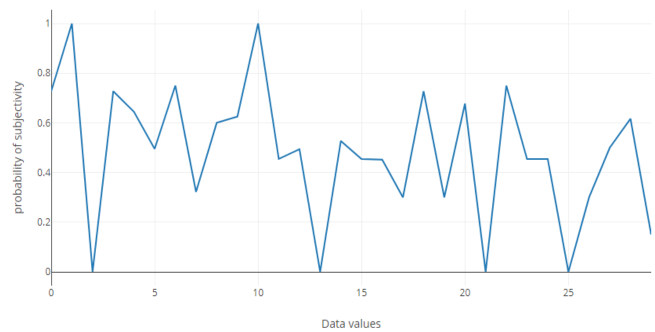


Fig. 4 Polarity using naive Bayes model

Fig. 5 Subjectivity using Naive Bayes

2) Support Vector Machine Algorithm:

It is a machine learning algorithm is used for categorization of huge amount of text. The main motive is to find a hyperplane that separates vectors in one class from vectors in other classes. The technique which is used here is called as the kernel trick. It is used to transform our training data and then based on these transformations, an optimal boundary is found out between the possible outputs. We used support vector machine algorithm to categorize and estimate the strength of positive and negative sentiments of the comments. The experiments are performed on the dataset containing comments of a Facebook page. Naïve Bayes classifier is better than SVM in sentiment classification. Because SVM doesn't perform well when the dataset is too large or if it contains noise. The preliminary results don't seem to be promising as it gives an accuracy of 58% for this training dataset which less than the accuracy percentage of Naïve Bayes Algorithm.

TABLE I
SENTIMENT LABELING USING SVM

Text	Polarity
@Kelsie_love really, my boyfriend isn't going to get any siblings? whatevas, you're totally going to have another one, bitch. FRODO! lbha	Negative
@littledutty nope not my job, not heard anything from that nor my assignment results im expecting just about life really its poo	Negative
@teesdejong too bad its not M'i bday party! hehe	Negative
Is working and then wedding at 2 yippy skippy	Neutral
Oh. Wow. dont look at people or scream cowor olberst	Neutral
@xia_hime i'm going to ayal gd a room onsite though ac cant hotel skare	Neutral
Finished garden for today. Hey fever broke out finally	Negative
@selby_peeks I kind of have to, for myself you don't like me though sweet dreamslove xoxoxoxoxox	Positive
@drewryanscott nope but i want one to	Neutral
@UNO_OUT what about me jay I want a keychain	Neutral
@thepistol good luck with ur concert in san diego! wish i could be there but i live all the way in buffalo i know u'll do greatttho! duh.	Positive

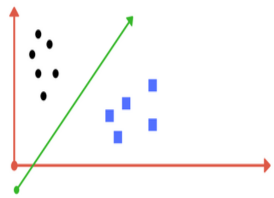


Fig.6 Sample cut to divide into two classes

The performance parameters after performing sentiment analysis on the training data uses the following formula:

TABLE II
PERFORMANCE PARAMETERS

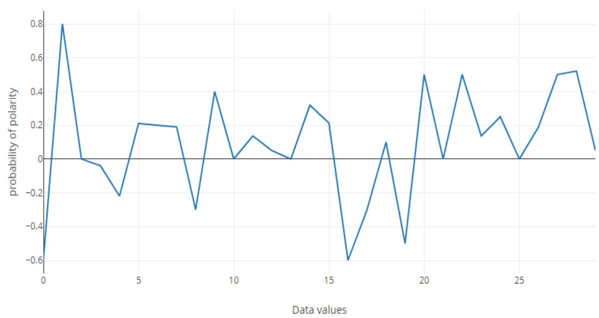
		Correct labels	
		Positive	Negative
Classified labels	Positive	True Positive(TP)	False Positive(FP)
	Negative	False Negative(FN)	True Negative(TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 * Precision * Recall}{Precision + Recall}$$



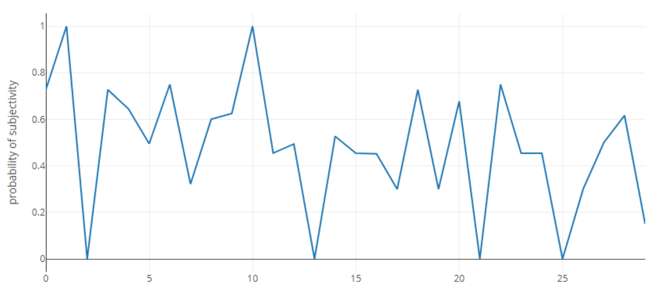


Fig. 7 Polarity using SVM

Fig. 8 Subjectivity using SVM

3) Neural Networks:

Neural network is one of the many techniques used in machine learning and is also known as Artificial Neural Networks (ANN). The main element of ANN is the novel structure of the information processing system, which is influenced by the way information is processed by biological nervous systems and the brain. It is made of an enormous number of highly interconnected units known as 'neurons', processing in coordination to execute a particular task. Artificial neural networks learn through examples, just like people. For eg, pattern recognition and data classification.

There are various deep neural network models available for data classification. We made use of the unsupervised model for our dataset. The main benefit of using the unsupervised mode is word2vec, that enables us to create a low dimension distributed representation of data. Skip-gram and continuous bag of words are the two popular models used in this mode. We decided to go to with skip-gram as it has a very simple and straightforward design. The primary logic behind skip gram is that it considers each word in a huge corpus and simultaneously also takes one word which surrounds it within a specific defined 'window'. It then trains the neural network in such manner that it will predict the probability of each word to occur in the window around the focus word. To make things convenient, we create a vocabulary of various words available in our data set. We then encode this data as a vector that has the same dimensions as our vocabulary. For eg, if we have a vocabulary made out of the words "the", "big", "black", "bear", "eats", "the" "white", "goat", the word "bear" is represented by this vector: [0, 0, 1, 0, 0, 0 ,0 ,0].
Based on predicting the nearby training words, a model depiction of current word is generated in the training process. After training, the word embedding is created which is the vector of weights from the hidden layer.

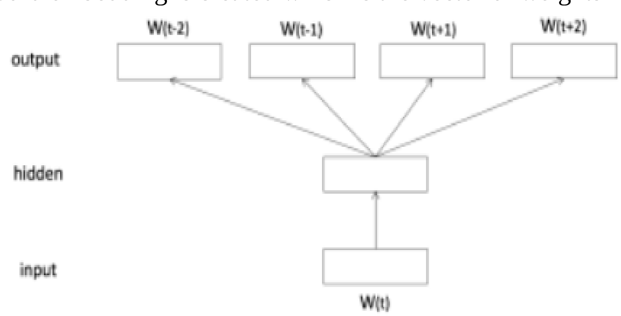


Fig. 9 Primary logic behind Neural Networks

W represents the set of vectors containing the words and the output layer predicts the next word in the scenario. Since the input is now ready, we use it to enter into the 2-layer neural network model. The probability for each text in the vocabulary will be displayed in the second layer.
The unsupervised mode gives us approximately an accuracy of 58% when we use the skip gram model. The Skip gram model reduces 39% of the errors that occur during the implementation Continuous Bag of Words Model.

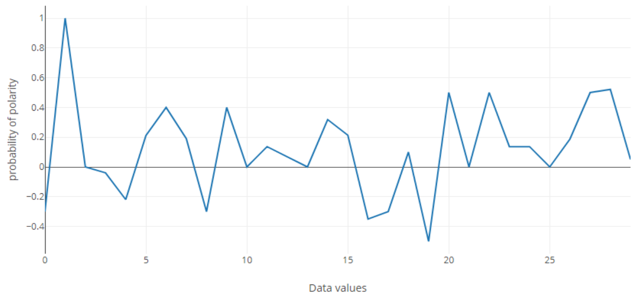


Fig. 10 Polarity using Neural Networks

Fig. 11 Subjectivity using Neural Networks

IV. RESULTS

A) Wordcloud

The fundamental idea of creating a word cloud is to represent a set words on a canvas, where its size depends on its importance. In our case, greater the size indicates higher degree of profanity usage. Word clouds, although a little old-fashioned, gave an interesting visualization of our model. In Python, word cloud can be generated with aid from some libraries like nltk, CountVectorizer or with a regular expression. We have used the Wordcloud library. First, we read the data in the csv format. Our goal is to randomly sample a part of the dataset and display it on a canvas based on its frequency, while making sure that the words don't overlap each other at any point. For plotting the Wordcloud, we used the matplotlib library.

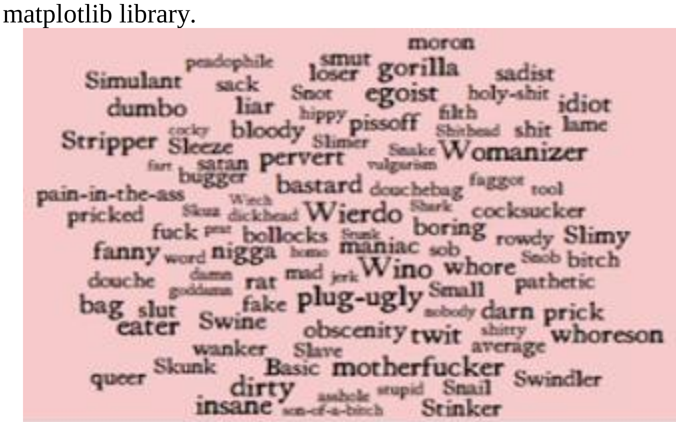


Fig. 12 Word cloud

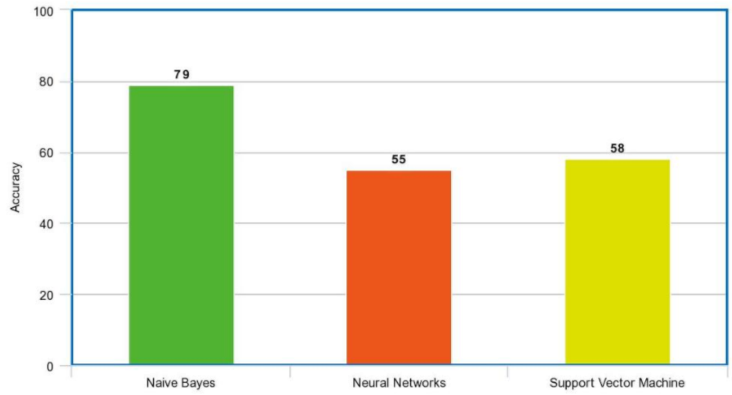


Fig. 13 Types of models

V. CONCLUSION

The emotional consequences that the preys of cyberbullying suffer can be disastrous and mortifying. It is even worse than face-to-face bullying, as the victim has no idea of who the bully is. On examining the computer forensic process of obtaining digital evidence from social media, and the legal aspects of such cases of cyberbullying, three models were used on the training dataset i.e. Naive Bayes, Support Vector Machine and Neural Networks. The accuracy obtained by Naïve Bayes Classification method was 79%, whereas SVM offered an accuracy of 55% and 58% of accuracy was achieved by the Artificial Neural Network Model. Thus, it is clear that the Naïve Bayes approach is the most efficient one and is therefore the best classifier for sentiment analysis, out of the three models chosen. The evidence provided by the application can be used by the users i.e., bloggers or any individual or organisation to report the crime to the cybercrime department for further legal actions.

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for supporting our project and extending help whenever required. We extend our immense gratitude to our mentor Assistant Professor, Mrs. Abha Tewari for her kind help and valuable guidance. We wish to express our profound thanks to all those who assisted us in the information gathering of this paper

REFERENCES:

[1] Forensic investigation of social networking applications. -Dr Mark Taylor, Dr John Haggerty, David Gresty, Peter Almond, Dr Tom Berry
[2] Using Naïve Bayes Algorithm in detection of Hate Tweets. -Kelvin Kiema Kiilu, George Okeyo, Richard Rimiru, Kennedy Ogada
[3] Mandola Monitoring and Detecting Online Hate Speech -D. Stefanidis, D. Paschalides, G. Pallis
[4] Sentiment Analysis on Twitter Data Using Support Vector Machine. -Bholane Savita D., Prof.Deepali Gore
[5] Analysis of Various Sentiment Classification Techniques -Vimalkumar B. Vaghela, Bhumika M. Jadav
[6] Sentiment Classification using Machine Learning Techniques. -Suchita V Wawre, Sachin N Deshmukh
[7] Using Machine Learning Techniques for Sentiment Analysis -Oscar Romero Llombort
[8] Social Media Sentiment Analysis using Machine Learning Classifiers -Bharat Naiknaware, Bindesh Kushwaha, Seema Kawathekar