Social Bulling Identification Using Sentiment Analysis

Submitted By

Kejal Naik

18MCEC04

तमसो मा ज्योतिर्गमय

NIRMA UNIVERSITY

INSTITUTE OF TECHNOLOGY

NAAC ACCREDITED ‘A’ GRADE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2019

Social Bulling Identificatin Using Sentiment Analysis

Major Project

Submitted in fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering

Submitted By

Kejal Naik

(18MCEC04)

Guided By

Prof. Malaram Kumhar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

December 2019

Certificate

This is to certify that the major project entitled **”Social Bulling Identification Using Sentiment Analysis”** submitted by **Kejal Naik (18MCEC04)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad, is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project part-I, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Malaram Kumhar
Guide & Assistant Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Priyanka Sharma
Professor,
Coordinator M.Tech - CSE (CSE)
Institute of Technology,
Nirma University, Ahmedabad

Dr. Madhuri Bhavsar
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Kejal Naik**, **18MCEC04**, give undertaking that the Major Project entitled "**Social Bulling Identification Using Sentiment Analysis**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

Signature of Student
Date:
Place:

Endorsed by
Prof. Malaram Kumhar
(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof Malaram Kumhar**, Assistant Professor, Computer Engineering Department, Institute of Technology, Nirma University, Ahmedabad for her valuable guidance and continual encouragement throughout this work. The appreciation and continual support she has imparted has been a great motivation to me in reaching a higher goal. Her guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Madhuri Bhavsar**, Hon'ble Head of Computer Science And Engineering Department, Institute of Technology, Nirma University, Ahmedabad for her kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Kejal Naik
18MCEC04

Abstract

Today social Media is used by more and more people using so many platform. People used to express their feeling about product, movie as well towards individual or group of people. As social media is very important and power full platform to connect people all over the world, it also introduce cyber crime like social Bulling. Social Bulling impact physical and mental condition of victim. Successful identification of message which lead to social bulling can prevent many people from become victim of this crime. The focus of this paper is to use sentiment analysis technique to identify bully comment on social media. Sentiment analysis is used to identify the polarity of sentence. In this paper we discussed about three different method of sentiment analysis which we will use for social bulling identification. Here we discuss different approach for text pre-processing, feature extraction which is use full to get good performance from social bulling identification model.

Abbreviations

SE	Sentiment Analysis.
BOW	Bag Of Word.
NLP	Natural Language Processing.
NLTK	Natural Language Tool Kit.
TF-IDF	Term Frequency-Inverse Document Frequency .
SVM	Support Vector Machine
NB	Naive Bayes

–

Contents

List of Figures

Chapter 1

Introduction

As availability and use of internet is increase, today people use many social media platforms in their daily life like facebook, what's up , and twitter. People use this platform to share their opinion , comment on and share their feelings using emoticons and text. As social media give so many advantages it introduce some cybercrime like cyberbullying.

As increasing the use of social media, every day huge amount of data is generated on it. people are knowingly or unknowingly put their view about some product as well as people or people behavoir on social media which some time lead to cyber crime like cyberbulling. Cyberbullying is a type of bullying which take place using electronic devices like cell phones, computers through text messages , chats using social media. Text messages, rumors that can be embarrassing to concern people can be considered cyberbullying. Platform form which social bullying can take place include chat rooms, social Media and gaming platform from where people can participate in the sharing of content.

Cyber bullying causing humiliation through spreading hateful comments on messaging app or on other online platform. It contains posting , sharing or shanding negative or incorrect information about some individual in intention of humiliation.

Research present that many teenagers face cyberbullying at time of using social media platform. 20% to 40% of teenagers become victims of cyberbullying. Cyberbiulling can be deleted by identify harmful word. It needs an intelligent system that can identify harmful words present in message or post from which can help to identify cyberbullying.

Objective of this project is to develop a model which can identify content as positive, negative or neutral.

Chapter 2

Literature Survey

2.1 Literature Summary

Paper Title	**Year**	**Type**	**Author**	**Summary**
Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis	2017	Paper	Zhao Jianqiang, Gui Xiaolin	This artical provide basic knowledge about preprocessing technique used for sentiment analysis. It discribe various type of noice present in text and method to remove noice from text. For accurate result of sentiment analysis pre-processing is very helpful.
Sentiment Analysis of News Articles: A Lexicon based Approach	2019	Paper	Soonh Taj,Baby Bakhtawer Shaikh, Areej Fatemah Meghji	This paper introduce Lexicon based approach for sentiment analysis. Lexicon based approach is worked based on lexicon score. In thia approach text is divided in to token and this token is compared with predeveloped dictionary. Dictionary contain all possible lexicon with corresponding score. After comparing text with dictionary, as depend of matching one lexicon is catogarize as posite, negative or neutral.

Table 2.1: Literature summary

Paper Title	Year	Type	Author	Summary
Approaches for Sentiment Analysis on Twitter: A State-of-Art study	2018	Paper	Harsh Thakkar, Dhiren Patelr	This paper present three different approach for sentiment analysis. Lexicon approach used lexicon score calculation approach to find polarity of text. This approach use lexicon dictionary is used to compare token and calculate score of sentence. In machine learning based approach, after text preprocessing and feature extraction different machine learning algoritham is used for text classification.
Sentiment Analysis of Tweets Using Machine Learning Approach	2018	Paper	Megha Rathi, Aditya Malik, Daksh Varshney, Rachita Sharma, Sarthak Mendiratta	In this paper author introduce hybrid approach for sentiment analysis. TF-IDF is used for feature extraction. After feature extraction SVM is applied on it. In hybrid approach after applying SVM diction tree is apply to get more accurate result of sentiment analysis. .

Table 2.2: Literature summary

Paper Title	Year	Type	Author	Summary
Sentiment analysis on facebook group using lexicon based approach	2016	Paper	Sanjida Akter, Muhammad Tareq Aziz	This article provides basic knowledge about sentiment analysis. Lexicon based approach is used for sentiment analysis.
Sentiment expression via emoticons on social media	2015	Paper	Hao Dong, Guang Yang, Fangde Liu, Yuanhan Mo, Yike Guo	This paper has introduced importance of emoticons for sentiment analysis. Social Media data contain text as well as different emoticons to express filling. This emoticons help to identify polarity of sentence more accurately. This paper many emoticons with their meaning and category in which they belong ex: positive or negative, sad or happy or angry.

Table 2.3: Literature summary

Chapter 3

Sentiment Analysis

3.1 Basics of Sentiment analysis

Sentiment analysis is the automated technique which is used for analyzing text data to identify the polarity of content. Sentiment analysis. sentiment analysis is also named as opinion mining, sentiment mining, opinion extracting, subjectively analysis, review mining and emotion analysis.As the use of social media, social blogging and online forums are increase, sentiment analysis use is also increase to make lots of research on data to get some interesting and useful result. Sentiment analysis used to understand an opinion of people about some subject or people using written or spoken language. Sentiment analysis has an ability to to extract the opinion, sentiment, attitude, emotion or view form data. With using sentiment analysis we can classify text into following category:

- Positive
- Negative
- Neutral

Sentiment analysis is a technique which provides methods to extract emotions and classify it into three different classes. Sentiment analysis is applied in different domain as described below.

- Social Media Monitoring
- Business analysis

- Customer review analysis
- News and blog analysis
- Product analysis using social media and public review

Sentiment analysis is mainly focused on two things:

- Identify that the textual entity is objective or subjective
- Polarity of text

sentiment analysis is perform on two different level. one is document level and other one is sentence level. In document level sentiment analysis, scan is perform on hole document and document is fully categorize as positive or negative. while in sentence level sentiment analysis scan should be perform on every sentence and sentence should be categorize as positive, negative or natural sentence.

3.1.1 Technique used for Sentiment Analysis

Sentiment Analysis used different approach which has it‘s own benefits. Sentiment Analysis use three technique to identify polarity of text :

- Lexicon based Approach
- Machine Learning based Approach
- Hybrid Approach

Lexicon Based Approach

This technique use dictionary which contain pre-tagged lexicon. First of all input text is converted into token using Tokenizer.[3] Then every Token is matched with the token present in dictionary. If token is match with any token present in dictionary, corresponding score is added to the total score of input text. Example : if input text contains “fantastic” as a token in it , it will match with positive token in dictionary and input text score is increased accordingly. On based on score input text is considered as positive otherwise negative .

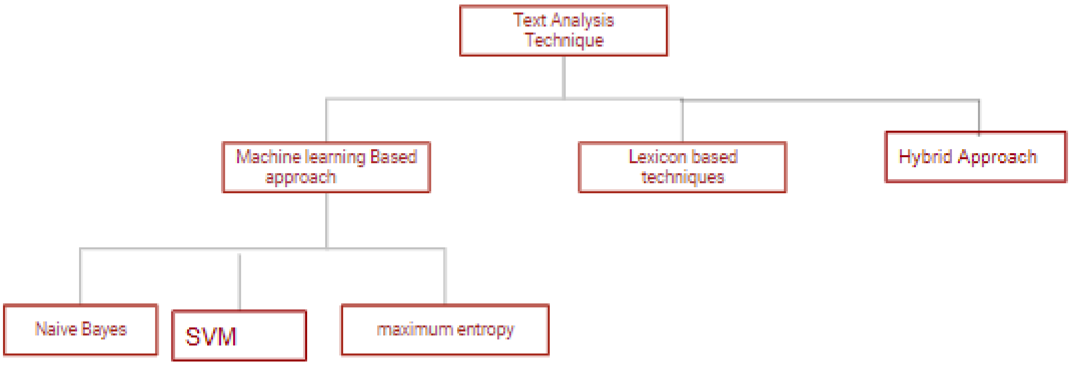


Figure 3.1: Technique for Sentiment Analysis

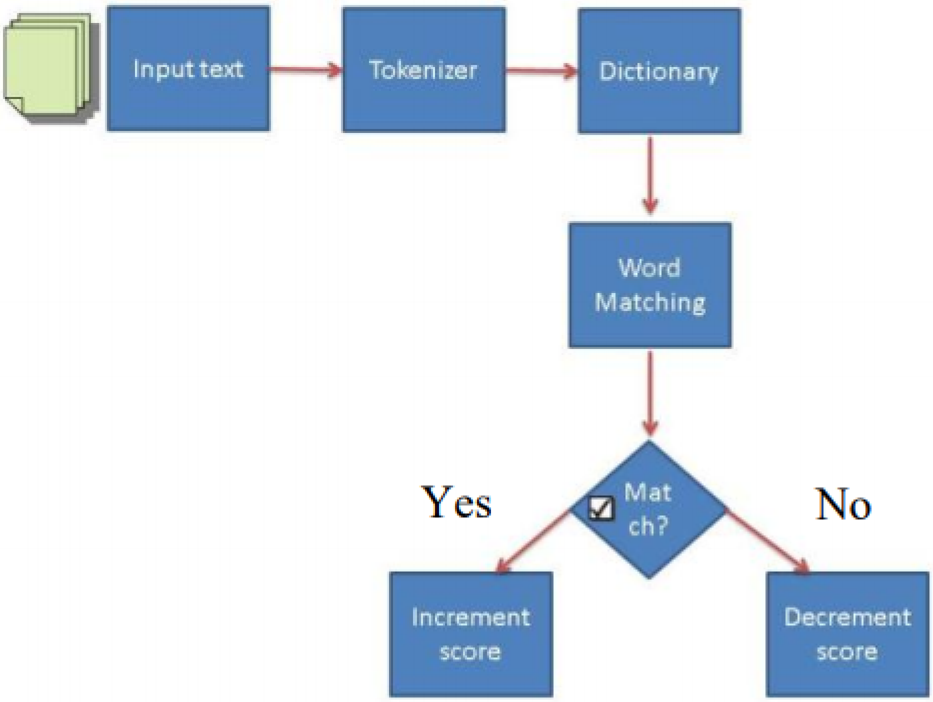


Figure 3.2: Technique for Sentiment Analysis [12]

Machine learning based Approach

Machine learning is more popular as it provides more speed and accuracy compared to other techniques. Machine learning provide two main approach supervised approach and unsupervised approach. Sentiment analysis used supervised approach for identifying polarity of content. In this technique labeled token is used as dataset for training purpose.

Hybrid Approach

Hybrid approach is the combination of machine learning and lexical approach as it provides advantages of both [12]. It provides more accuracy as machine learning approach and speed as lexical approach. Many research is performed to make this approach working. [12] In this, two word lexicon and unlabeled data is used. Firstly they divide two-word lexicon into two different classes as positive and negative. Training document encompassing all words from the lexicon set which is chosen and created. On the basis of this calculate the entropy of the testing document and according to this value, training document is considered positive or negative. Then this training document is pass to machine learning classifier with a view to train the model.

3.1.2 Flow of Sentiment Analysis

The Flow of Sentiment Analysis is illustrated in figure . After the data collection, data cleaning and prepossessing take place.

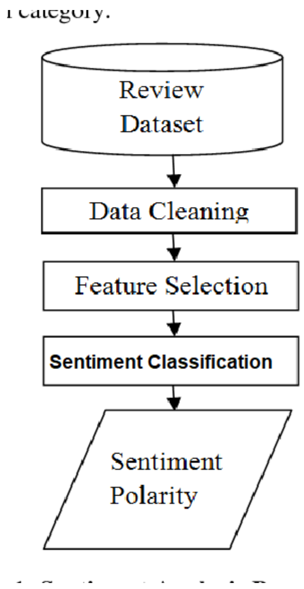


Figure 3.3: sentiment-analysis-flow

Data Collection

tweeter data is stored using tweeter API and stored in to CSV files. Training data set contain tweets with sentiment either 0 (indicate negative tweet) or 1 (indicate positive tweet) . Test dataset or testing tweet contain only tweet with out any sentiment.

Dataset is contain different type of data like words, URLs, reference to different people, symbols and emoticons. In this variety of data word and emoticons are use full to get sentiment form it.

Pre-processing

Raw data contain lots of noise. This noise is the result of casual nature of people on social Media. Tweets contain certain characteristic as it contain user mention, retweets and emoticons, repetition of alphabet in word. we have to apply number of pre-processing steps to normalize the text and possibly reduce the size. pre-processing of raw data contain following basic steps

- replace more than two dots with space

  As human habit many social media contain text having many dot (abc....) , so have to remove all extra text dots.

- replace more than two space with one single space

  As social media data is the platform where people put their text. This text represent many human habit to represent the text. It is possible to get a data which contain more than one space between words. This extra extra space is not use full so have to remove extra space and manage data with single space between words.

- convert text to lower space

  To handle all the data equally , one of the step of data-prepossessing is to conver all data into lower case.

There are some special feature need to take care for pre-processing of text:

Handle URL

In social Media, many user used to mention url related to specific content to explain some particular topic. This url are contain general address related to page but it is not use full to identify sentiment of any text. so we remove this url or replace hole url with word URL. To perform this operation regular expression we used is ((www[̇\S]+)\(https?://[\S]+)) and replace it with word URL or remove this url[10].

Handle User Mention

In many content of social media contain handle with it. User used to mention other user name with @ tag to define related user. we have to ignore user mention. To perform this operation regular expression e used is @[\S]+ and we have to remove it from data[7].

3.1.3 Handle HashTag

Hashtags are unspaced expressions prefixed by the hash image () which is habitually utilized by person. to make reference to a drifting subject on social Media. We supplant all the hashtags with the words with the hash image. For instance, topten is supplanted by topten. To perform this operation regular expression we used is (\S+)[1].

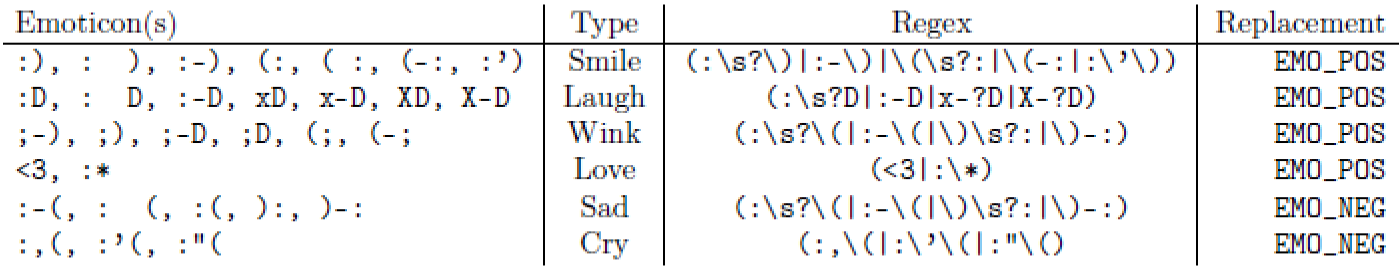
3.1.4 Handle Emoticons

In social media people used to express their with text as well as with many emoticons. Emoticons is the word stand for 'emotion + icon'[13]. In the era of internet emoticons provide another way to express human feeling. emoticons is the group of some symbols which stands to express feeling like happy, sad, angry . As it is one of the visual way to express the feeling people like to use them on social media.

Emoticons play very important role to identify sentiment of the text. It is not possible to identify all the emoticons, but we can use some of the common and most used one to identify sentiment of the sentence. Here is the list of different emoticons which are used most commonly used. we have to identify them and assign lexicon score accordingly or have to replace them with EMO_POS [9] or EMO_NEG.

3.1.5 Retweet

In retweeting the current user is going to repost the text which is already posted by some one else. We have to remove this retweet as it is not to be consider as text classification.[7]



<table>
<tr><th>Emoticon(s)</th><th>Type</th><th>Regex</th><th>Replacement</th></tr>
<tr><td>:), :), :-), (:, (:, (-:, :')</td><td>Smile</td><td>(:\s?\)|:-\)|\(\s?:|\(-:|:\'\))</td><td>EMO_POS</td></tr>
<tr><td>:D, : D, :-D, xD, x-D, XD, X-D</td><td>Laugh</td><td>(:\s?D|:-D|x-?D|X-?D)</td><td>EMO_POS</td></tr>
<tr><td>;-), ;), ;-D, ;D, (;, (-;</td><td>Wink</td><td>(:\s?\(|:-\(|\)\s?:|\)-:)</td><td>EMO_POS</td></tr>
<tr><td><3, :*</td><td>Love</td><td>(<3|:*)</td><td>EMO_POS</td></tr>
<tr><td>:-(, : (, :(,):,)-:</td><td>Sad</td><td>(:\s?\(|:-\(|\)\s?:|\)-:)</td><td>EMO_NEG</td></tr>
<tr><td>:,(, :'(, :"(</td><td>Cry</td><td>(:,\(|:\'\(|:"\()</td><td>EMO_NEG</td></tr>
</table>

Figure 3.4: frequently used emoticons

3.1.6 Handle punctuation

With a view of get clear text for classification, have to remove any punctuation like (,!?.():;') present with word in text[1].

3.1.7 Handle repeating letter

On social we find many word with repeating letters in it. People use to repeating letter their feeling like to describe more happiness people use to write "happyyyyyyy" instead of "happy". we have to remove all extra letter present in word .

3.2 Feature Extraction

After data prepossessing, next step of sentiment analysis is feature extraction. from social media we get input in form of natural language. so after pre-process the text we have to extract the feature to identify correct sentiment form it. There are many different approach used for feature extraction like count vector, BOW, TF-IDF, NLP based techniques[9].

3.2.1 TF-IDF

To know the importance of word in sentence or data TF-IDF technique is used. we can use this technique as , after pre-processing we are able to get the list of use full word from the sentence.[9] To calculate the term frequency of any word, have to calculate number of occurrence of particular word in text against number of word present in text.some time

IDF is also calculate as log(NO/NODF)[2], where NO represent number of occurrence or word, NODF represent number of word present in document. TF-IDF provide better option to present text information into Vector Space Model(VSM)[2].

3.2.2 N-Gram

N-Gram is the feature extraction technique mostly used with supervised machine learning[2].In this technique all token are divided as sequence of n-token and treated as single feature. Most frequently used N-gram category are as follow:

- Uni-Gram

  Most commonly used way of feature extraction in text classification is treat every single word or token as feature for classification[2]. In this approach we pic single word form training dataset and perform frequency cont for it.It is more convenient to use this technique after removal of noise as frequency count of stop word is considering very high but this words are not use full for Sentiment analysis[2]. Unigram provide good result but only issue with uni-gram method is that some time it lead to wrong sentiment Identification because of negation.

- BiGram

  In bigram is a pair of word or token which occur in sequence in text are consider as single feature. Biagram overcome limitation of unigram. By using bigram we can handle negation present in text[2].

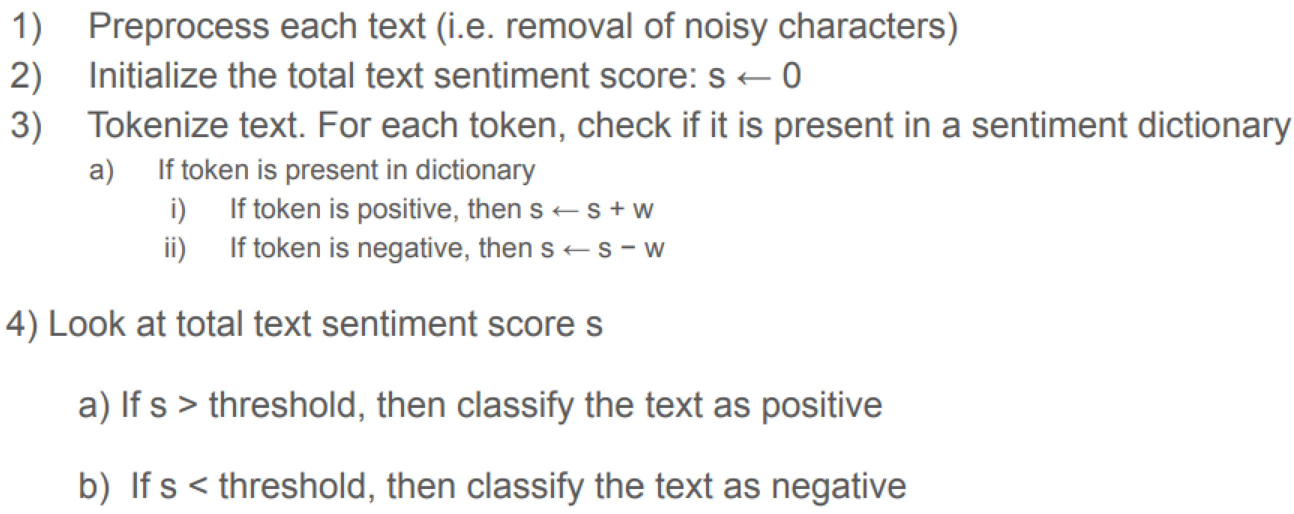
3.3 sentiment Classification

Sentiment Classification is done by using different sentiment Analysis Technique which discussed above. In This project we are going to compare sentiment analysis done by using Lexicon based based approach and machine learning based approach.
After feature extraction, list of feature is passed for sentiment classification.

3.3.1 Classification using Lexicon based Approach

In this step score of lexicon is count. on basis of this score text is classify as positive or negative token. Token word is compare with word present in dictionary, if it match with any positive word in dictionary then score will be consider as +1 and if it match with negative word in dictionary then score will be consider as -1, else it will be score as 0.

Total score will count as : Score = Pos(x, ”pos word”) Neg(x, ”neg word”). following is the basic algorithm used in lexicon based approach.



```
1) Preprocess each text (i.e. removal of noisy characters)
2) Initialize the total text sentiment score: $s \leftarrow 0$
3) Tokenize text. For each token, check if it is present in a sentiment dictionary
    a) If token is present in dictionary
        i) If token is positive, then $s \leftarrow s + w$
        ii) If token is negative, then $s \leftarrow s - w$
4) Look at total text sentiment score s
    a) If s > threshold, then classify the text as positive
    b) If s < threshold, then classify the text as negative
```

Figure 3.5: Lexicon based approach Algorithm

3.3.2 classification Using Machine Learning based Approach

Machine Learning provide many different classifier to identify polarity of sentence. Machine learning used different type of approach to train the model like supervised and unsupervised. Here sentiment analysis mostly use supervised machine learning approach to train the model. In Machine learning based approach two part of dataset it used. Train dataset and test dataset. While using different classifier train datasdet is used to train the machine learning model and test dataset is apply as input to that train model to test correctness and accuracy of the model.

In training phase labeled data is used in training dataset, which is passed to classifier to train it and then test data set(unlabeled dataset) is passed as input to classifier to test it. There are many classifier which are used for sentiment analysis are naive Bayes, SVM, KNN .

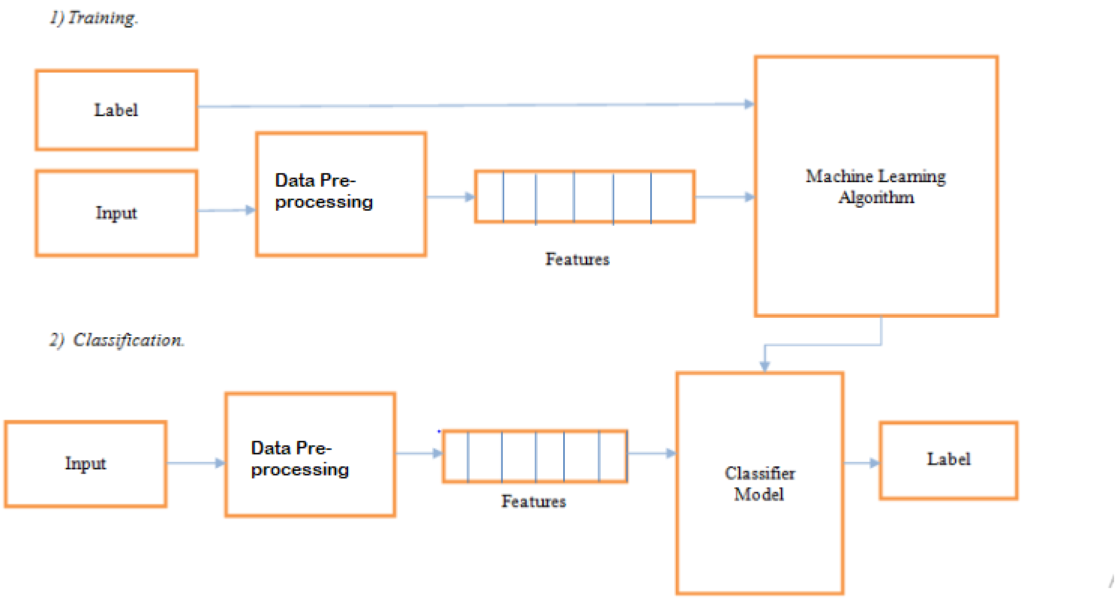


Figure 3.6: Machine Learning Model for SA

Chapter 4

Comparative Study of Existing Data

Paper Title	**Classifier**	**Accuracy**
A Lexicon-based Approach for Hate Speech Detection	Lexicon Based Approach¸	63.7 %
Predicting the Effects of News Sentiments on the Stock Market	Lexicon Based Approach [11]	70.59 %
Sentiment analysis for the news data based on the social media [11]	Naive Bayes	68.60 %
Deep Convolution Neural Networks for Sentiment Analysis of Short Texts[5]	Naive Bayes	82.7 %
	Maximum Entropy	83%

Table 4.1: Comparative Study

Paper Title	**Classifier**	**Accuracy**
Hate me, hate me not: Hate speech detection on Facebook [4]	SVM	74.61 %
	LSTM	70.4 %
Twitter Sentiment Classification using Distant Supervision [6]	Naive Bayes	81.3 %
	Maximum Entropy	80.5
	SVM	82.2 %
Sentiment Analysis of Tweeter data - survey of Technique [8]	Naive Bayes	76.44 %
	SVM	76.68 %
	Maximum Entropy	74.93 %
	Logistic Regression	73.65 %

Table 4.2: Comparative Study

Chapter 5

Implementation

Goal of this project to perform sentiment Analysis on social media data to identify sentiment of the text and identify that given text contain any bully comment or not. To achieve this goal here we aim to implement both technique of sentiment analysis to identify which perform better in context of social bulling identification.

Here we divide project into two part. In first half, implement system which identify bully comment using lexicon based approach of SA. In second half we are going to use different machine learning library to process the text and use different machine learning classifier to perform sentiment analysis to identify weather the given text is bully comment or not.

5.0.1 Flow of Implementation:

Following figure describe hole work flow of Implementation:5.1

5.0.2 Implementation of Lexicon Based Approach:

As this approach work on calculating lexicon score by using dictionary. So first task of this approach to prepare dictionary which contain all possible positive and negative as well as emoticons. 5.2

following figure shows implementation of Lexicon based approach for social bulling Identification:

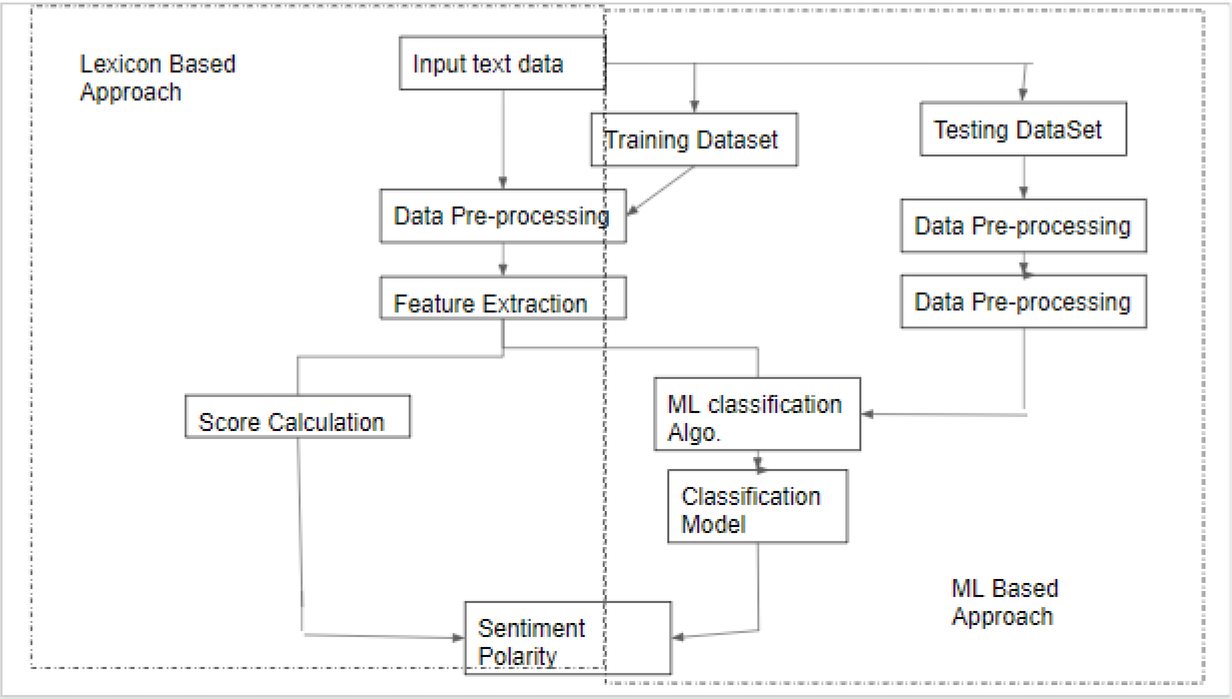
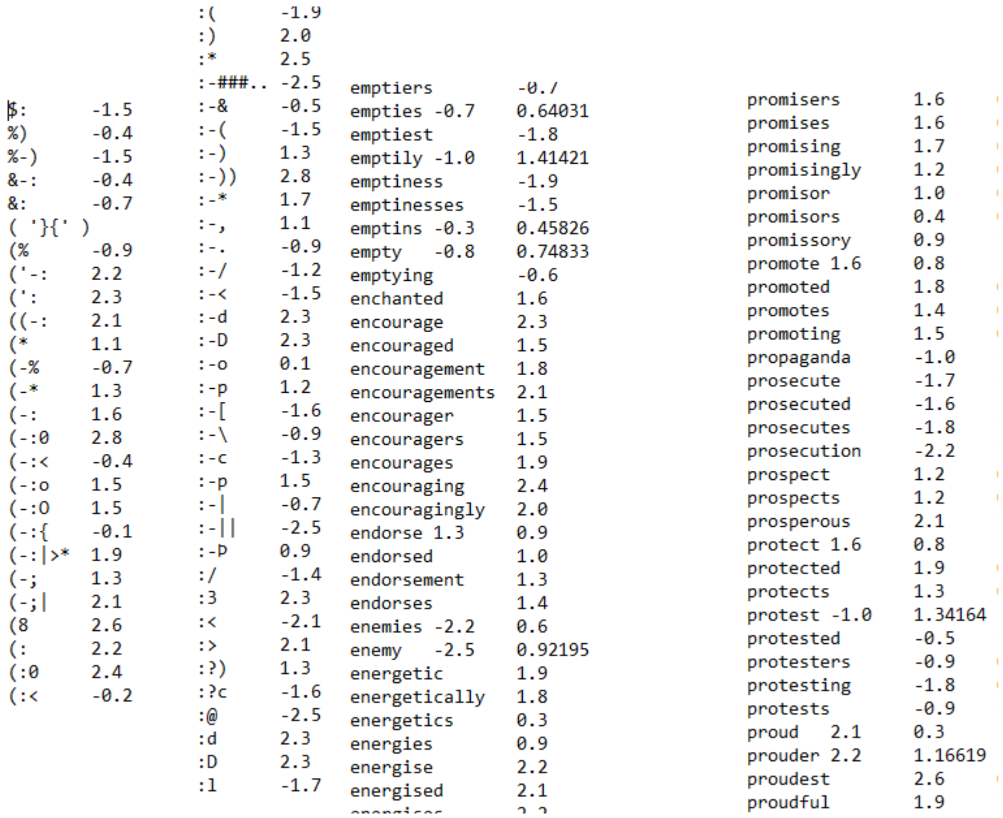


Figure 5.1: Flow of implementation



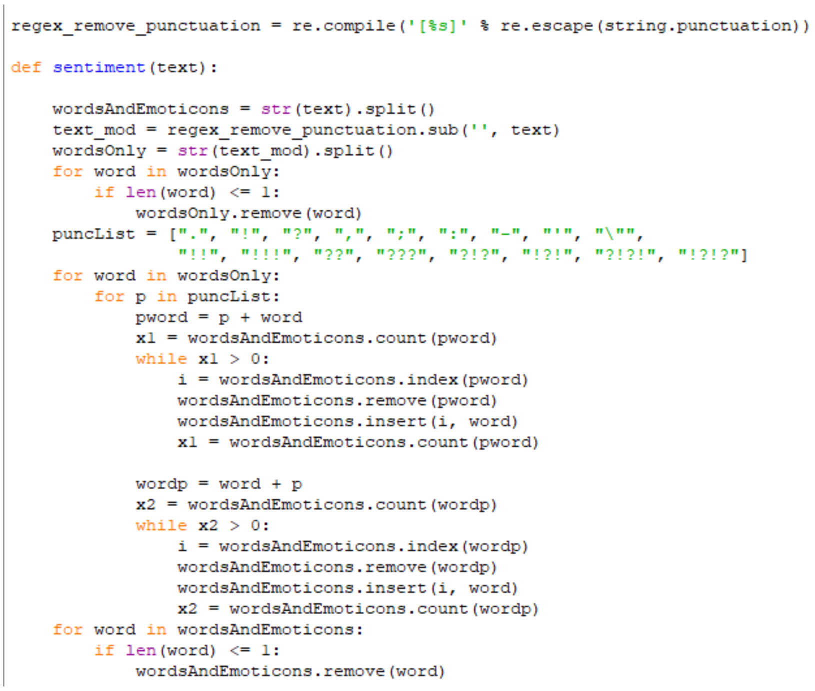
```
$:      -1.5
%)      -0.4
%-)     -1.5
&-:     -0.4
&:      -0.7
( '}{' )
(%      -0.9
('-:    2.2
(':     2.3
((-:    2.1
(*      1.1
(-%     -0.7
(-*     1.3
(-:     1.6
(-:0    2.8
(-:<    -0.4
(-:o    1.5
(-:O    1.5
(-:{    -0.1
(-:|>*  1.9
(-;     1.3
(-;|    2.1
(8      2.6
(:      2.2
(:0     2.4
(:<     -0.2
```

```
:(      -1.9
:)      2.0
:*      2.5
:-###.. -2.5
:-&     -0.5
:-(     -1.5
:-)     1.3
:-))    2.8
:-*     1.7
:-,     1.1
:-.     -0.9
:-/     -1.2
:-<     -1.5
:-d     2.3
:-D     2.3
:-o     0.1
:-p     1.2
:-[     -1.6
:-\     -0.9
:-c     -1.3
:-p     1.5
:-|     -0.7
:-||    -2.5
:-Þ     0.9
:/      -1.4
:3      2.3
:<      -2.1
:>      2.1
:?)     1.3
:?c     -1.6
:@      -2.5
:d      2.3
:D      2.3
:l      -1.7
```

```
emptiers        -0.7
empties -0.7    0.64031
emptiest        -1.8
emptily -1.0    1.41421
emptiness       -1.9
emptinesses     -1.5
emptins -0.3    0.45826
empty   -0.8    0.74833
emptying        -0.6
enchanted       1.6
encourage       2.3
encouraged      1.5
encouragement   1.8
encouragements  2.1
encourager      1.5
encouragers     1.5
encourages      1.9
encouraging     2.4
encouragingly   2.0
endorse 1.3     0.9
endorsed        1.0
endorsement     1.3
endorses        1.4
enemies -2.2    0.6
enemy   -2.5    0.92195
energetic       1.9
energetically   1.8
energetics      0.3
energies        0.9
energise        2.2
energised       2.1
```

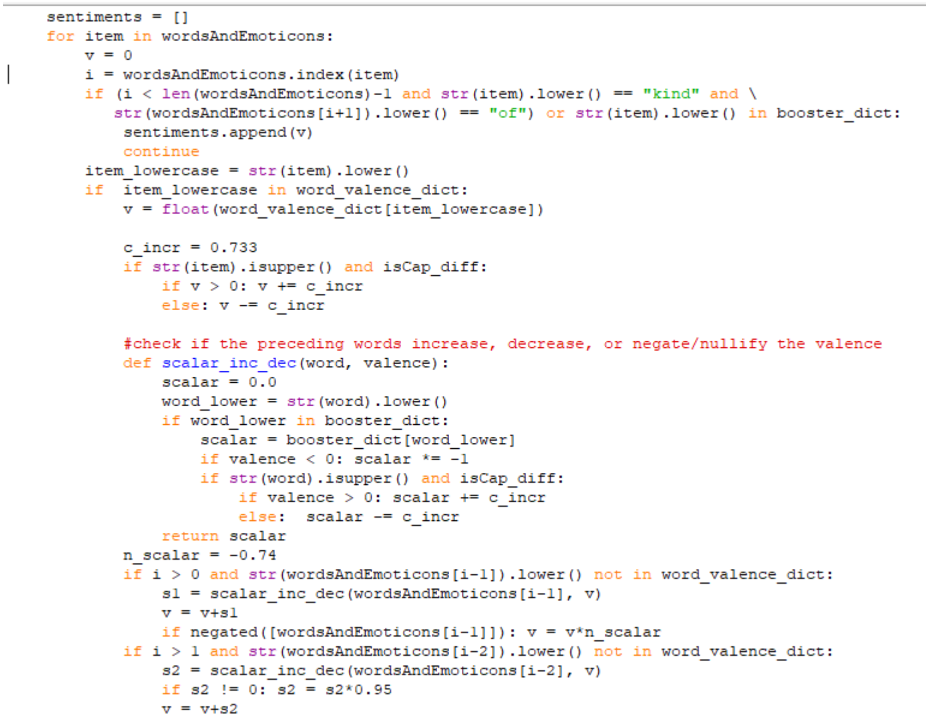
```
promisers       1.6
promises        1.6
promising       1.7
promisingly     1.2
promisor        1.0
promisors       0.4
promissory      0.9
promote 1.6     0.8
promoted        1.8
promotes        1.4
promoting       1.5
propaganda      -1.0
prosecute       -1.7
prosecuted      -1.6
prosecutes      -1.8
prosecution     -2.2
prospect        1.2
prospects       1.2
prosperous      2.1
protect 1.6     0.8
protected       1.9
protects        1.3
protest -1.0    1.34164
protested       -0.5
protesters      -0.9
protesting      -1.8
protests        -0.9
proud   2.1     0.3
prouder 2.2     1.16619
proudest        2.6
proudful        1.9
```

Figure 5.2: example of content store in Dictionary



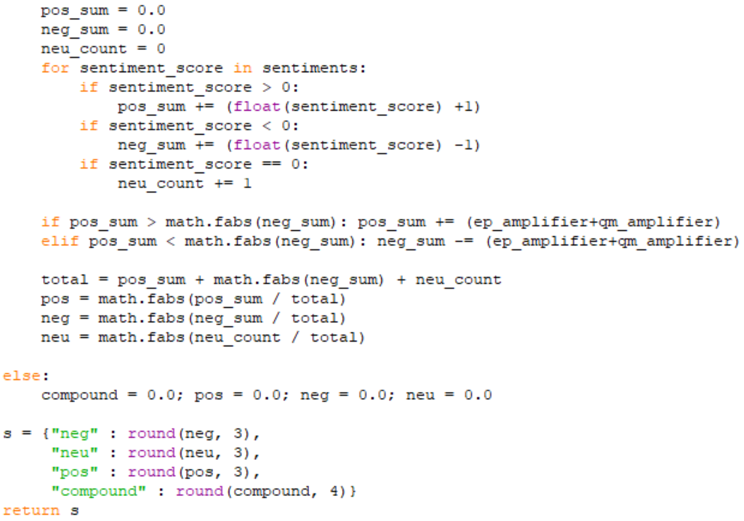
```
regex_remove_punctuation = re.compile('[%s]' % re.escape(string.punctuation))

def sentiment(text):

    wordsAndEmoticons = str(text).split()
    text_mod = regex_remove_punctuation.sub('', text)
    wordsOnly = str(text_mod).split()
    for word in wordsOnly:
        if len(word) <= 1:
            wordsOnly.remove(word)
    puncList = [".", "!", "?", ",", ";", ":", "-", "'", "\"",
                "!!", "!!!", "??", "???", "?!?", "!?!", "?!?!", "!?!?"]
    for word in wordsOnly:
        for p in puncList:
            pword = p + word
            x1 = wordsAndEmoticons.count(pword)
            while x1 > 0:
                i = wordsAndEmoticons.index(pword)
                wordsAndEmoticons.remove(pword)
                wordsAndEmoticons.insert(i, word)
                x1 = wordsAndEmoticons.count(pword)

            wordp = word + p
            x2 = wordsAndEmoticons.count(wordp)
            while x2 > 0:
                i = wordsAndEmoticons.index(wordp)
                wordsAndEmoticons.remove(wordp)
                wordsAndEmoticons.insert(i, word)
                x2 = wordsAndEmoticons.count(wordp)
    for word in wordsAndEmoticons:
        if len(word) <= 1:
            wordsAndEmoticons.remove(word)
```

Figure 5.3: removal of punchulation_stopword_singleLetterWord



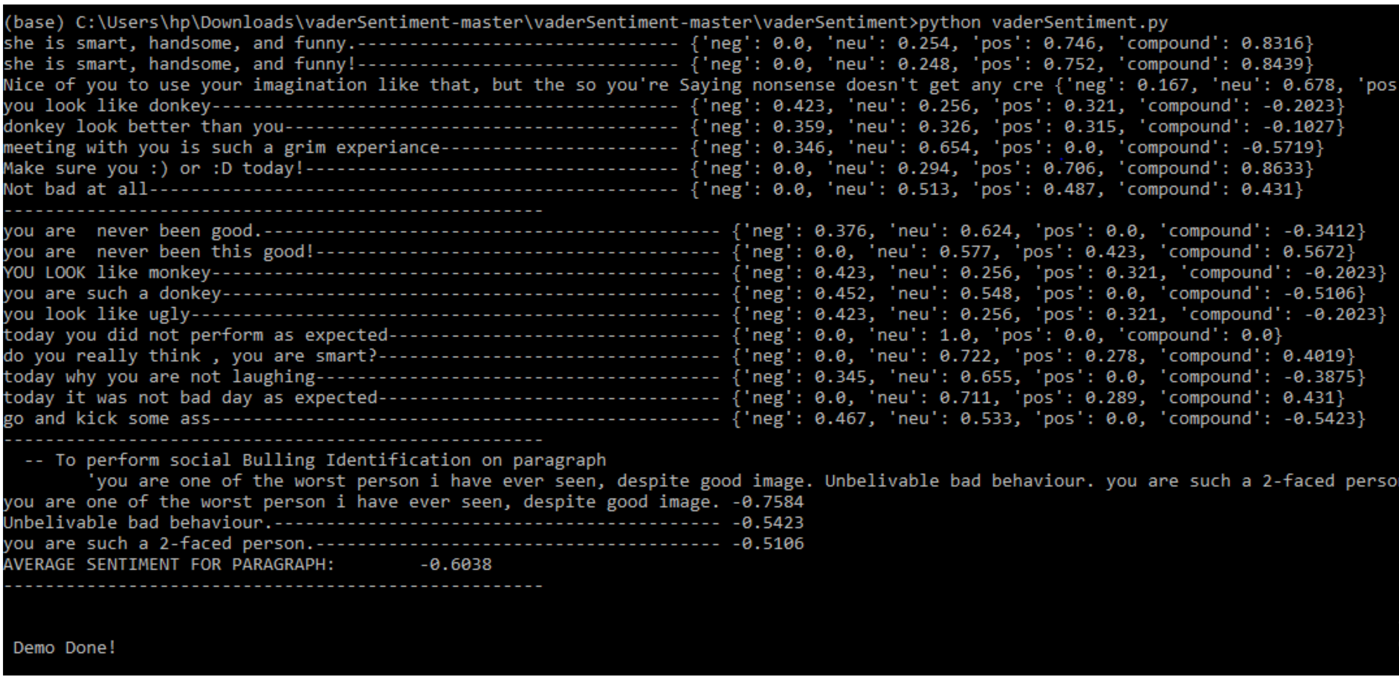
```
sentiments = []
for item in wordsAndEmoticons:
    v = 0
    i = wordsAndEmoticons.index(item)
    if (i < len(wordsAndEmoticons)-1 and str(item).lower() == "kind" and \
       str(wordsAndEmoticons[i+1]).lower() == "of") or str(item).lower() in booster_dict:
        sentiments.append(v)
        continue
    item_lowercase = str(item).lower()
    if  item_lowercase in word_valence_dict:
        v = float(word_valence_dict[item_lowercase])

        c_incr = 0.733
        if str(item).isupper() and isCap_diff:
            if v > 0: v += c_incr
            else: v -= c_incr

        #check if the preceding words increase, decrease, or negate/nullify the valence
        def scalar_inc_dec(word, valence):
            scalar = 0.0
            word_lower = str(word).lower()
            if word_lower in booster_dict:
                scalar = booster_dict[word_lower]
                if valence < 0: scalar *= -1
                if str(word).isupper() and isCap_diff:
                    if valence > 0: scalar += c_incr
                    else:  scalar -= c_incr
            return scalar
        n_scalar = -0.74
        if i > 0 and str(wordsAndEmoticons[i-1]).lower() not in word_valence_dict:
            s1 = scalar_inc_dec(wordsAndEmoticons[i-1], v)
            v = v+s1
            if negated([wordsAndEmoticons[i-1]]): v = v*n_scalar
        if i > 1 and str(wordsAndEmoticons[i-2]).lower() not in word_valence_dict:
            s2 = scalar_inc_dec(wordsAndEmoticons[i-2], v)
            if s2 != 0: s2 = s2*0.95
            v = v+s2
```

Figure 5.4: checking_inTo_dictionary



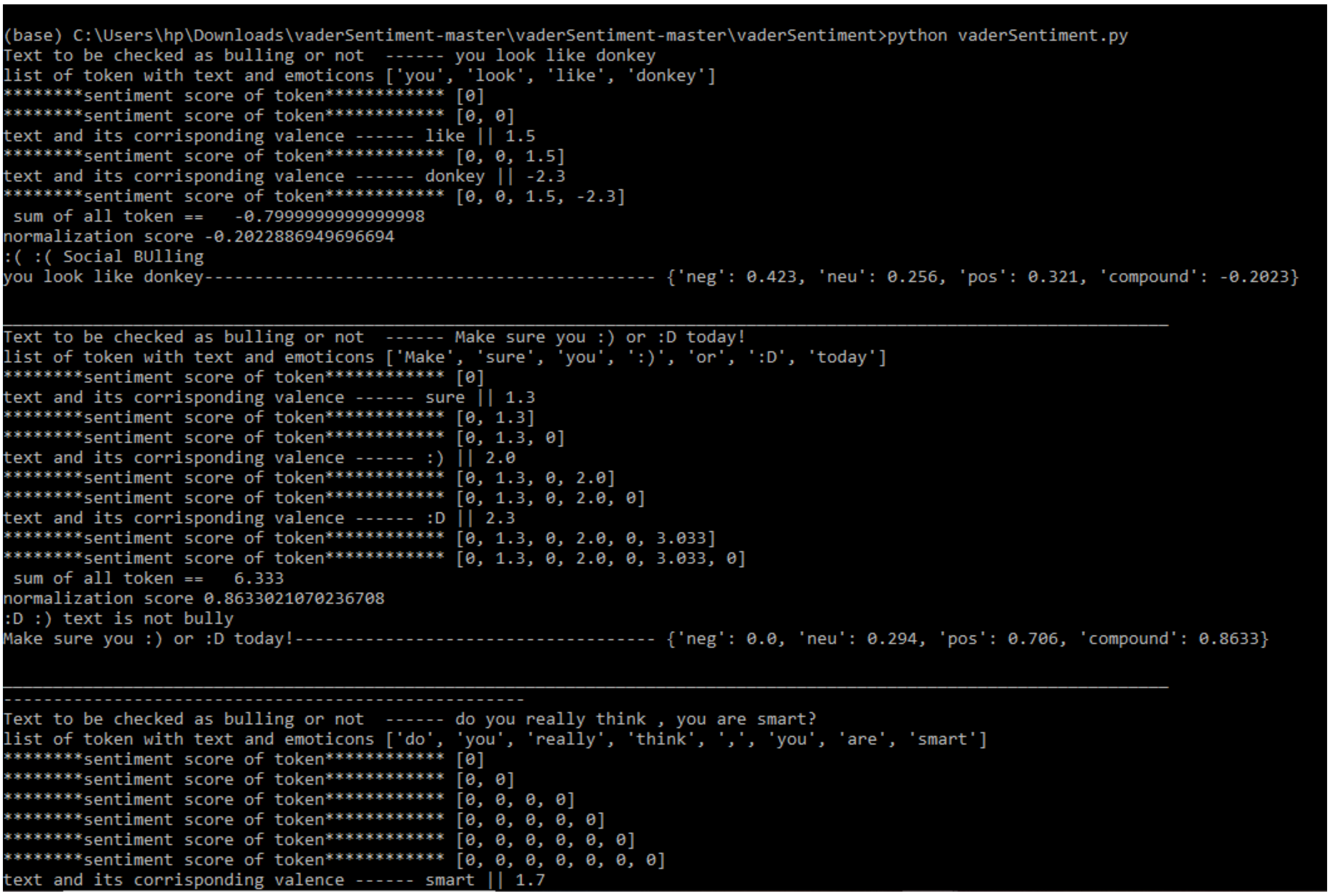
```
    pos_sum = 0.0
    neg_sum = 0.0
    neu_count = 0
    for sentiment_score in sentiments:
        if sentiment_score > 0:
            pos_sum += (float(sentiment_score) +1)
        if sentiment_score < 0:
            neg_sum += (float(sentiment_score) -1)
        if sentiment_score == 0:
            neu_count += 1

    if pos_sum > math.fabs(neg_sum): pos_sum += (ep_amplifier+qm_amplifier)
    elif pos_sum < math.fabs(neg_sum): neg_sum -= (ep_amplifier+qm_amplifier)

    total = pos_sum + math.fabs(neg_sum) + neu_count
    pos = math.fabs(pos_sum / total)
    neg = math.fabs(neg_sum / total)
    neu = math.fabs(neu_count / total)

else:
    compound = 0.0; pos = 0.0; neg = 0.0; neu = 0.0

s = {"neg" : round(neg, 3),
     "neu" : round(neu, 3),
     "pos" : round(pos, 3),
     "compound" : round(compound, 4)}
return s
```

Figure 5.5: Lexicon_score_calculation



```
(base) C:\Users\hp\Downloads\vaderSentiment-master\vaderSentiment-master\vaderSentiment>python vaderSentiment.py
she is smart, handsome, and funny.------------------------------- {'neg': 0.0, 'neu': 0.254, 'pos': 0.746, 'compound': 0.8316}
she is smart, handsome, and funny!------------------------------- {'neg': 0.0, 'neu': 0.248, 'pos': 0.752, 'compound': 0.8439}
Nice of you to use your imagination like that, but the so you're Saying nonsense doesn't get any cre {'neg': 0.167, 'neu': 0.678, 'pos
you look like donkey--------------------------------------------- {'neg': 0.423, 'neu': 0.256, 'pos': 0.321, 'compound': -0.2023}
donkey look better than you-------------------------------------- {'neg': 0.359, 'neu': 0.326, 'pos': 0.315, 'compound': -0.1027}
meeting with you is such a grim experiance----------------------- {'neg': 0.346, 'neu': 0.654, 'pos': 0.0, 'compound': -0.5719}
Make sure you :) or :D today!------------------------------------ {'neg': 0.0, 'neu': 0.294, 'pos': 0.706, 'compound': 0.8633}
Not bad at all--------------------------------------------------- {'neg': 0.0, 'neu': 0.513, 'pos': 0.487, 'compound': 0.431}
----------------------------------------------------
you are  never been good.-------------------------------------------- {'neg': 0.376, 'neu': 0.624, 'pos': 0.0, 'compound': -0.3412}
you are  never been this good!--------------------------------------- {'neg': 0.0, 'neu': 0.577, 'pos': 0.423, 'compound': 0.5672}
YOU LOOK like monkey------------------------------------------------- {'neg': 0.423, 'neu': 0.256, 'pos': 0.321, 'compound': -0.2023}
you are such a donkey------------------------------------------------ {'neg': 0.452, 'neu': 0.548, 'pos': 0.0, 'compound': -0.5106}
you look like ugly--------------------------------------------------- {'neg': 0.423, 'neu': 0.256, 'pos': 0.321, 'compound': -0.2023}
today you did not perform as expected-------------------------------- {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}
do you really think , you are smart?--------------------------------- {'neg': 0.0, 'neu': 0.722, 'pos': 0.278, 'compound': 0.4019}
today why you are not laughing--------------------------------------- {'neg': 0.345, 'neu': 0.655, 'pos': 0.0, 'compound': -0.3875}
today it was not bad day as expected--------------------------------- {'neg': 0.0, 'neu': 0.711, 'pos': 0.289, 'compound': 0.431}
go and kick some ass------------------------------------------------- {'neg': 0.467, 'neu': 0.533, 'pos': 0.0, 'compound': -0.5423}
----------------------------------------------------
  -- To perform social Bulling Identification on paragraph
        'you are one of the worst person i have ever seen, despite good image. Unbelivable bad behaviour. you are such a 2-faced perso
you are one of the worst person i have ever seen, despite good image. -0.7584
Unbelivable bad behaviour.------------------------------------------- -0.5423
you are such a 2-faced person.--------------------------------------- -0.5106
AVERAGE SENTIMENT FOR PARAGRAPH:        -0.6038
----------------------------------------------------


 Demo Done!
```

Figure 5.6: Lexicon based Approach



```
(base) C:\Users\hp\Downloads\vaderSentiment-master\vaderSentiment-master\vaderSentiment>python vaderSentiment.py
Text to be checked as bulling or not  ------ you look like donkey
list of token with text and emoticons ['you', 'look', 'like', 'donkey']
********sentiment score of token************ [0]
********sentiment score of token************ [0, 0]
text and its corrisponding valence ------ like || 1.5
********sentiment score of token************ [0, 0, 1.5]
text and its corrisponding valence ------ donkey || -2.3
********sentiment score of token************ [0, 0, 1.5, -2.3]
 sum of all token ==   -0.7999999999999998
normalization score -0.2022886949696694
:( :( Social BUlling
you look like donkey--------------------------------------------- {'neg': 0.423, 'neu': 0.256, 'pos': 0.321, 'compound': -0.2023}

____________________________________________________________________________________________________________________
Text to be checked as bulling or not  ------ Make sure you :) or :D today!
list of token with text and emoticons ['Make', 'sure', 'you', ':)', 'or', ':D', 'today']
********sentiment score of token************ [0]
text and its corrisponding valence ------ sure || 1.3
********sentiment score of token************ [0, 1.3]
********sentiment score of token************ [0, 1.3, 0]
text and its corrisponding valence ------ :) || 2.0
********sentiment score of token************ [0, 1.3, 0, 2.0]
********sentiment score of token************ [0, 1.3, 0, 2.0, 0]
text and its corrisponding valence ------ :D || 2.3
********sentiment score of token************ [0, 1.3, 0, 2.0, 0, 3.033]
********sentiment score of token************ [0, 1.3, 0, 2.0, 0, 3.033, 0]
 sum of all token ==   6.333
normalization score 0.8633021070236708
:D :) text is not bully
Make sure you :) or :D today!------------------------------------ {'neg': 0.0, 'neu': 0.294, 'pos': 0.706, 'compound': 0.8633}

____________________________________________________________________________________________________________________
----------------------------------------------------
Text to be checked as bulling or not  ------ do you really think , you are smart?
list of token with text and emoticons ['do', 'you', 'really', 'think', ',', 'you', 'are', 'smart']
********sentiment score of token************ [0]
********sentiment score of token************ [0, 0]
********sentiment score of token************ [0, 0, 0, 0]
********sentiment score of token************ [0, 0, 0, 0, 0]
********sentiment score of token************ [0, 0, 0, 0, 0, 0]
********sentiment score of token************ [0, 0, 0, 0, 0, 0, 0]
text and its corrisponding valence ------ smart || 1.7
```

Figure 5.7: Lexicon based Approach

5.0.3 Using Machine Learning Based Approach

In order to identify the bully content from social data another approach is machine learning based approach. As we discussed in lexicon based approach comment or data which is provided on social media is in human language and contain many noise which need to clean before passed to classification model.

while using machine learning based approach we are using tweeter data set generated during trump election campaign.we divide the dataset into two part. 1) Training Dataset 2) Testing Dataset

In Machine learning based approach following approaches are used to classify the data: 1) Supervised learning: 2) Unsupervised learning: 3) Reinforcement learning:

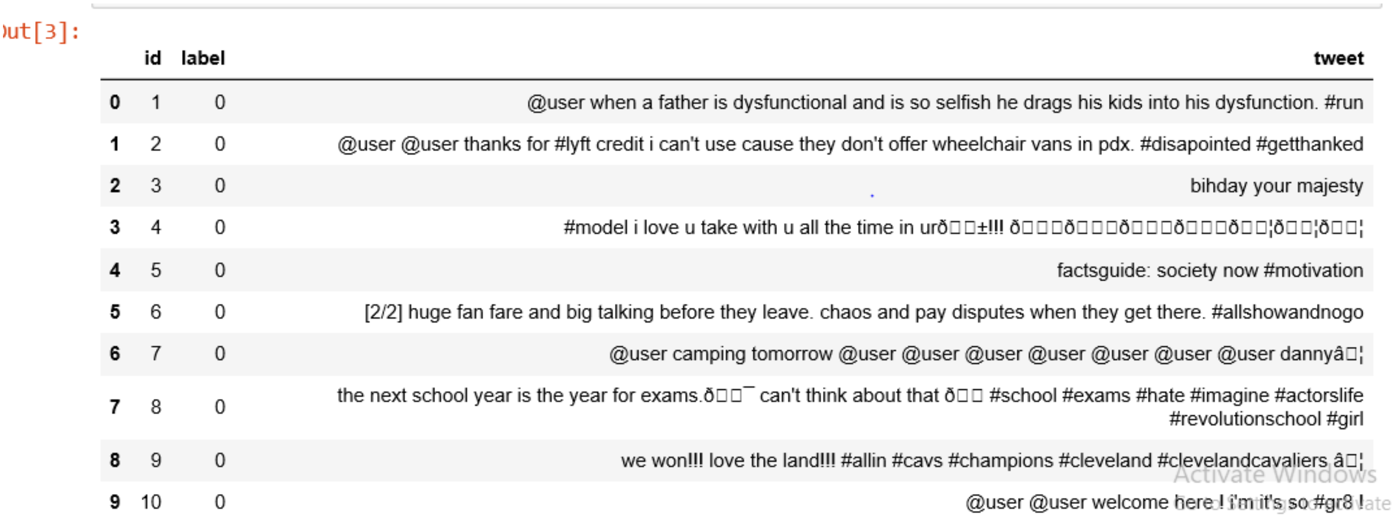
Here to identify social bulling from social media we are going to use supervised machine learning approach to train and test the data.

In supervised learning approach, labeled dataset is passed to train the model and unlabeled dataset is passed to already trained model. following are the steps to be performed in machine learning based approach.

- Data Gathering
- Data Prepossessing
- Data Normalization
- Feature Extraction
- classification
- result

Data Gathering

Here we are using tweeter dataset containing comment from Trump election campaign. This dataset is divided into two phase training and testing .Training dataset contain labeled dataset containing two type of data bulling and non-bulling statement. Here data with label "0" is consider as non-bulling statement5.8 and with label "1" is consider as butting statement 5.9 to train the classification model.5.10

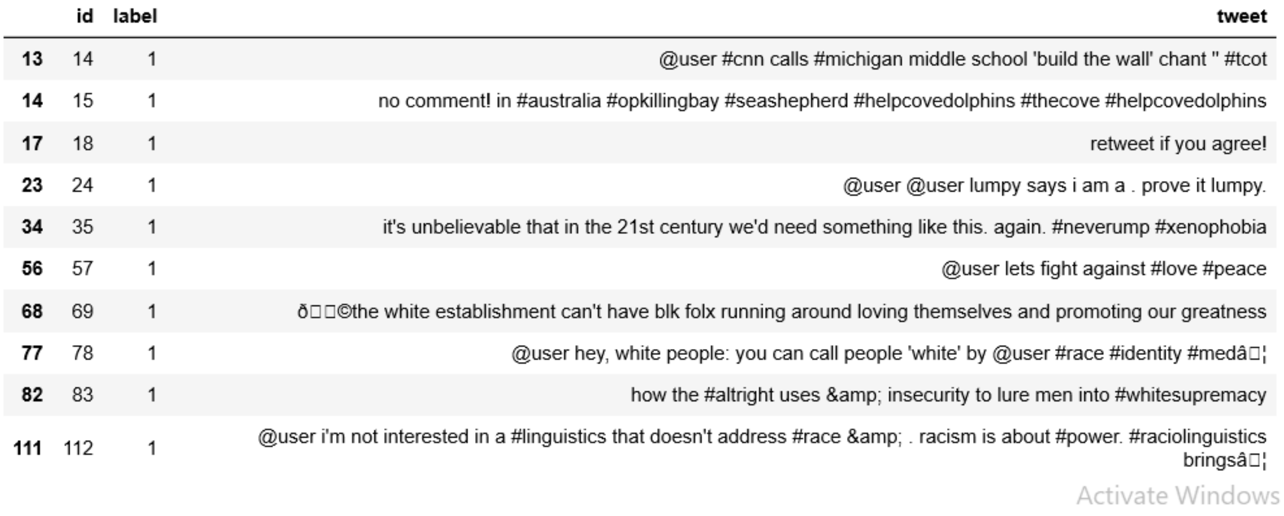


ut[3]:

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦
4	5	0	factsguide: society now #motivation
5	6	0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo
6	7	0	@user camping tomorrow @user @user @user @user @user @user @user dannyâ□¦
7	8	0	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl
8	9	0	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers â□¦
9	10	0	@user @user welcome here ! i'm it's so #gr8 !

Activate Windows
Go to Settings to activate Windows

Figure 5.8: Database with label 0



	id	label	tweet
13	14	1	@user #cnn calls #michigan middle school 'build the wall' chant " #tcot
14	15	1	no comment! in #australia #opkillingbay #seashepherd #helpcovedolphins #thecove #helpcovedolphins
17	18	1	retweet if you agree!
23	24	1	@user @user lumpy says i am a . prove it lumpy.
34	35	1	it's unbelievable that in the 21st century we'd need something like this. again. #neverump #xenophobia
56	57	1	@user lets fight against #love #peace
68	69	1	ð□□©the white establishment can't have blk folx running around loving themselves and promoting our greatness
77	78	1	@user hey, white people: you can call people 'white' by @user #race #identity #medâ□¦
82	83	1	how the #altright uses &amp; insecurity to lure men into #whitesupremacy
111	112	1	@user i'm not interested in a #linguistics that doesn't address #race &amp; . racism is about #power. #raciolinguistics bringsâ□¦

Activate Windows

Figure 5.9: Database with label 1

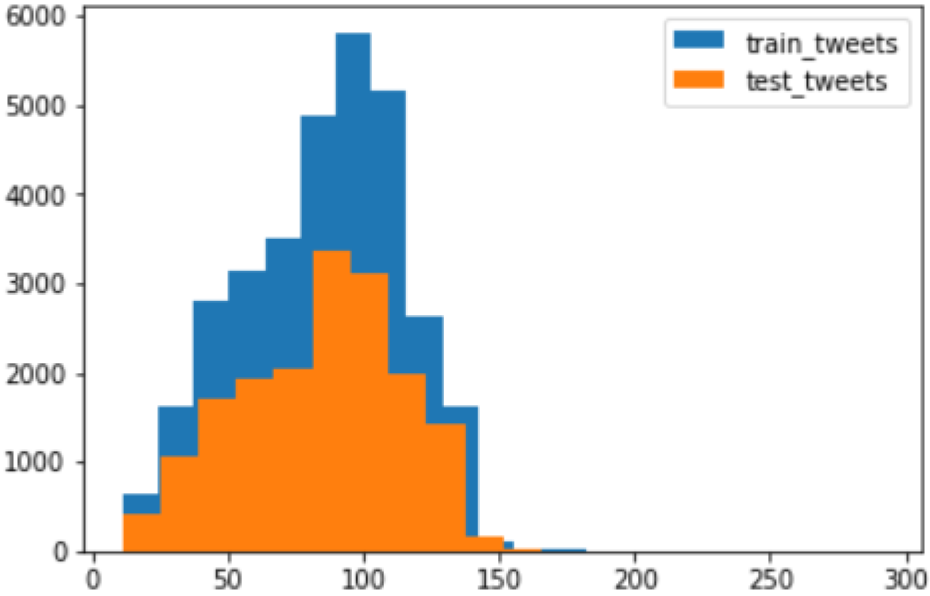


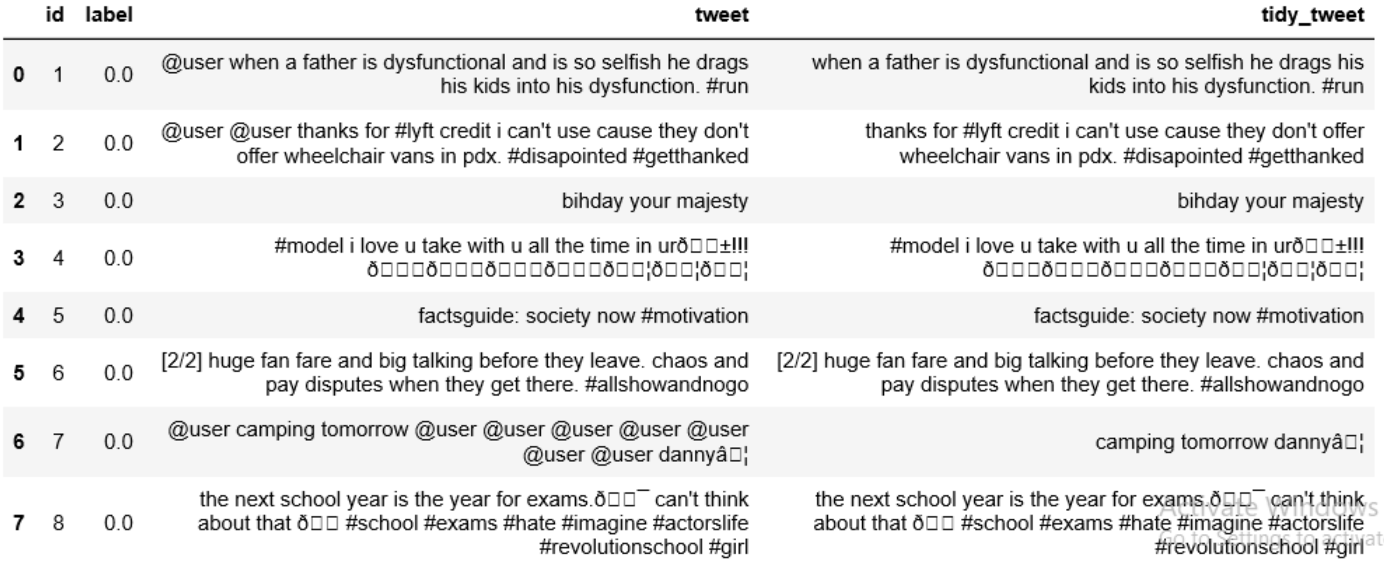
Figure 5.10: Bulling vs Non-Bulling data graphical presentation

2 Data Prepossessing: Data gathered from tweeter containing many kind of noise like @ notation, tag , punctuation mark, stop word etc. To clean this data following are the prepossessing step to be covered in data prepossessing step.

- Removal of symbol
- Removal of Punctuation Mark
- Removal of stop word

Removal of @ symbol

In the tweeter data, use usually used to mention other user by mention their name with @ notation. This kind of data contain user name which use not much use full to identify that the given statement is bully statement or not. In this step we identify the word start with @ symbol and remove them from data5.11



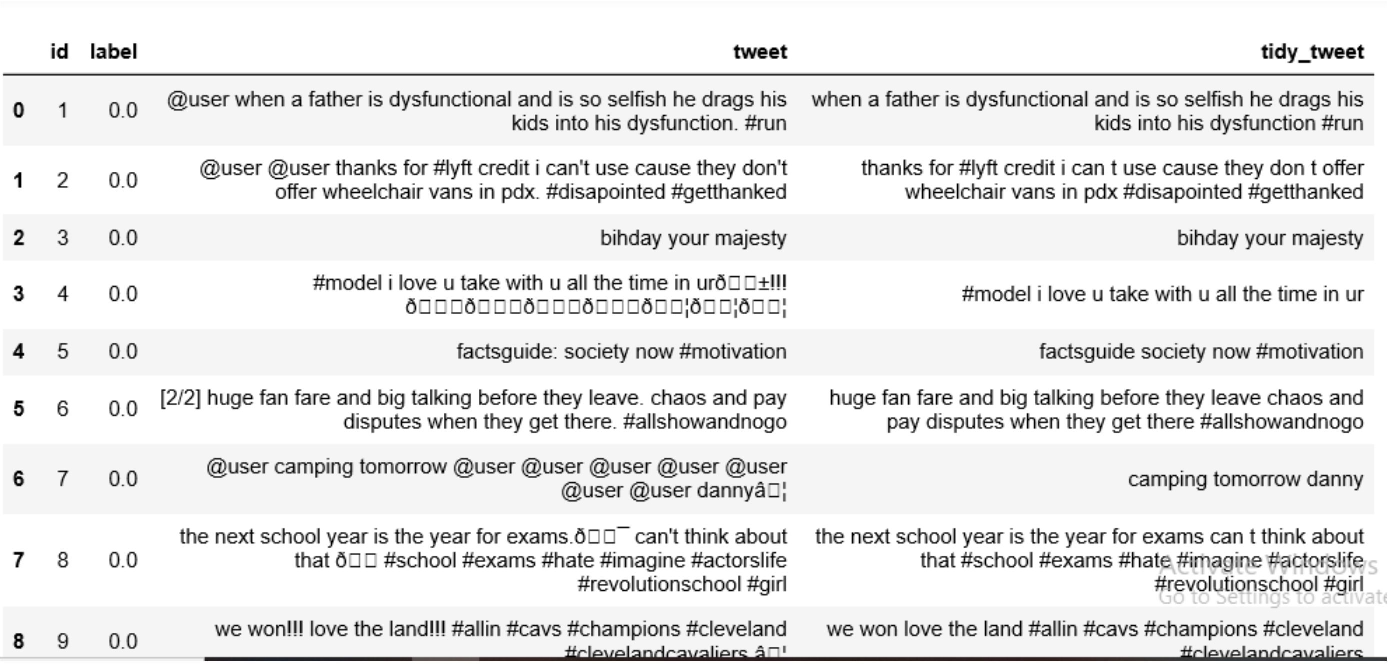
	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦
4	5	0.0	factsguide: society now #motivation	factsguide: society now #motivation
5	6	0.0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo
6	7	0.0	@user camping tomorrow @user @user @user @user @user @user @user dannyâ□¦	camping tomorrow dannyâ□¦
7	8	0.0	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl

Activate Windows
Go to Settings to activate

Figure 5.11: Removal of @ symbol

Removal of punctuation In this section we remove all the punctuation mark, number and special character from the data as this data is not needed for social bulling identification. 5.12

Removal of Stop Words Data generated from social media is in human language witch contain many stop words which are not need for further classification. In this step

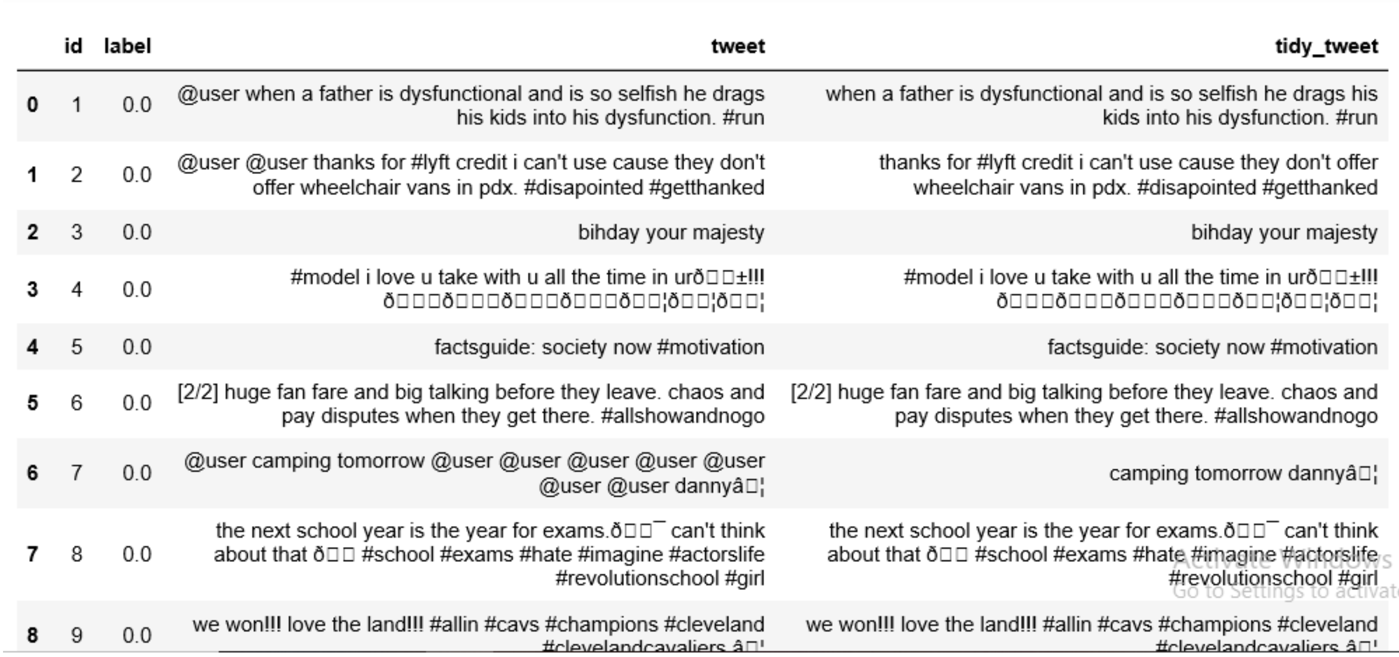


	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when a father is dysfunctional and is so selfish he drags his kids into his dysfunction #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks for #lyft credit i can t use cause they don t offer wheelchair vans in pdx #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦	#model i love u take with u all the time in ur
4	5	0.0	factsguide: society now #motivation	factsguide society now #motivation
5	6	0.0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo	huge fan fare and big talking before they leave chaos and pay disputes when they get there #allshowandnogo
6	7	0.0	@user camping tomorrow @user @user @user @user @user @user @user dannyâ□¦	camping tomorrow danny
7	8	0.0	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl	the next school year is the year for exams can t think about that #school #exams #hate #imagine #actorslife #revolutionschool #girl
8	9	0.0	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers â□¦	we won love the land #allin #cavs #champions #cleveland #clevelandcavaliers

Activate Windows
Go to Settings to activate

Figure 5.12: Removal of punctuation mark, number and special character

we are removing stop word like I, am, the, is all other one. 5.13



	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦	#model i love u take with u all the time in urð□□±!!! ð□□□ð□□□ð□□□ð□□□ð□□¦ð□□¦ð□□¦
4	5	0.0	factsguide: society now #motivation	factsguide: society now #motivation
5	6	0.0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo
6	7	0.0	@user camping tomorrow @user @user @user @user @user @user @user dannyâ□¦	camping tomorrow dannyâ□¦
7	8	0.0	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl	the next school year is the year for exams.ð□□¯ can't think about that ð□□ #school #exams #hate #imagine #actorslife #revolutionschool #girl
8	9	0.0	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers â□¦	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers â□¦

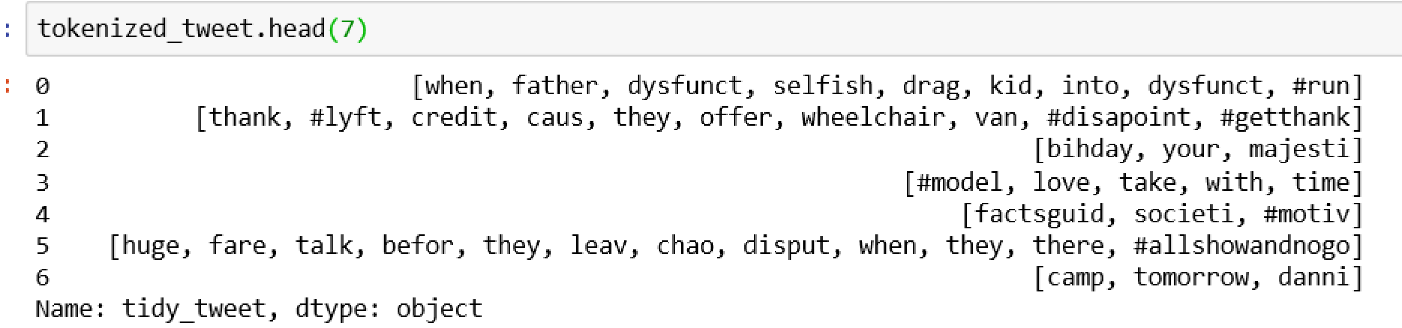
Activate Windows
Go to Settings to activate

Figure 5.13: Removal of Stop Word

3 Normalization

Before normalizing the text, we have to devide the sentance in to single token. First step before normalization is text tokenization.

Tokenization In this step we use nltk to devide the hole statement in to single token, this token are used for further classification.5.14



```
: tokenized_tweet.head(7)

: 0                         [when, father, dysfunct, selfish, drag, kid, into, dysfunct, #run]
  1          [thank, #lyft, credit, caus, they, offer, wheelchair, van, #disapoint, #getthank]
  2                                                                 [bihday, your, majesti]
  3                                                          [#model, love, take, with, time]
  4                                                             [factsguid, societi, #motiv]
  5    [huge, fare, talk, befor, they, leav, chao, disput, when, they, there, #allshowandnogo]
  6                                                                  [camp, tomorrow, danni]
  Name: tidy_tweet, dtype: object
```

Figure 5.14: After tokenization

what to do with # tag ?

In most of the text prepossessing technique # also removed from the data to get the noise free data during classification. To get to the correct decision we can display the data connected with # tag.5.15 5.16 In tweeter data in most of the cases # is used to define the latest trend. By considering this detail we are going to use this data to identify the social bulling content.

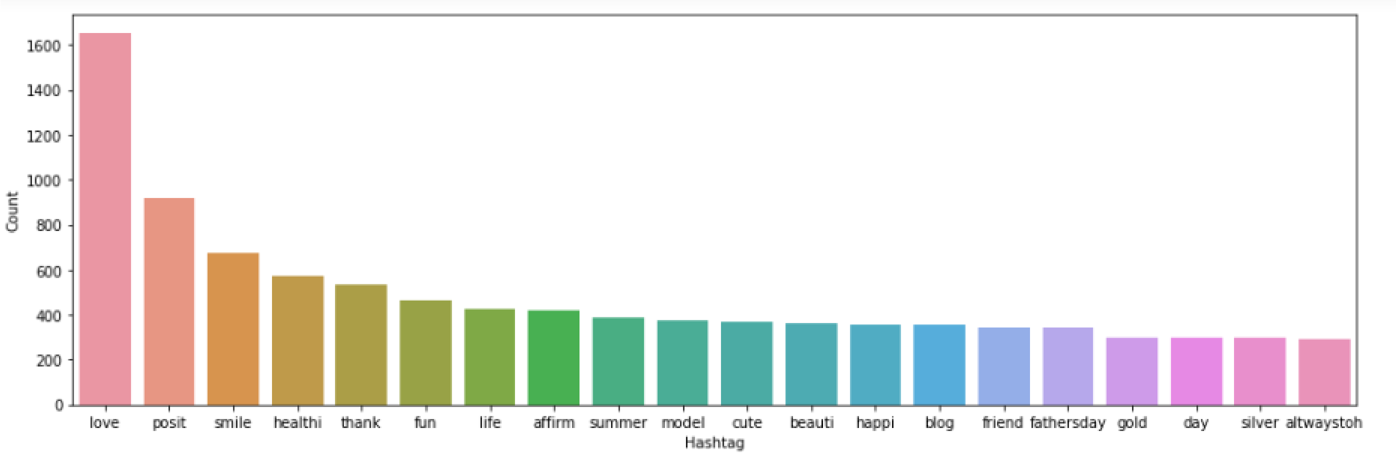


Figure 5.15: Hashtag with label 0

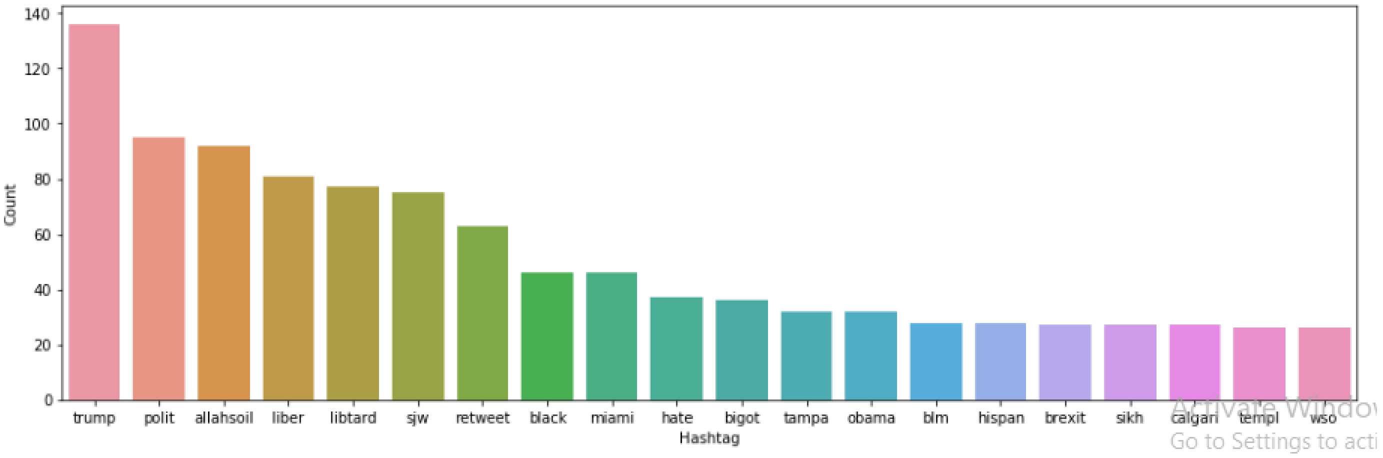


Figure 5.16: Hashtag with label 1

Feature Extraction

Feature Extraction is one of the most important step to identify most important feature from prepossess text to get more accurate result during classification.[14] Here we list some of the feature extraction technique used to identify better feature from the text data.[14]

- Bag of Word
- TF_IDF
- Word2Vec

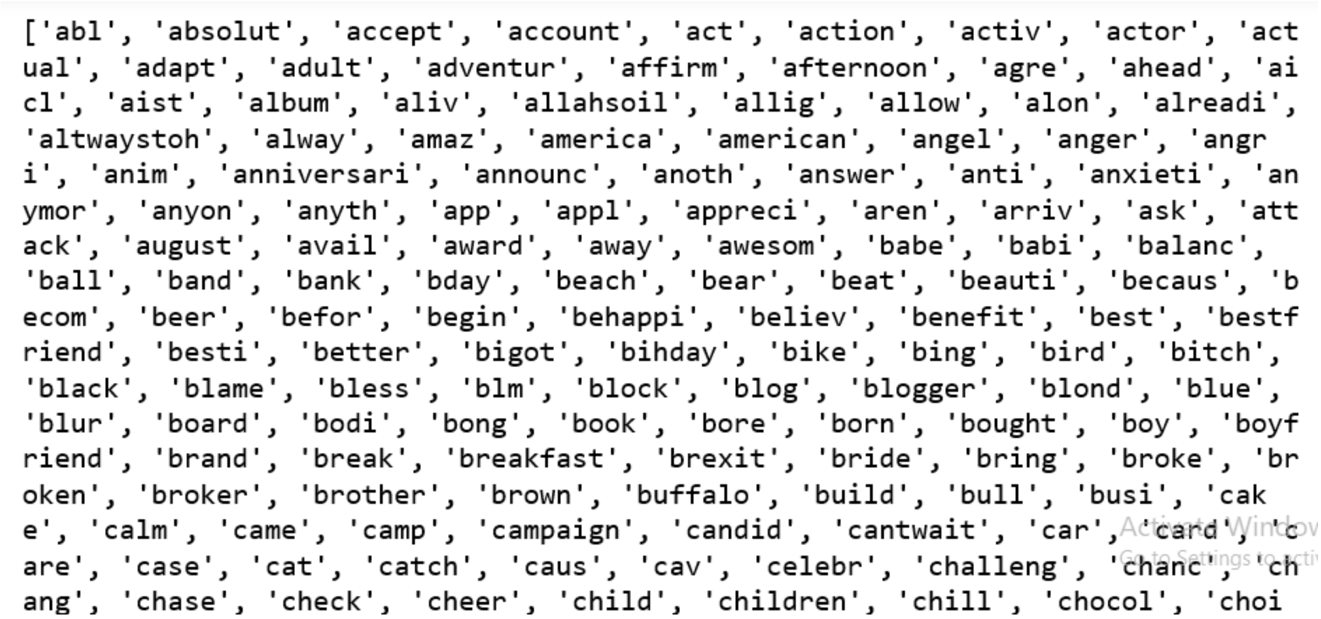
I) Bag of Word: Bag of word is the feature extraction technique which form the set of the feature from all the word of instance. Unlike TF_IDF this Bag of Word technique don't care about the number of time word occur in the database, it only concern about either the word is present in list or not. This is most simple and common method for feature extraction. we will use this technique to get the list of feature and use with different classifier.**??**

II) TF_IDF

As we discussed in the earlier part of TF_IDF is the another most used feature extraction technique which is used to get the list of most use full feature based on the frequency of the terms in the dataset.

III) Word2Vec

Word2Vec is known as word to vectore feature extraction technique. It is used to create word embedding. Its input is the text coupus and it give the output as set of vector. We use this to give input as different classification model.



```
['abl', 'absolut', 'accept', 'account', 'act', 'action', 'activ', 'actor', 'act
ual', 'adapt', 'adult', 'adventur', 'affirm', 'afternoon', 'agre', 'ahead', 'ai
cl', 'aist', 'album', 'aliv', 'allahsoil', 'allig', 'allow', 'alon', 'alreadi',
'altwaystoh', 'alway', 'amaz', 'america', 'american', 'angel', 'anger', 'angr
i', 'anim', 'anniversari', 'announc', 'anoth', 'answer', 'anti', 'anxieti', 'an
ymor', 'anyon', 'anyth', 'app', 'appl', 'appreci', 'aren', 'arriv', 'ask', 'att
ack', 'august', 'avail', 'award', 'away', 'awesom', 'babe', 'babi', 'balanc',
'ball', 'band', 'bank', 'bday', 'beach', 'bear', 'beat', 'beauti', 'becaus', 'b
ecom', 'beer', 'befor', 'begin', 'behappi', 'believ', 'benefit', 'best', 'bestf
riend', 'besti', 'better', 'bigot', 'bihday', 'bike', 'bing', 'bird', 'bitch',
'black', 'blame', 'bless', 'blm', 'block', 'blog', 'blogger', 'blond', 'blue',
'blur', 'board', 'bodi', 'bong', 'book', 'bore', 'born', 'bought', 'boy', 'boyf
riend', 'brand', 'break', 'breakfast', 'brexit', 'bride', 'bring', 'broke', 'br
oken', 'broker', 'brother', 'brown', 'buffalo', 'build', 'bull', 'busi', 'cak
e', 'calm', 'came', 'camp', 'campaign', 'candid', 'cantwait', 'car', 'card', 'c
are', 'case', 'cat', 'catch', 'caus', 'cav', 'celebr', 'challeng', 'chanc', 'ch
ang', 'chase', 'check', 'cheer', 'child', 'children', 'chill', 'chocol', 'choi
```

Activate Windo
Go to Settings to acti

Figure 5.17: Using bagofword Technique

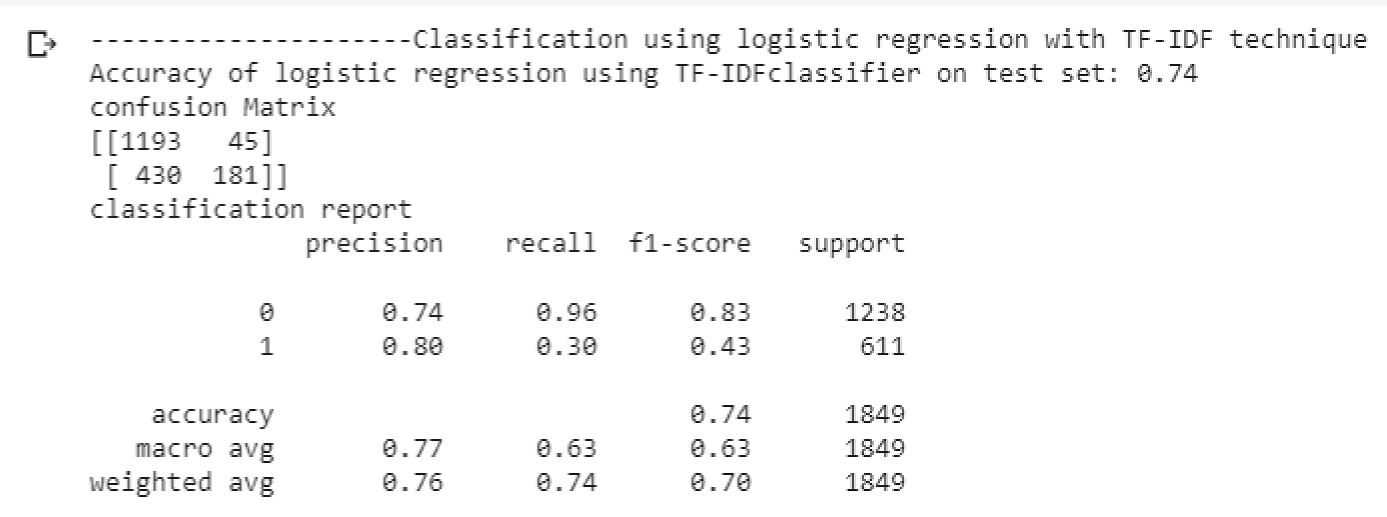
Classification: After feature extraction, featured data is passed to classification model to classify the text as bully or non bully content. Here we use three different classification technique to compare result of them and get the more accurate result.

- Logistic Regression
- Naive Bayes
- Support Vector Machine

In this implementation we use use different feature extraction technique and pass this all to this three classification model and try to get the more accurate one.

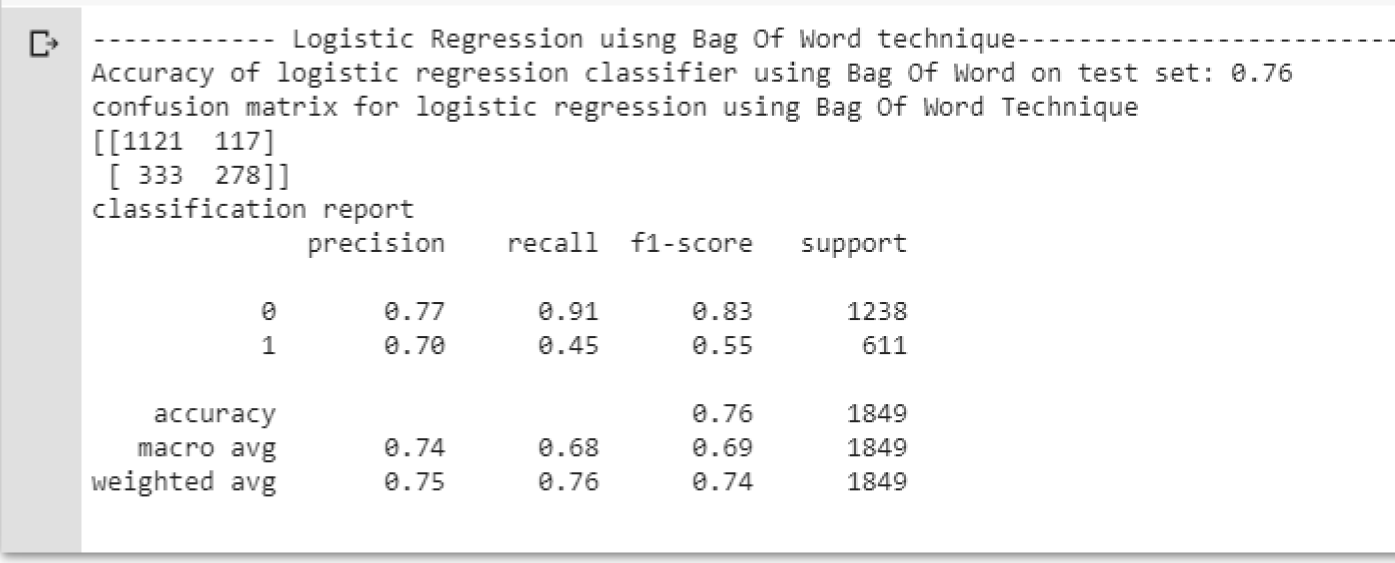
I) Logistic Regression:

Logistic Regression is the statistic model which is use logical function to classify the data. It is the machine learning model used to predict the probability of data according to different classes. Lodistic regression is binary variable it is either 0 or 1. In this model 0 is the non-bulling comment and 1 is the bully comment. In ths logistic regration the most important thing is that dependant variable must be binary. And another thing is that data set should be much larger.



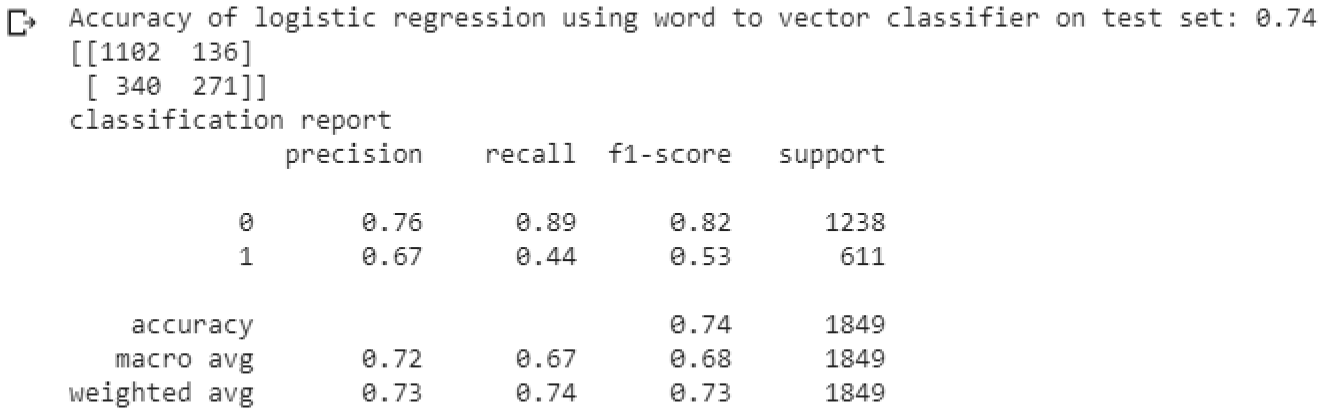
```
--------------------Classification using logistic regression with TF-IDF technique
Accuracy of logistic regression using TF-IDFclassifier on test set: 0.74
confusion Matrix
[[1193   45]
 [ 430  181]]
classification report
              precision    recall  f1-score   support

           0       0.74      0.96      0.83      1238
           1       0.80      0.30      0.43       611

    accuracy                           0.74      1849
   macro avg       0.77      0.63      0.63      1849
weighted avg       0.76      0.74      0.70      1849
```

Figure 5.18: Logistic Regression with TF_IDF feature set



```
------------ Logistic Regression uisng Bag Of Word technique-------------------------
Accuracy of logistic regression classifier using Bag Of Word on test set: 0.76
confusion matrix for logistic regression using Bag Of Word Technique
[[1121  117]
 [ 333  278]]
classification report
              precision    recall  f1-score   support

           0       0.77      0.91      0.83      1238
           1       0.70      0.45      0.55       611

    accuracy                           0.76      1849
   macro avg       0.74      0.68      0.69      1849
weighted avg       0.75      0.76      0.74      1849
```

Figure 5.19: Logistic Regression with Bag of Word feature set

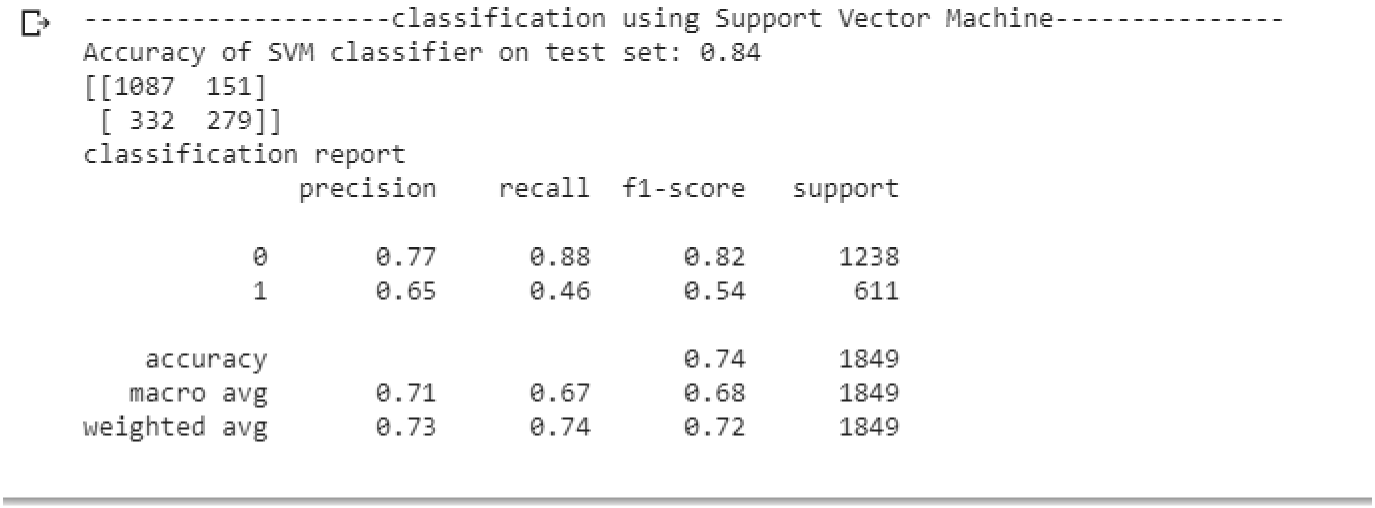


```
Accuracy of logistic regression using word to vector classifier on test set: 0.74
[[1102  136]
 [ 340  271]]
classification report
              precision    recall  f1-score   support

           0       0.76      0.89      0.82      1238
           1       0.67      0.44      0.53       611

    accuracy                           0.74      1849
   macro avg       0.72      0.67      0.68      1849
weighted avg       0.73      0.74      0.73      1849
```

Figure 5.20: Logistic Regression with W2V feature set

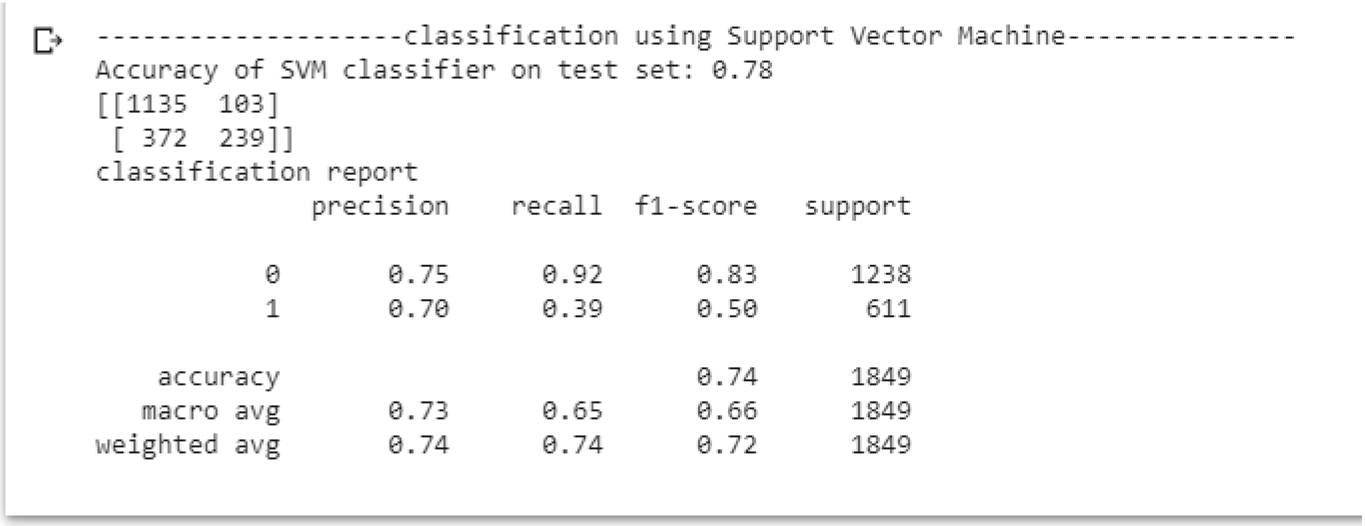
Support Vector Machine

Support Vector Machine is the one of the most used superwised machine learning classifier. It classify the data by finding the hyperplane betwwn N-dimentinal space and divide the data into appropriate classs. It find the hyper plane based on the number of frature and data point. Data point which belog to either side of hyper plane is consider as part of that perticular class after classification.5.21 5.22 5.23



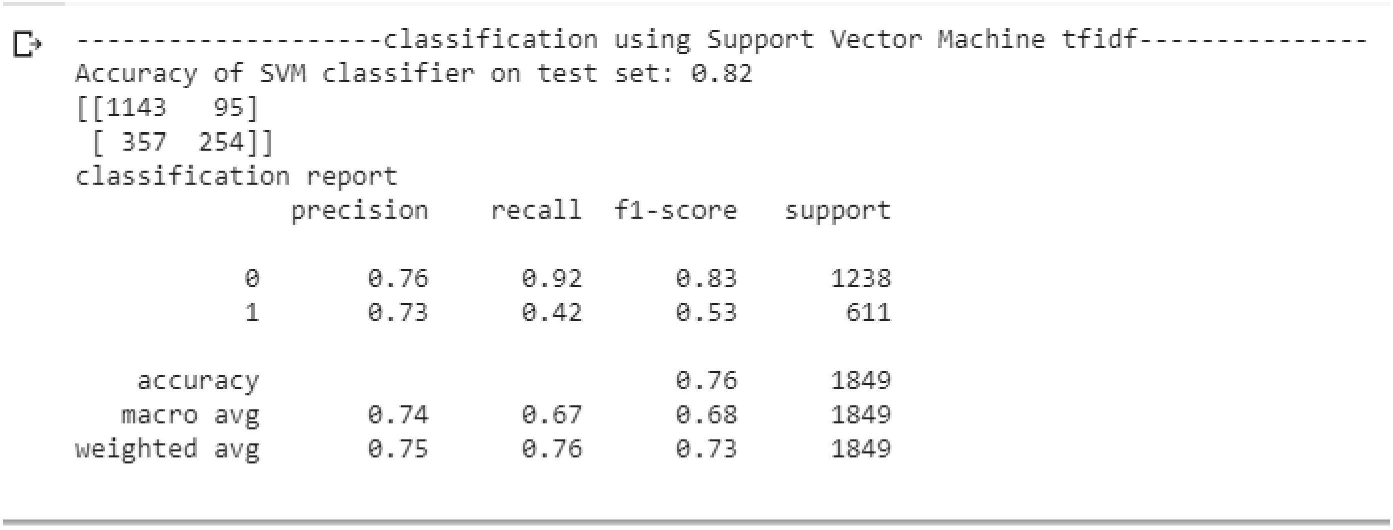
```
--------------------classification using Support Vector Machine---------------
Accuracy of SVM classifier on test set: 0.84
[[1087  151]
 [ 332  279]]
classification report
              precision    recall  f1-score   support

           0       0.77      0.88      0.82      1238
           1       0.65      0.46      0.54       611

    accuracy                           0.74      1849
   macro avg       0.71      0.67      0.68      1849
weighted avg       0.73      0.74      0.72      1849
```

Figure 5.21: SVM with BOW feature set



```
--------------------classification using Support Vector Machine---------------
Accuracy of SVM classifier on test set: 0.78
[[1135  103]
 [ 372  239]]
classification report
              precision    recall  f1-score   support

           0       0.75      0.92      0.83      1238
           1       0.70      0.39      0.50       611

    accuracy                           0.74      1849
   macro avg       0.73      0.65      0.66      1849
weighted avg       0.74      0.74      0.72      1849
```

Figure 5.22: SVM with W2V feature set



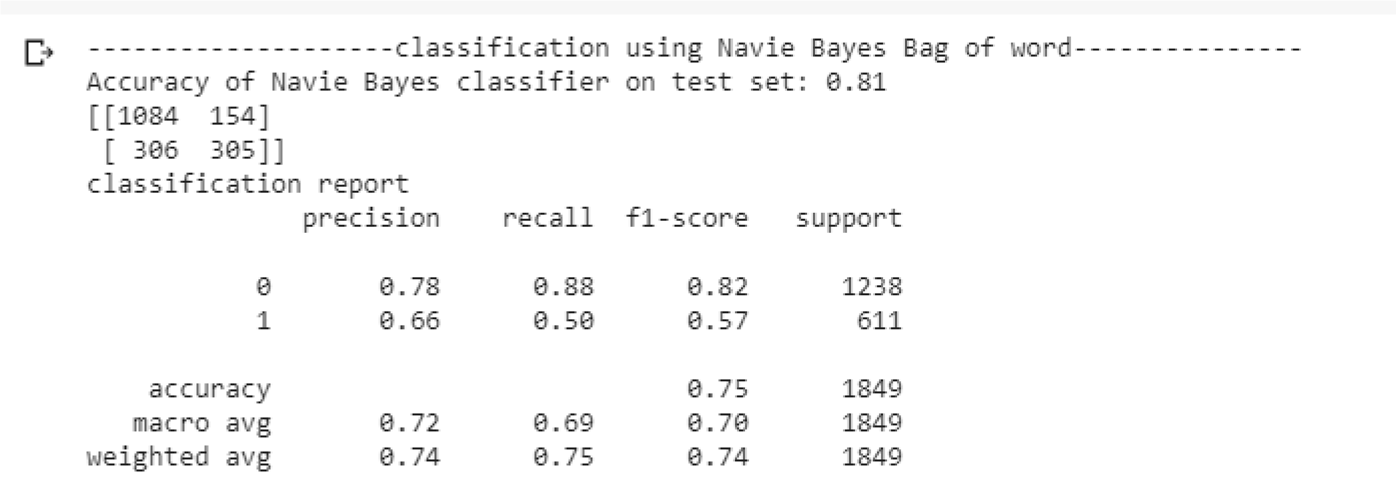
```
--------------------classification using Support Vector Machine tfidf---------------
Accuracy of SVM classifier on test set: 0.82
[[1143   95]
 [ 357  254]]
classification report
              precision    recall  f1-score   support

           0       0.76      0.92      0.83      1238
           1       0.73      0.42      0.53       611

    accuracy                           0.76      1849
   macro avg       0.74      0.67      0.68      1849
weighted avg       0.75      0.76      0.73      1849
```

Figure 5.23: SVM with TF_IDF feature set

Naive Bayes :

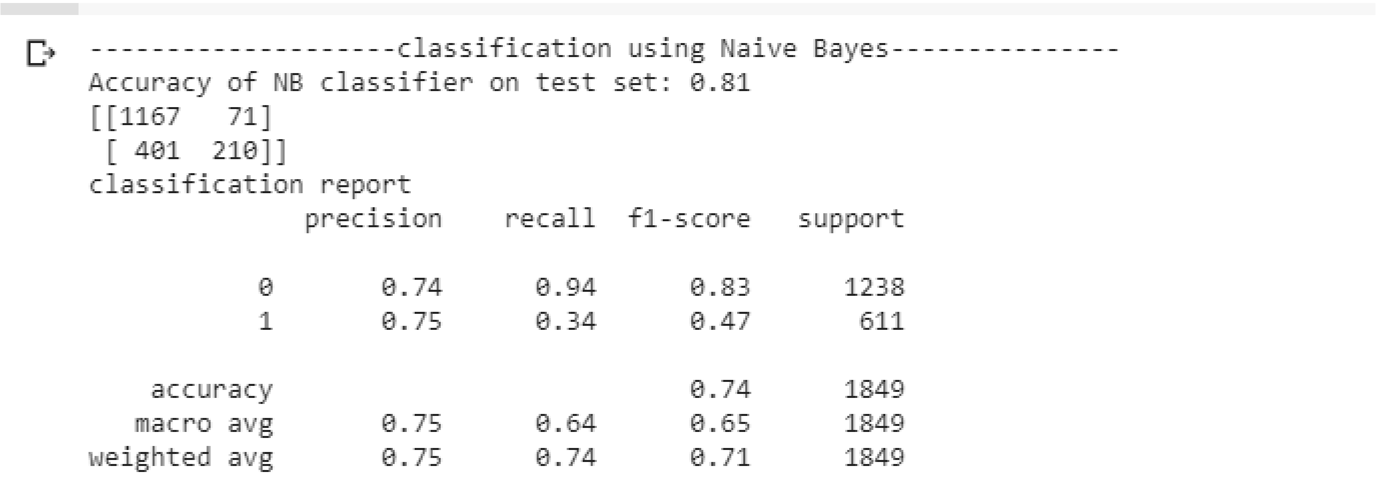
Naive Bayes is one of the most used and simple classifier from supervised Machine learning classifier. Naive Bayes is the probabilistic classifier, which apply Bayes theorem to classify the data into appropriate class. Naive Bayes is work based on Bayes theorem. ?? In our case we can consider A as the bulling content across the B total occurring . A is consider as hypothesis and B as evidence.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$



```
--------------------classification using Navie Bayes Bag of word---------------
Accuracy of Navie Bayes classifier on test set: 0.81
[[1084  154]
 [ 306  305]]
classification report
              precision    recall  f1-score   support

           0       0.78      0.88      0.82      1238
           1       0.66      0.50      0.57       611

    accuracy                           0.75      1849
   macro avg       0.72      0.69      0.70      1849
weighted avg       0.74      0.75      0.74      1849
```

Figure 5.24: NB with BOW feature set



```
--------------------classification using Naive Bayes---------------
Accuracy of NB classifier on test set: 0.81
[[1167   71]
 [ 401  210]]
classification report
              precision    recall  f1-score   support

           0       0.74      0.94      0.83      1238
           1       0.75      0.34      0.47       611

    accuracy                           0.74      1849
   macro avg       0.75      0.64      0.65      1849
weighted avg       0.75      0.74      0.71      1849
```

Figure 5.25: NB with TF_IDF feature set

Chapter 6

Result

Classifier	**Feature Set**	**Accuracy**
Logistic Regression	Bag of Word	76 %
Logistic Regression	TF_ IDF	74%
Logistic Regression	Word2Vec	74%
Support Vector Machine	Bag of Word	84%
Support Vector Machine	TF _ IDF	82 %
Support Vector Machine	Word2Vec	78 %
Naive Bayes	Bag of Word	81 %
NaiveBayes	TF _ IDF	81 %

Table 6.1: Result Analysis

Chapter 7

Conclusion

Social Media is most Wildly used platform by people all over the world. Sentiment Analysis provide way to identify social bulling over Social media using different technique. In this project we use Lexicon Based approach to identify social Bulling over social Media and going to use Machine Learning Based approach in future. After applying machine learning technique and NLP , we can conclude that while using tweeter using # tag data is suitable to get more accurate result. Based on the result 6.1 we get using different feature extraction technique with 3 different classifier we conclude that Bag of Word Technique give more accurate result compare to other with all 3 classifier. As well as Support Vector Machine give the highest accuracy compare to other 2 classifier using all 3 feature set. At the end we can identify that combination of Bag of Word technique and Support Vector Machine give the highest accuracy to identify bully content using sentiment analysis approach over the social media data.

Bibliography

[1] M. H. Abd El-Jawad, R. Hodhod, and Y. M. K. Omar. Sentiment analysis of social media networks using machine learning. In *2018 14th International Computer Engineering Conference (ICENCO)*, pages 174–176, Dec 2018.

[2] Ravinder Ahuja, Aakarsha Chug, Shruti Kohli, Shaurya Gupta, and Pratyush Ahuja. The impact of features extraction on the sentiment analysis. *Procedia Computer Science*, 152:341–348, 2019.

[3] S. Akter and M. T. Aziz. Sentiment analysis on facebook group using lexicon based approach. In *2016 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, pages 1–4, Sep. 2016.

[4] Fabio Del Vigna12, Andrea Cimino23, Felice Dell'Orletta, Marinella Petrocchi, and Maurizio Tesconi. Hate me, hate me not: Hate speech detection on facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17)*, pages 86–95, 2017.

[5] Cicero Dos Santos and Maira Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78, 2014.

[6] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009.

[7] Z. Jianqiang and G. Xiaolin. Comparison research on text pre-processing methods on twitter sentiment analysis. *IEEE Access*, 5:2870–2879, 2017.

[8] Vishal Kharde and Sheetal Sonawane. Sentiment analysis of twitter data: A survey of techniques. *International Journal of Computer Applications*, 139:5–15, 04 2016.

[9] R. Martins, M. Gomes, J. J. Almeida, P. Novais, and P. Henriques. Hate speech classification in social media using emotional analysis. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pages 61–66, Oct 2018.

[10] M. Rathi, A. Malik, D. Varshney, R. Sharma, and S. Mendiratta. Sentiment analysis of tweets using machine learning approach. In *2018 Eleventh International Conference on Contemporary Computing (IC3)*, pages 1–3, Aug 2018.

[11] Dev Shah, Haruna Isah, and Farhana Zulkernine. Predicting the effects of news sentiments on the stock market. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 4705–4708. IEEE, 2018.

[12] Harsh Thakkar and Dhiren Patel. Approaches for sentiment analysis on twitter: A state-of-art study. *arXiv preprint arXiv:1512.01043*, 2015.

[13] Hao Wang and Jorge A Castanon. Sentiment expression via emoticons on social media. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 2404–2408. IEEE, 2015.

[14] Resham N Waykole and A Thakare. A review of feature extraction methods for text classification. *IJAERD*, 5(04), 2018.