DETECTION OF CYBERBULLYING IN SOCIAL MEDIA

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

SAMANA SRIRAM DATTA MANIKANTA (37110671)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119

MARCH - 2021

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **SAMANA SRIRAM DATTA MANIKANTA (37110671)** who carried out the project entitled **"DETECTION OF CYBERBULLYING IN SOCIAL MEDIA"** undermy supervision from November 2020 to March 2021.

Internal Guide

Dr. G.MEERA GANDHI M.E, Ph.D

Head of the Department

Dr. S. VIGNESHWARI M.E., Ph.D.

Submitted for Viva-Voce Examination held on:

Internal Examiner **External Examiner**

DECLARATION

I **SAMANA SRIRAM DATTA MANIKANTA (37110671) ,** hereby declare that the Project Report entitled **"DETECTION OF CYBERBULLYING IN SOCIAL MEDIA"** done by me under the guidance of **Dr. G.MEERA GANDHI M.E, Ph.D** is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks **to Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA M.E., Ph.D.,** Dean, School Of Computing, **Dr. S. VIGNESHWARI M.E., Ph.D.,** and **Dr. L. LAKSHMANAN M.E.,Ph.D.,** Heads Of the Department, Department of Computer Science and Engineering for providing us the necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. G.MEERA GANDHI M.E, Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express our thanks to all Teaching and Non-teaching staff of the **Department of COMPUTER SCIENCE & ENGINEERING** who were helpful in many ways for the completion of the project.

ABSTRACT

In this work, there is an argue for a focus on the latter problem for practical reasons. This project show that it is a much more challenging task, as the analysis of the language in the typical datasets shows that hate speech lacks unique, discriminative features and therefore is found in the ‘long tail’ in a dataset that is difficult to discover. Later in this project there is an propose of Deep Neural Network structures serving as feature extractors that are particularly effective for capturing the semantics of hate speech. These methods are evaluated on the largest collection of hate speech datasets based on Twitter, and are shown to be able to outperform state of the art by up to 6 percentage points in macro-average F1, or 9 percentage points in the more challenging case of identifying hateful content.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The exponential growth of social media such as Twitter and community forums has revolutionised communication and content publishing but is also increasingly exploited for the propagation of hate speech and the organisation of hate-based activities [1, 3]. The anonymity and mobility afforded by such media has made the breeding and spread of hate speech eventually leading to hate crime effortless in a virtual land scape beyond the realms of traditional law enforcement. The term ‘hate speech’ was formally defined as ‘any communication that disparages a person or a group on the basis of some characteristics (to be referred to as types of hate or hate classes) such as race, colour, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics’. In the UK, there has been significant increase of hate speech towards the migrant and Muslim communities following recent events including leaving the EU, the Manchester and the London attacks. In the EU, surveys and reports focusing on young people in the EEA (European Economic Area) region show rising hate speech. And related crimes based on religious beliefs, ethnicity, sexual orientation or gender, as 80% of respondents have encountered hate speech online and 40% felt attacked or threatened. Statistics also show that in the US, hate speech and crime is on the rise since the Trump election. The urgency of this matter has been increasingly recognised, as a range of international initiatives have been launched towards the qualification of the problems and the development of countermeasures.

Cyberbullying or **cyberharassment** is a formof bullying or harassment using electronic means. Cyberbullying and cyberharassment are also known as **online bullying**. It has become increasingly common, especially among teenagers, as the digital sphere has expanded and technology has advanced. Cyberbullying is when someone, typically a teenager, bullies or harasses others on the internet and in other digital spaces, particularly on social media sites. Harmful bullying behavior can include posting rumors, threats, sexual remarks, a victims' personal information, or pejorative labels (i.e. hate speech). Bullying or harassment can be identified by repeated behavior and an intent to harm. Victims of cyberbulling may experience lower self-esteem, increased suicidal ideation, and a variety of negative emotional

responses including being scared, frustrated, angry, or depressed. Cyberbullying can take place on social media sites such as Facebook, Myspace, and Twitter. "By 2008, 93% of young people between the ages of 12 and 17 were online. In fact, youth spend more time with media than any single other activity besides sleeping." The last decade has witnessed a surge of cyberbullying, which is categorized as bullying that occurs through the use of electronic communication technologies, such as e-mail, instant messaging, social media, online gaming, or through digital messages or images sent to a cellular phone.

There are many risks attached to social media sites, and cyberbullying is one of the larger risks. One million children were harassed, threatened or subjected to other forms of cyberbullying on Facebook during the past year, while 90 percent of social-media-using teens who have witnessed online cruelty say they have ignored mean behaviour on social media, and 35 percent have done so frequently. Ninety-five percent of social-media-using teens who have witnessed cruel behaviour on social networking sites say they have seen others ignoring the mean behaviour, and 55 percent have witnessed this frequently. Terms like "Facebook depression" have been coined specifically in regard to the result of extended social media use, with cyberbullying playing a large part in this.

OBJECTIVE

This aims to classify textual content into non-hate or hate speech, in which case the method may also identify the targeting characteristics (i.e., types of hate, such as race, and religion) in the hate speech.

PROBLEM STATEMENT

Twitter is a popular social networking website where members create and interact with messages known as “tweets”. This serves as a mean for individuals to express their thoughts or feelings about different subjects. Various different parties such as consumers and marketers have done sentiment analysis on such tweets to gather insights into products or to conduct market analysis. Furthermore, with the recent advancements in machine learning algorithms, we are able improve the accuracy of our sentiment analysis predictions. In this report, we will attempt to conduct sentiment analysis on “tweets” using various different machine learning algorithms. We attempt to

classify the polarity of the tweet where it is either positive or negative. If the tweet has both positive and negative elements, the more dominant sentiment should be picked as the final label. We use the dataset from Kaggle which was crawled and labeled positive/negative. The data pro-vided comes with emoticons, usernames and hashtags which are required to be processed and converted into a standard form. We also need to extract useful features from the text such uni-grams and bigrams which is a form of representation of the "tweet". We use various machine learning algorithms to conduct sentiment analysis using the extracted features. However, just relying on individual models did not give a high accuracy so we pick the top few models to generate a model ensemble. Ensembling is a form of meta learning algorithm technique where we combine different classifiers in order to improve the prediction accuracy. Finally, we report our experimental results and findings at the end.

EXISTING SYSTEM

Existing methods primarily cast the problem as a supervised document classification task [33]. These can be divided into two categories: one relies on manual feature engineering that are then consumed by algorithms such as SVM, Naive Bayes, and Logistic Regression [3, 9, 11, 15, 19, 23, 35–39] (classic methods); the other represents the more recent deep learning paradigm that employs neural networks to automatically learn multi-layers of abstract features from raw data [13, 26, 30, 34] (deep learning methods).The existing system lacks in calculating values with algorithms. These algorithms provide less accuracy.in existing system less data is tested if more data is tested then there will be an operational issue and provides less accuracy

DISADVANTAGES OF EXISTING SYSTEM

- Existing studies on hate speech detection have primarily reported their results using micro-average Precision, Recall and F1 [1, 13, 30, 36, 37, 40].
- The problem with this is that in an unbalanced dataset where instances of one class (to be called the 'dominant class') significantly out-number others (to be called 'minority classes'), micro-averaging can mask the real performance on minority classes.

PROPOSED SYSTEM

All datasets are significantly biased towards non-hate, as hate Tweets account between only 5.8% (DT) and 31.6% (WZ). When we inspect specific types of hate, some can be even scarcer, such as ‘racism’ and as mentioned before, the extreme case of ‘both’. This has two implications. First, an evaluation measure such as the micro F1 that looks at a system’s performance on the entire dataset regardless of class difference can be biased to the system’s ability of detecting ‘non-hate’. In other words, a hypothetical system that achieves almost perfect F1 in identifying ‘racism’ Tweets can still be overshadowed by its poor F1 in identifying ‘non-hate’, and vice versa. Second, compared to non-hate, the training data for hate Tweets are very scarce. This may not be an issue that is easy to address as it seems, since the datasets are collected from Twitter and reflect the real nature of data imbalance in this domain. Thus to annotate more training data for hateful content we will almost certainly have to spend significantly more effort annotating non-hate.

ADVANTAGE OF PROPOSED SYSTEM

- Also, as we shall show in the following, this problem may not be easily mitigated by conventional methods of over- or under-sampling.
- Because the real challenge is the lack of unique, discriminative linguistic characteristics in hate Tweets compared to non-hate.
- As a proxy to quantify and compare the linguistic characteristics of hate and non-hate Tweets, we propose to study the ‘uniqueness’ of the vocabulary for each class.

CHAPTER 2

LITERATURE SURVEY

MOTIVATION

We have chosen to work with twitter since we feel it is a better approximation of public sentiment as opposed to conventional internet articles and web blogs. The reason is that the amount of relevant data is much larger for twitter, as compared to traditional blogging sites. Moreover, the response on twitter is prompterand also more general (since the number of users who tweet is substantially more than those who write web blogs on a daily basis). Sentiment analysis of public is highly critical in macro-scale socioeconomic phenomena like predicting the stock market rate of a particular firm. This could be done by analysing overall public sentiment towards that firm with respect to time and using economics tools for finding the correlation between public sentiment and the firm's stock market value. Firms can also estimate how well their product is responding in the market, which areas of the market is it having a favourable response and in which a negative response (since twitter allows us to download stream of geo-tagged tweets for particular locations. If firms can get this information, they can analyse the reasons behind geographically differentiated response, and so they can market their product in a more optimized manner by looking for appropriate solutions like creating suitable market segments. Predicting the results of popular political elections and polls is also an emerging application to sentiment analysis. One such study was conducted by Tumasjan et al. in Germany for predicting the outcome of federal elections in which concluded that twitter is a good reflection of offline sentiment .

LITERATURE SURVEy

[1] In afsaneh Asaei et al, Perceptual Information Loss because of Impaired Speech Production, Phonological classes characterize without articulatory and articulatory-bound telephone properties. Profound neural system is utilized to gauge the likelihood of phonological classes from the discourse signal. In principle, a one of a kind mix of telephone characteristics structure a phoneme personality. Probabilistic induction of phonological classes in this manner empowers estimation of their compositional phoneme probabilities. An epic data theoretic system is concocted to measure the data passed on by each telephone trait, and survey the discourse creation quality for view of phonemes. As an utilization case, we theorize that interruption in discourse creation prompts data misfortune in telephone properties, and in this manner disarray in phoneme recognizable proof. We evaluate the measure of data misfortune because of dysarthria enunciation recorded in the TORGO database. A tale data measure is figured to assess the deviation from a perfect telephone credit creation driving us to recognize solid creation from obsessive discourse.

[2] duanpei, m.tanaka and R.chen et al, a robust speech detection algorithm for speech activated hands-on application, depicts a novel commotion vigorous discourse discovery calculation that can work dependably in serious vehicle boisterous conditions. Superior has been acquired with the accompanying methods: (1) clamor concealment dependent on head part examination for pre-handling, (2) vigorous endpoint identification utilizing dynamic parameters [I] and (3) discourse check utilizing periodicity of voiced signs with symphonious improvement. Clamor concealment improves the SNR as contrasted and nonlinear range subtraction by around 20 db. This causes the endpoint location to work dependably in SNRs down to - 10 dB. In vehicle situations, street knock clamors are tricky for discourse identifiers causing mis-discovery blunders. Discourse confirmation assists with evacuating these blunders. This innovation is being utilized in Sony vehicle route items.

[3] M.izzad, nursuriati jamil and zainab abu bakar et al, speech/Non-Speech Detection in Malay Language Spontaneous Speech, is to segregate discourse and non-discourse sections in Malay language unconstrained discourse as discourse/non-discourse discovery is significant in numerous discourse handling applications. Off base sentence limits are a significant reason for mistakes in programmed discourse acknowledgment and a preprocessing stage that portions the discourse signal into times of discourse and non-discourse is priceless in improving the acknowledgment precision. We proposed a mix of three sound highlights that is vitality, zero intersection rate (ZCR) and key recurrence (F0) for the discourse/non-discourse location as each element has interesting properties to separate discourse and non-discourse sections. Tests are directed on one-hour Malay language unconstrained discourse comprising of in excess of 20,000 discourse/non-discourse portions. A precision assessment uncovers that the proposed technique accomplished 97.8% exactness rate. On-discourse fragments will additionally be utilized as up-and-comers of sentence limit in our next test

[4] Bujar Raufi, Ildi Xhaferri et al, Application of Machine Learning Techniques for Hate Speech Detection in Mobile Applications, The proliferation of data through various platforms and applications is in constant increase. The versatility of data and its omnipresence makes it very hard to detect the trustworthiness and intention of the source. This is very evident in dynamic environments such as mobile applications. As a result, designing mobile applications that will monitor, control and block any type of malintents is important. Technique used automatic hate speech detection, machine learning, artificial neural networks (ANNs)

[5] Arum Sucia Saksesi, Muhammad Nasrun, Casi Setianingsih et al, Analysis Text of Hate Speech Detection Using Recurrent Neural Network , Twitter is very important for the success and destruction of one's image due to the many sentences of opinion that can compete the users. Examples of phrases that mean evil refer to hate speech to others. Evil perspectives can be categorized in hate speech, which hates speech is regulated in Article 28 of the ITE Law. Not a few people who intentionally and unintentionally oppose social media that contain hate speech. Unfortunately, social media does not have the ability to aggregate information about an existing conversation into a conclusion.

[6] Ioanna K. Lekea, Panagiotis Karampelas et al , Detecting Hate Speech within the Terrorist Argument: A Greek Case , Hate speech can be used by a terrorist group as a means of judging possible targets' guilt and deciding on their punishment, as well as a means of making people to accept acts of terror or even as propaganda for possibly attracting new members. To decide on how the automatic classification will be performed, we experimented with different text analyzing techniques such as critical discourse and content analysis and based on the preliminary results of these techniques a classification algorithm is proposed that can classify the communiqués in three categories depending on the presence of hate speech.

[7] Axel Rodríguez, Carlos Argueta, Yi-Ling Chen et al, Automatic Detection of Hate Speech on Facebook Using Sentiment and Emotion Analysis, Hate speech has been an issue since the start of the Internet, but the advent of social media has brought it to unimaginable heights. To address such an important issue, in this paper, we explore a novel framework to effectively detect highly discussed topics that generate hate speech on Facebook. With the use of graph, sentiment, and emotion analysis techniques, we cluster and analyze posts on prominent Facebook pages. the definition of hate speech is conduct that uses direct assault with words on people who have particular traits, and this kind of assault usually has a tendency of violence or carries a tone of debasement.

CHAPTER 3

METHODOLOGY

AIM & SCOPE

The primary goal of this project is to detect the cyberbullying words or tweets in social media.One million children were harassed, threatened or subjected to other forms of cyberbullying on social media during the past year, while 90 percent of social-media-using teens who have witnessed online cruelty

To stop this type of harassment in social media. The project will detect the bullying words or tweets in social media and try to prevent the behaviour from the user

HARDWARE REQUIREMENTS:

Software engineers use the hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system as the starting point for the system design. It shows what the system does, and it should be implemented

SYSTEM : I3CORE(MIN)

HARD DISK : 120 GB OR ABOVE

RAM :4 GB(MIN) OR ABOVE

KEYBOARD : STANDARD 102 KEYS

MOUSE : 3 BUTTONS

SOFTWARE REQUIREMENTS:

The software requirements are description of features and functionalities of the target system**.** Requirements convey the expectations of users from the software product. The requirements can be obvious or

hidden, known or unknown, expected or unexpected from client's point of view**.**

OPERATING SYSTEM : WINDOWS7. OR ABOVE

CODING LANGUAGE : PYTHON

TOOLS USED :

- ANACONDA NAVIGATOR
- PYTHON BUILT-IN MODULES
    - NUMPY
    - PANDAS
    - MATPLOTLIB
    - SKLEARN
    - SEABORM

SOFTWARE FEATURES

In this part various software are discussed which are used to develop the system

PYTHON

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is Interpreted: Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is like PERL and PHP.

Python is Interactive: You can sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented: Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language: Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

HISTORY OF PYTHON

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, Unix shell, and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

FEATURES OF PYTHON

A simple language which is easier to learn, Python has a very simple and elegant syntax. It's much easier to read and write Python programs compared to other languages like C++, Java, C#. Python makes programming fun and allows you to focus on the solution rather than syntax. If you are a newbie, it's a great choice to start your journey with Python.

- **Free and open source**
  You can freely use and distribute Python, even for commercial use. Not only can you use and distribute software’s written in it, you can even make

changes to the Python's source code. Python has a large community constantly improving it in each iteration.

- **Portability**

  You can move Python programs from one platform to another and run it without any changes. It runs seamlessly on almost all platforms including Windows, Mac OS X and Linux.

- **Extensible and Embeddable**

  Suppose an application requires high performance. You can easily combine pieces of C/C++ or other languages with Python code. This will give your application high performance as well as scripting capabilities which other languages may not provide out of the box.

- **A high-level, interpreted language**

  Unlike C/C++, you don't have to worry about daunting tasks like memory management, garbage collection and so on. Likewise, when you run Python code, it automatically converts your code to the language your computer understands. You don't need to worry about any lower level operations.

- **Large standard libraries to solve common tasks**

  Python has a number of standard libraries which makes life of a programmer much easier since you don't have to write all the code yourself. For example: Need to connect MySQL database on a Web server You can use MySQLdb library using import MySQL db Standard libraries in Python are well tested and used by hundreds of people. So you can be sure that it won't break your application.

- **Object-oriented**

  Everything in Python is an object. Object oriented programming (OOP) helps you solve a complex problem intuitively. With OOP, you are able to divide these complex problems into smaller sets by creating object

NUMPY INTRODUCTION

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array.

Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project.

Operations using NumPy

Using NumPy, a developer can perform the following operations:

- Mathematical and logical operations on arrays.
- Fourier transforms and routines for shape manipulation.
- Operations related to linear algebra. NumPy has in-built functions for linear algebra and random number generation.

NumPy – A Replacement for MatLab

NumPy is often used along with packages like SciPy (Scientific Python) and Mat−plotlib (plotting library). This combination is widely used as a replacement for MatLab, a popular platform for technical computing. However, Python alternative to MatLab is now seen as a more modern and complete programming language.

It is open source, which is an added advantage of NumPy.

Numpy − Environment

Standard Python distribution doesn't come bundled with NumPy module. A lightweight alternative is to install NumPy using popular Python package installer, **pip**.

pip install numpy

The best way to enable NumPy is to use an installable binary package specific to your operating system. These binaries contain full SciPy stack (inclusive of NumPy, SciPy, matplotlib, IPython, SymPy and nose packages along with core Python).

PANDAS

In computer programming, **pandas** is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals.

MATPLOTLIB

matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUItoolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib.

Matplotlib was originally written by John D. Hunter, since then it has an active development community,] and is distributed under a BSD-style license. Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012, and further joined by Thomas Caswell.[6][7]

Matplotlib 2.0.x supports Python versions 2.7 through 3.6. Python 3 support started with Matplotlib 1.2. Matplotlib 1.4 is the last version to support Python 2.6. Matplotlib has pledged to not support Python 2 past 2020 by signing the Python 3 Statement.[

MACHINE LEARNING

Machine learning (ML) is the study of computer algorithms that improve automatically through experience.[1] It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so.[2][3]:2 Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision, where it is difficult or infeasible to develop conventional algorithms to perform the needed task.

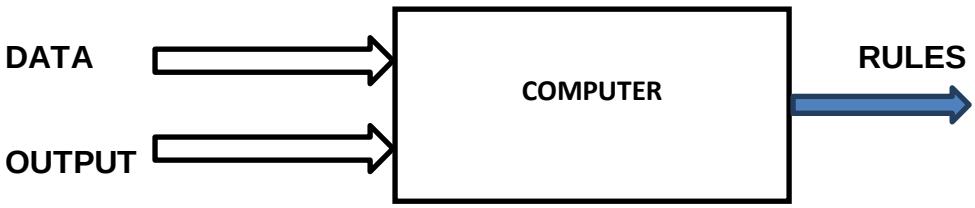


Fig 3.5.1 **Machine Learning**

The core objective of machine learning is the **learning** and **inference**. First of all, the machine learns through the discovery of patterns. This discovery is made thanks to the **data**. One crucial part of the data scientist is to choose carefully which data to provide to the machine. The list of attributes used to solve a problem is called a **feature vector.** You can think of a feature vector as a subset of data that is used to tackle a problem. The machine uses some fancy algorithms to simplify the reality and transform this discovery into a **model**. Therefore, the learning stage is used to describe the data and summarize it into a model.

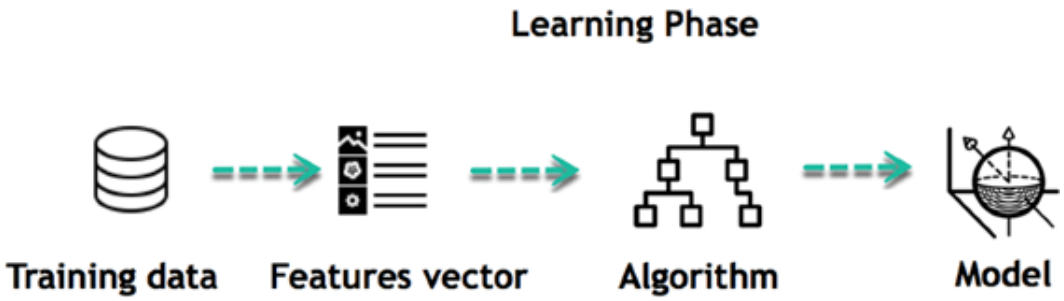


Fig 3.5.2 learning phase

For instance, the machine is trying to understand the relationship between the wage of an individual and the likelihood to go to a fancy restaurant. It turns out the machine finds a positive relationship between wage and going to a high-end restaurant: This is the model

Inferring

When the model is built, it is possible to test how powerful it is on never-seen-before data. The new data are transformed into a features vector, go through the model and give a prediction. This is all the beautiful part of machine learning. There is no need to update the rules or train again the model. You can use the model previously trained to make inference on new data.

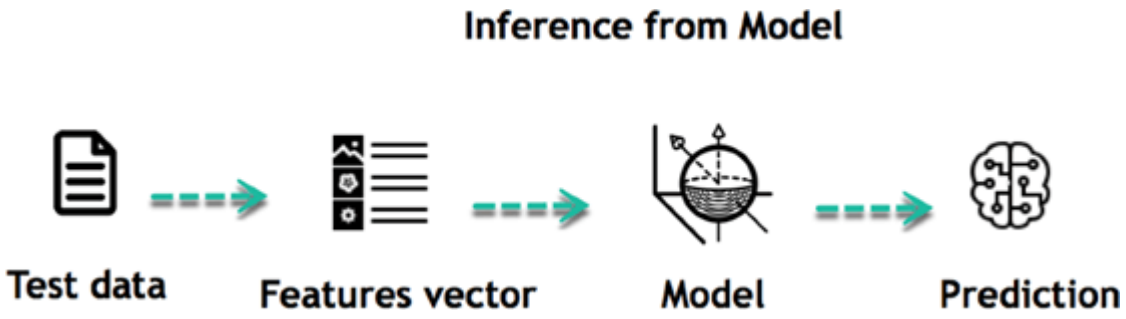


Fig 3.5.3 inference model

Machine learning Algorithms and where they are used?

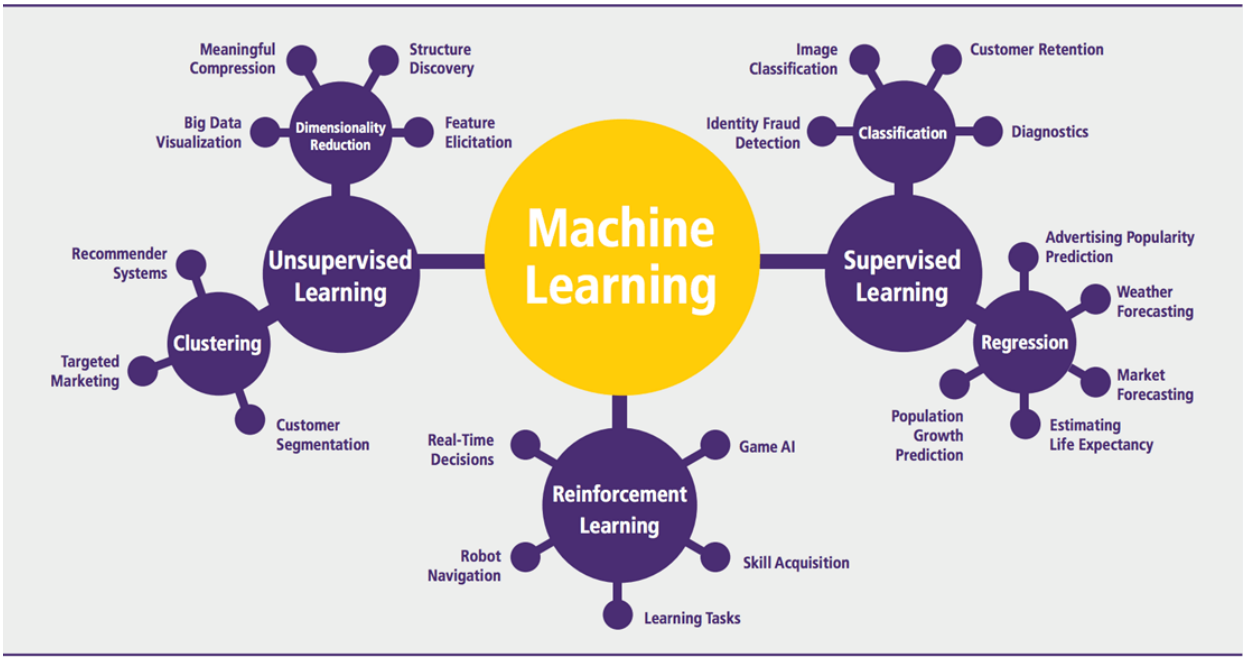


Fig 3.5.4 Machine learning algorithms

Machine learning can be grouped into two broad learning tasks: Supervised and Unsupervised. There are many other algorithms

SUPERVISED LEARNING

An algorithm uses training data and feedback from humans to learn the relationship of given inputs to a given output. For instance, a practitioner can use marketing expense and weather forecast as input data to predict the sales of cans.

You can use supervised learning when the output data is known. The algorithm will predict new data.

There are two categories of supervised learning:

- Classification task
- Regression task

Classification

Imagine you want to predict the gender of a customer for a commercial. You will start gathering data on the height, weight, job, salary, purchasing basket, etc. from your

customer database. You know the gender of each of your customer, it can only be male or female. The objective of the classifier will be to assign a probability of being a male or a female (i.e., the label) based on the information (i.e., features you have collected). When the model learned how to recognize male or female, you can use new data to make a prediction. For instance, you just got new information from an unknown customer, and you want to know if it is a male or female. If the classifier predicts male = 70%, it means the algorithm is sure at 70% that this customer is a male, and 30% it is a female

Regression

When the output is a continuous value, the task is a regression. For instance, a financial analyst may need to forecast the value of a stock based on a range of feature like equity, previous stock performances, macroeconomics index. The system will be trained to estimate the price of the stocks with the lowest possible error.

UNSUPERVISED LEARNING

In unsupervised learning, an algorithm explores input data without being given an explicit output variable (e.g., explores customer demographic data to identify patterns)

You can use it when you do not know how to classify the data, and you want the algorithm to find patterns and classify the data for you

TENSORFLOW

The most famous deep learning library in the world is Google's TensorFlow. Google product uses machine learning in all of its products to improve the search engine, translation, image captioning or recommendations.

To give a concrete example, Google users can experience a faster and more refined the search with AI. If the user types a keyword a the search bar, Google provides a recommendation about what could be the next word.

TensorFlow Architecture

TensorFlow architecture works in three parts:

- Pre-processing the data
- Build the model
- Train and estimate the model

It is called TensorFlow because it takes input as a multi-dimensional array, also known as **tensors**. You can construct a sort of **flowchart** of operations (called a Graph) that you want to perform on that input. The input goes in at one end, and then it flows through this system of multiple operations and comes out the other end as output.

This is why it is called TensorFlow because the tensor goes in it flows through a list of operations, and then it comes out the other side.

MODULES EXPLANATION

The process of detecting cyberbully words from input dataset. An Input dataset is sent to data pre-processing which is applied to improve the quality of the input data. The data pre-processing also includes removing stop words and special characters. After performed the data preprocessing the output data is sent to classification algorithm for detecting the cyberbully words in tweets.

The proposed work will detect Cyberbully words on tweets from input dataset, which includes 3 Steps,

Data Preprocessing

Cyberbully Detection

Classification

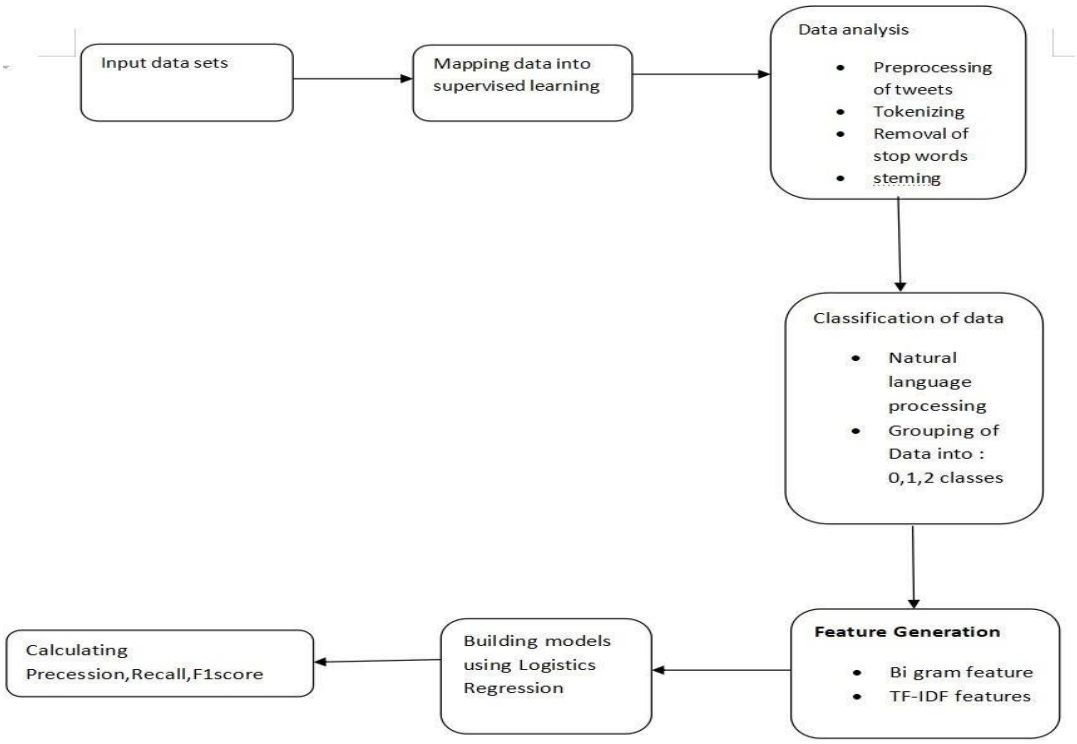


Fig 3.6.1 architecture diagram

Data Preprocessing

Social network data are noisy, thus preprocessing has been applied to improve the accuracy of the input data. This includes removing stop words. Stop words are usually like “a”, “as”, “have”, “is”, “the”, “or”, etc. Stop words mainly used for consumed memory space and processing time.

Feature Extraction

Noun Adjective and pronoun Extraction

Parsing noun, adjective and pronoun involves two steps such as part-of-speech tagging and extracting noun, adjective and pronoun from the tagged output. The part-of-speech tagging(POS tagging or POST), also called grammatical tagging, is the process of marking up a word in a text as corresponding to apart of speech.

The Part-of-speech tagging is carried out using the package provided by Stanford Natural Language Processing.

Frequency Extraction

The frequency extraction module involves extracting the occurrence count of the words parsed in the parser module. Using the frequency of the word .

Cyberbully Detection

In cyberbully detection, the bullying words in the tweet contents are detected using the machine learning algorithms. After getting the output from the preprocessing step, the output file will sent to the classification algorithms. There the trained classifier will be used for detection. The training dataset consist of list of cyberbullying words. With the training dataset the preprocessed twitter dataset is tested for bullying word presence or not. The natural language processing and the logistic regression are mainly used to detect the cyberbully words present in the tweet contents.

Data Description:

The data given is in the form of a comma-separated values files with tweets and their correspond-ing sentiments. The training dataset is a csv file of type tweet_id,sentiment,tweet where the tweet_id is a unique integer identifying the tweet, sentiment is either 1 (positive) or 0 (neg-ative), and tweet is the tweet enclosed in "". Similarly, the test dataset is a csv file of type tweet_id,tweet.

CLASSIFICATION ALGORITHMS

NATURAL LANGUAGE PROCESSING

Natural Language Processing is the technology used to aid computers to understand the human's natural language. It's not an easy task teaching machines to understand how we communicate.

LeandRomaf, an experienced software engineer who is passionate at teaching people how artificial intelligence systems work, says that “in recent years, there have been significant breakthroughs in empowering computers to understand language just as we do.”

This article will give a simple introduction to Natural Language Processing and how it can be achieved.

What is Natural Language Processing?

Natural Language Processing, usually shortened as NLP, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language.The ultimate objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable.Most NLP techniques rely on machine learning to derive meaning from human languages.

What are the techniques used in NLP?

Syntactic analysis and semantic analysis are the main techniques used to complete Natural Language Processing tasks.

Here is a description on how they can be used.

Syntax

Syntax refers to the arrangement of words in a sentence such that they make grammatical sense.

In NLP, syntactic analysis is used to assess how the natural language aligns with the grammatical rules.

Computer algorithms are used to apply grammatical rules to a group of words and derive meaning from them.

Here are some syntax techniques that can be used:

Lemmatization: It entails reducing the various inflected forms of a word into a single form for easy analysis.

- **Morphological segmentation**: It involves dividing words into individual units called morphemes.
- **Word segmentation**: It involves dividing a large piece of continuous text into distinct units.
- **Part-of-speech tagging**: It involves identifying the part of speech for every word.
- **Parsing**: It involves undertaking grammatical analysis for the provided sentence.
- **Sentence breaking**: It involves placing sentence boundaries on a large piece of text.
- **Stemming**: It involves cutting the inflected words to their root form.

Semantics

Semantics refers to the meaning that is conveyed by a text. Semantic analysis is one of the difficult aspects of Natural Language Processing that has not been fully resolved yet.

It involves applying computer algorithms to understand the meaning and interpretation of words and how sentences are structured.

Here are some techniques in semantic analysis:

- **Named entity recognition (NER):** It involves determining the parts of a text that can be identified and categorized into preset groups. Examples of such groups include names of people and names of places.
- **Word sense disambiguation:** It involves giving meaning to a word based on the context.

- **Natural language generation**: It involves using databases to derive semantic intentions and convert them into human language

LOGISTIC REGRESSION

In statistics, the **logistic model** (or **logit model**) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc. Each object being detected in the image would be assigned a probability between 0 and 1 and the sum adding to one.

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, **logistic regression** (or **logit regression**) is estimating the parameters of a logistic model (a form of binary regression). Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labelled "0" and "1". In the logistic model, the log-odds (the logarithm of the odds) for the value labelled "1" is a linear combination of one or more independent variables ("predictors"); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value). The corresponding probability of the value labeled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The unit of measurement for the log-odds scale is called a logit, from ***log****istic un****it***, hence the alternative names. Analogous models with a different sigmoid functioninstead of the logistic function can also be used, such as the probit model; the defining characteristic of the logistic model is that increasing one of the independent variables multiplicatively scales the odds of the given outcome at a *constant* rate, with each independent variable having its own parameter; for a binary dependent variable this generalizes the odds ratio.

CHAPTER 4

RESULTS

	precision	recall	f1-score	support
0	0.56	0.12	0.19	164
1	0.90	0.97	0.93	1905
2	0.84	0.81	0.83	410
accuracy			0.88	2479
macro avg	0.77	0.63	0.65	2479
weighted avg	0.87	0.88	0.86	2479

Accuracy Score: 0.8846308995562727

the data is tested under bi gram and the accuracy has been calculated

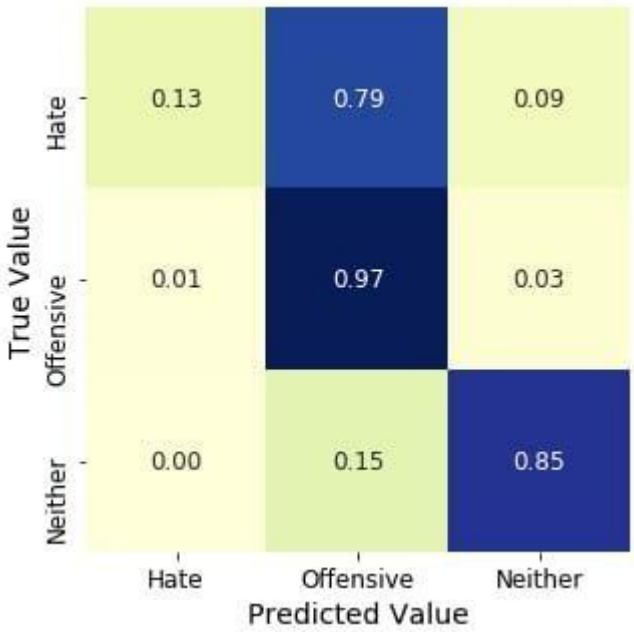
	precision	recall	f1-score	support
0	0.59	0.13	0.22	164
1	0.91	0.97	0.94	1905
2	0.84	0.85	0.84	410
accuracy			0.89	2479
macro avg	0.78	0.65	0.67	2479
weighted avg	0.88	0.89	0.87	2479

Accuracy Score: 0.891085114965712

The data is tested under td idf mode and the accuracy score is calculated and also macro avg, weighted avg.

	Neg	Pos	Neu	Compound	url_tag	mention_tag	hash_tag
0	0.000	0.120	0.880	0.4563	0.0	1.0	0.0
1	0.237	0.000	0.763	-0.6876	0.0	1.0	0.0
2	0.538	0.000	0.462	-0.9550	0.0	2.0	0.0
3	0.000	0.344	0.656	0.5673	0.0	2.0	0.0
4	0.109	0.229	0.662	0.6331	0.0	1.0	1.0
...	...	...	...	...	...	...	...
24778	0.000	0.000	1.000	0.0000	0.0	3.0	3.0
24779	0.454	0.000	0.546	-0.8074	0.0	0.0	0.0
24780	0.000	0.219	0.781	0.4738	0.0	0.0	0.0
24781	0.573	0.000	0.427	-0.7717	0.0	0.0	0.0
24782	0.000	0.218	0.782	0.5994	1.0	0.0	0.0

24783 rows × 7 columns



This is the final output in which the true values and predicted values are denoted . The values denote the percentage of classified words of hate, offensive, neither.

CHAPTER 5

CONCLUSION

Conclusion:

As hate speech continues to be a societal problem, the need for automatic hate speech detection systems becomes more apparent. In this report, we proposed a solution to the detection of hate speech and offensive language on Twitter through machine learning using Bag of Words and TF IDF values. We performed comparative analysis of Logistic Regression, Naive Bayes, Decision Tree, Random Forest and Gradient Boosting on various sets of feature values and model parameters. The results showed that Logistic Regression performs comparatively better with the TF IDF approach. We presented the current problems for this task and our system that achieves reasonable accuracy (89%) as well as recall (84%). Given all the challenges that remain, there is a need for more research on this problem, including both technical and practical matters.

Future work

Generate new future's like removing those cyberbullying tweets that are posted in social media . increase more accuracy in prediction and warring the user about their tweets. We believe there are ways that design can help stop online aggression. Adding live detection or printing*** for bullying words can help in reduce of bullying or online distress. Better feedback from sites can encourage users to report aggression or harassment. Finally, existing designs can help support low-risk interventions.As we've seen, designing to help bystanders takes careful planning. It also requires sensitivity for the ways people use social media. Still, there's no shortage of ways to empower bystanders to stand up against online bullying.

REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. the Journal of machine Learning research, 3:993–1022, 2003.

[2] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), 2(3):27, 2011.

[3] G. Chowdhury. Introduction to modern information retrieval. Facet publishing, 2010.

[4] K. Dinakar, R. Reichart, and H. Lieberman. Modeling the detection of textual cyberbullying. In The Social Mobile Web, 2011.

[5] M. Fekkes, F. I. Pijpers, A. M. Fredriks, T. Vogels, and S. P. Verloove-Vanhorick. Do bullied children get ill, or do ill children get bullied? a prospective cohort study on the relationship between bullying and health-related symptoms. Pediatrics, 117(5):1568–1574, 2006.

[6] G. Gini and T. Pozzoli. Association between bullying and psychosomatic problems: A meta-analysis. Pediatrics, 123(3):1059–1065, 2009.

[7] F. Godin, B. Vandersmissen, W. De Neve, and R. Van de Walle. Named entity recognition for twitter microposts using distributed word representations. In Proceedings of the Workshop on Noisy User-generated Text, pages 146–153, Beijing, China, July 2015. Association for Computational Linguistics.

[8] J. Juvonen and E. F. Gross. Extending the school grounds?ˆaAˇTbullying experiences in cyberspace. ˇ Journal of School health, 78(9):496–505, 2008.

[9] R. M. Kowalski, G. W. Giumetti, A. N. Schroeder, and M. R. Lattanner. Bullying in the digital age: A critical review and meta-analysis of cyberbullying research among youth. 2014.

[10] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. Discourse processes, 25(2-3):259–284, 1998.

[11] B. L. McLaughlin, A. A. Braga, C. V. Petrie, M. H. Moore, et al. Deadly Lessons:: Understanding Lethal School Violence. National Academies Press, 2002.

[12] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pages 3111–3119, 2013.

[13] V. Nahar, X. Li, and C. Pang. An effective approach for cyberbullying detection. Communications in Information Science and Management Engineering, 2012.

[14] V. Nahar, X. Li, and C. Pang. An effective approach for cyberbullying detection. Communications in Information Science and Management Engineering, 3(5):238, 2013.

[15] H. C. Wu, R. W. P. Luk, K. F. Wong, and K. L. Kwok. Interpreting tf-idf term weights as making relevance decisions. ACM Transactions on Information Systems (TOIS), 26(3):13, 2008.

[16] J.-M. Xu, K.-S. Jun, X. Zhu, and A. Bellmore. Learning from bullying traces in social media. In Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies, pages 656–666. Association for Computational Linguistics, 2012.

[17] M. Ybarra. Trends in technology-based sexual and non-sexual aggression over time and linkages to nontechnology aggression. National Summit on Interpersonal Violence and Abuse Across the Lifespan: Forging a Shared Agenda, 2010.

[18] R. Zhao and K. Mao. Semi-random projection for dimensionality reduction and extreme learning machine in high-dimensional space. Computational Intelligence Magazine, IEEE, 10(3):30–41, 2015.

APENDEX

SOURCE CODE

"cells": [

{

"cell_type": "code",

"execution_count": 1,

"metadata": {},

"outputs": [],

"source": [

"import pandas as panda\n",

"from nltk.tokenize import word_tokenize\n",

"from nltk.corpus import stopwords\n",

"from nltk.stem.porter import *\n",

"import string\n",

"import nltk\n",

"from sklearn.feature_extraction.text import CountVectorizer\n",

"from sklearn.feature_extraction.text import TfidfVectorizer\n",

"from sklearn.metrics import confusion_matrix\n",

"import seaborn\n",

"from textstat.textstat import *\n",

"from sklearn.linear_model import LogisticRegression\n",

"from sklearn.model_selection import train_test_split\n",

"from sklearn.metrics import f1_score\n",

"from sklearn.feature_selection import SelectFromModel\n",

"from sklearn.metrics import classification_report\n",

"from sklearn.metrics import accuracy_score\n",

"from sklearn.svm import LinearSVC\n",

"import numpy as np\n",

"from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer as VS\n",

"import warnings\n",

"warnings.simplefilter(action='ignore', category=FutureWarning)\n",

"%matplotlib inline\n",

1. Removal of punctuation and capitlization\n",

"## 2. Tokenizing\n",

"## 3. Removal of stopwords\n",

"## 4. Stemming\n"

"stopwords = nltk.corpus.stopwords.words(\"english\")\n"

"#extending the stopwords to include other words used in twitter such as retweet(rt) etc.\n",

"other_exclusions = [\"#ff\", \"ff\", \"rt\"]\n",

"stopwords.extend(other_exclusions)\n",

"stemmer = PorterStemmer()\n"

"def preprocess(tweet): \n",

"   # removal of extra spaces\n",

"   regex_pat = re.compile(r'\\s+')\n",

"   tweet_space = tweet.str.replace(regex_pat, ' ')\n",

"   # removal of @name[mention]\n",

"   regex_pat = re.compile(r'@[\\w\\-]+')\n",

"   tweet_name = tweet_space.str.replace(regex_pat, '')\n",

"   # removal of links[https://abc.com]\n",

"   giant_url_regex = re.compile('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'\n",

"   '[!*\\(\\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')\n",

"   tweets = tweet_name.str.replace(giant_url_regex, '')\n",

"   # removal of punctuations and numbers\n",

```
"    punc_remove = tweets.str.replace(\"[^a-zA-Z]\", \" \")\n",
"    # removal of capitalization\n",
"    tweet_lower = punc_remove.str.lower()\n",
"    # tokenizing\n",
"    tokenized_tweet = tweet_lower.apply(lambda x: x.split())\n",
"    # removal of stopwords\n",
"    tokenized_tweet= tokenized_tweet.apply(lambda x: [item for item in x if item not in stopwords])\n",
"    # stemming of the tweets\n",
"    tokenized_tweet = tokenized_tweet.apply(lambda x: [stemmer.stem(i) for i in x]) \n",
"    for i in range(len(tokenized_tweet)):\n",
"        tokenized_tweet[i] = ' '.join(tokenized_tweet[i])\n",
"        tweets_p= tokenized_tweet\n",
"    return tweets_p\n",
"processed_tweets = preprocess(tweet)   \n",
"dataset['processed_tweets'] = processed_tweets\n",
"dataset"
"# visualizing which of the word is most commonly used in the twitter dataset\n",
"import matplotlib.pyplot as plt\n",
 "from wordcloud import WordCloud\n",
"all_words = ' '.join([text for text in dataset['processed_tweets'] ])\n",
"wordcloud = WordCloud(width=800, height=500, random_state=21, max_font_size=110).generate(all_words)\n",
   "\n",
   "plt.figure(figsize=(10, 7))\n",
   "plt.imshow(wordcloud, interpolation=\"bilinear\")\n",
```

```
    "plt.axis('off')\n",

    "plt.show()"

"# visualizing which of the word is most commonly used for offensive speech\n",

"offensive_words = ' '.join([text for text in dataset['processed_tweets'][dataset['class'] == 2]])\n",

    "wordcloud = WordCloud(width=800, height=500,\n",

    "random_state=21, max_font_size=110).generate(offensive_words)\n",

    "plt.figure(figsize=(10, 7))\n",

    "plt.imshow(wordcloud, interpolation=\"bilinear\")\n",

    "plt.axis('off')\n",

    "plt.show()"

"# Bigram Features\n",

    "bigram_vectorizer = CountVectorizer(ngram_range=(1,2),max_df=0.75, min_df=1, max_features=10000)\n",

    "# bigram feature matrix\n",

    "bigram = bigram_vectorizer.fit_transform(processed_tweets).toarray()\n",

"#TF-IDF Features\n",

    "tfidf_vectorizer = TfidfVectorizer(ngram_range=(1, 2),max_df=0.75, min_df=5, max_features=10000)\n",

    "# TF-IDF feature matrix\n",

    "tfidf = tfidf_vectorizer.fit_transform(dataset['processed_tweets'] )\n",

    "tfidf"

"# Using Bigram Features\n",

    "X = panda.DataFrame(bigram)\n",

    "y = dataset['class'].astype(int)\n",

    "X_train_bow, X_test_bow, y_train, y_test = train_test_split(X, y, random_state=42, test_size=0.1)\n",
```

```
    "model = LogisticRegression(class_weight='balanced',penalty=\"l1\",
C=0.01).fit(X_train_bow,y_train)\n",

    "y_preds = model.predict(X_test_bow)\n",

    "report = classification_report( y_test, y_preds )\n",

    "print(report)\n",

    "print(\"Accuracy Score:\" , accuracy_score(y_test,y_preds))\n"

# Running the model Using TFIDF without additional features\n",

    "X = tfidf\n",

    "y = dataset['class'].astype(int)\n",

    "X_train_tfidf, X_test_tfidf, y_train, y_test = train_test_split(X, y,
random_state=42, test_size=0.1)\n",

    "model = LogisticRegression().fit(X_train_tfidf,y_train)\n",

    "y_preds = model.predict(X_test_tfidf)\n",

    "report = classification_report( y_test, y_preds )\n",

    "print(report)\n",

    "print(\"Accuracy Score:\" , accuracy_score(y_test,y_preds))"
```