PROJECT REPORT

ON

SENTIMENTAL ANALYSIS AND DEPRESSION DETECTION IN TWEETS USING MACHINE LEARNING

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

The advent of different social networking sites has enabled anyone to easily create, express, and share their ideas, thoughts, opinions, and feelings about anything with millions of other people around the world. With the advancement of technology, minicomputers and smartphones have come into human pockets and now it is very easy to share your idea about anything on social media platforms like Facebook, Twitter, Wikipedia, LinkedIn, Google+, Instagram, etc. Due to the tremendous growth in population and communication technologies during the last decade, the use of social networks is on the rise and they are being used for many different purposes. One such service for which their use may be explored is an analysis of users' posts to diagnose depression. Depression is a common illness worldwide with potentially severe implications. Early identification of depressive symptoms is a crucial first step towards assessment, intervention, and relapse prevention. With an increase in data sets with relevance for depression and the advancement of machine learning, there is a potential to develop intelligent systems to detect symptoms of depression in written material. Sentiment analysis is the process of detecting positive or negative sentiment in text. It's often used by businesses to detect sentiment in social data, gauge brand reputation, and understand customers. Sentiment analysis focuses on the polarity of a text (positive, negative, neutral) but it also goes beyond polarity to detect specific feelings and emotions (angry, happy, sad, etc), urgency (urgent, not urgent), and even intentions (interested v. not interested). This project suggests the use of sentiment analysis classification as an effective method for examining textual data coming from a variety of resources on the internet. Sentiment analysis is a method of data mining that evaluates textual data-consuming machine learning techniques. Due to the tremendous expanse of opinions of users, their reviews, feedback, and suggestions available over the web resources, it is so much indispensable to discover, analyze and consolidate their views for enhanced decision making. Sentiment analysis presents an effective and efficient opinion of consumers in real-time which can greatly affect the decision-making process in the business domain.

CHAPTER 1: INTRODUCTION

CHAPTER 1: INTRODUCTION

1.1. Introduction to Sentimental Analysis

In recent years, a huge number of people have been attracted to social networking platforms like Facebook, Twitter, and Instagram. Most use social sites to express their emotions, beliefs, or opinions about things, places, or personalities. Methods of sentiment analysis can be categorized predominantly as machine-learning, Lexicon-based, and hybrid. Similarly, another categorization has been presented with the categories of statistical, knowledge-based, and hybrid approaches. There is a space for performing challenging research in broad areas by computationally analyzing opinions and sentiments. Therefore, a gradual practice has grown to extract the information from data available on social networks for the prediction of an election, to use for educational purposes, or for the fields of business, communication, and marketing. The accuracy of sentiment analysis and predictions can be obtained by behavioral analysis based on social networks.

Depression is a common mental illness and a leading cause of disability worldwide, which may cause suicide. Globally, more than 300 million people are estimated to suffer from depression every year. Generally, depression is diagnosed through face-to-face clinical depression criteria. However, in the early stages of depression, 70% of the patients would not consult doctors, which may take their condition to advance stages. Recently, there has been a movement to leverage social medial data for detecting, estimating, and tracking the changes in the occurrence of a disease. The ubiquity of social media provides a rich opportunity to enhance the data available to mental health clinicians and researchers, enabling a better-informed and -equipped mental health field. In addition, contagious negative emotions in social networks adversely affect people, leading to depression and other mental illnesses. Mental illness is known as a major risk factor for suicide; almost 80% of those who attempt or die by suicide are known to have had some form of mental illness. Depression is considered the most common mental illness, but because of its unrecognition or denial, it has remained undiagnosed or untreated. The onset of major depression can be prevented by early recognition of its symptoms and their treatment through timely intervention. Many studies have detected physical and mental illnesses derived from social media's huge information, in precise, some studies were dedicated to depression found that tweets posted by individuals with major depressive disorder, as well as their social media activity, can be utilized to classify and predict if they are suffering from depression or are likely to suffer in the future.

In this project, we aim to perform a depression analysis on Facebook data collected from an online public source. To investigate the effect of depression detection, we propose the machine learning technique as an efficient and scalable method. We have evaluated the efficiency of our proposed method using a set of various psycholinguistic features. We show that our proposed method can significantly improve the accuracy and classification error rate. In addition, the result shows that in different experiments Decision

Tree (DT) gives the highest accuracy of other ML approaches to finding the depression. One of the applications of machine learning techniques is to study pattern recognition and the development of computational systems that can learn themselves. The machine learning algorithms, train the system on the basis of available training data sets and after the training, the system can predict the future data values. We present how to find the depression level of a person by observing and extracting emotions from the text, using emotion theories, machine learning techniques, and natural language processing techniques on data collected from different social media platforms.

This project also aims to detect whether the user is depressed, from the nature of his/her tweets and activity in the network. It can be further used to identify other mental illnesses and might even form an underlying infrastructure for new mechanisms that would help detect and limit depression diffusion in social networks.

The main purpose of selecting Twitter's profile data is that we can get qualitative information from this platform because Twitter contains the authenticated accounts of politicians, which is not the case with Facebook or Instagram etc. Additionally, in contrast with Facebook, Twitter restricts users to give their compact and complete opinions in 280 characters. Recent studies have proven that with Twitter it is possible to get people's insight from their profiles in contrast to traditional ways of obtaining information about perceptions.

Depression is a mental health disorder that is generally characterized by frequent mood swings, loss of interest and pleasure, lack of concentration, varying sleep and appetite, feelings of low self-worth, and more similar symptoms. Depression is a phenomenon that most people nowadays are familiar with, either from a personal experience or through a person close to them, a friend, family, or a relative, who is suffering from it. Based on the reports by the World Health Organization (WHO) and the World Bank, depression is currently ranked as the major cause of disability throughout the globe. The World Health Organization (WHO) has even gone so far as to predict that it will become the top physical or mental disorder worldwide by the year 2030. Recently, the treatment of depressive disorders has progressed a lot and is proven to be effective in most cases. Even so, detecting depression in an individual and accurately diagnosing it is still considered a challenge and a barrier to effective treatment. Detecting an individual is suffering from depression is not an easy task. It is completely based on the individual reporting it, either by themselves or through a person close to them. It is rarely found that an individual is diagnosed accurately on the basis of a clinical judgment of symptom severity. Based on the analysis done by the World Health Organization (WHO), currently, there isn't any reliable and effective diagnosis of depression and it is mainly due to the overall lack of proper resources and trained health care providers. With it becoming ever so prominent and an even more pressing issue, specific measures need to be taken to identify depression in an individual in its early stage and then, use suitable measures to treat the individual. Also, there have to be certain measures that can prevent depression altogether.

1.2. What is Sentimental Analysis ?

Sentiment analysis is powered by natural language processing (NLP) and machine learning (ML) algorithms. These artificially intelligent bots are trained on millions of pieces of text to detect if a message is positive, negative, or neutral. Sentiment analysis works by breaking a message down into topic chunks and then assigning a sentiment score to each topic.
For example, take the following social post:
I tried out the new Dell G5 Gaming computer. I was really impressed. The graphics card was a little disappointing, but it's hard to beat the G5 at that price.
A sentiment analysis tool would break this into topic chunks and then assign a sentiment score to each topic, depending on a pre-determined scale:

- Dell G5 Gaming computer…really impressed = +4
- graphics card…disappointing…= -2
- hard to beat…that price = +3

The bot would then sum up the scores or use each score individually to evaluate components of the statement. In this case, there was an overall positive sentiment, but a negative sentiment towards the graphics card.

1.3. Feasibility of Sentimental Analysis

Sentiment analysis has many beneficial applications:

1.3.1. Gauge Public Opinion in Real-Time

If you launch a new marketing campaign, product, or service, you can quickly track public opinion in real-time. For example, if everyone hates the jingle in your new commercial, it may be time to go back to studio

1.3.2. Conduct Market Research

You can also use sentiment analysis to benchmark your brand against competitors or to research what's hot on the market. If people are feeling negative about your brand but love your competitor, you may want to pivot your strategy.

1.3.3. Track Your Customer Service

If your brand uses social for customer service inquiries, you can track the positive-negative impact of your support efforts. For example, how often your team is able to shift a customer's negative sentiment to a positive one.

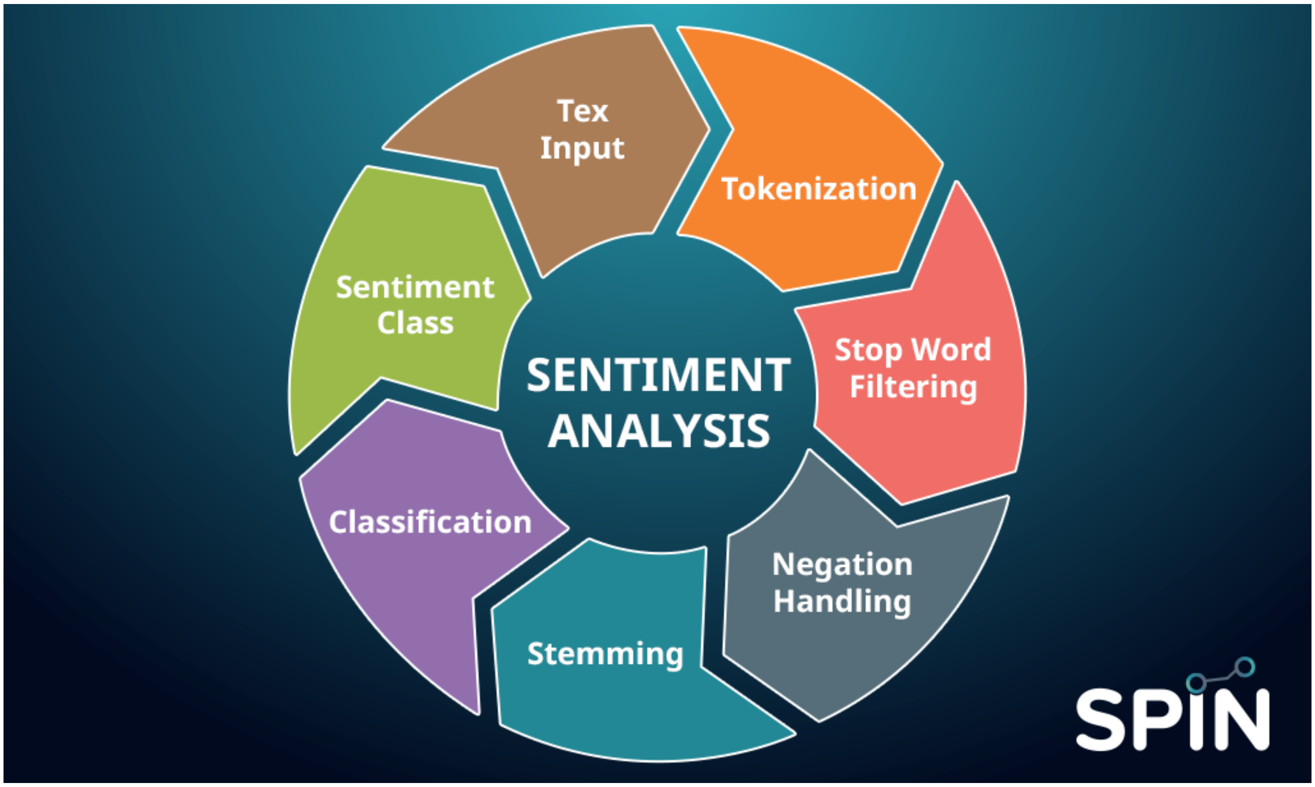


Figure 1: The Benefits of Using Sentiment Analysis

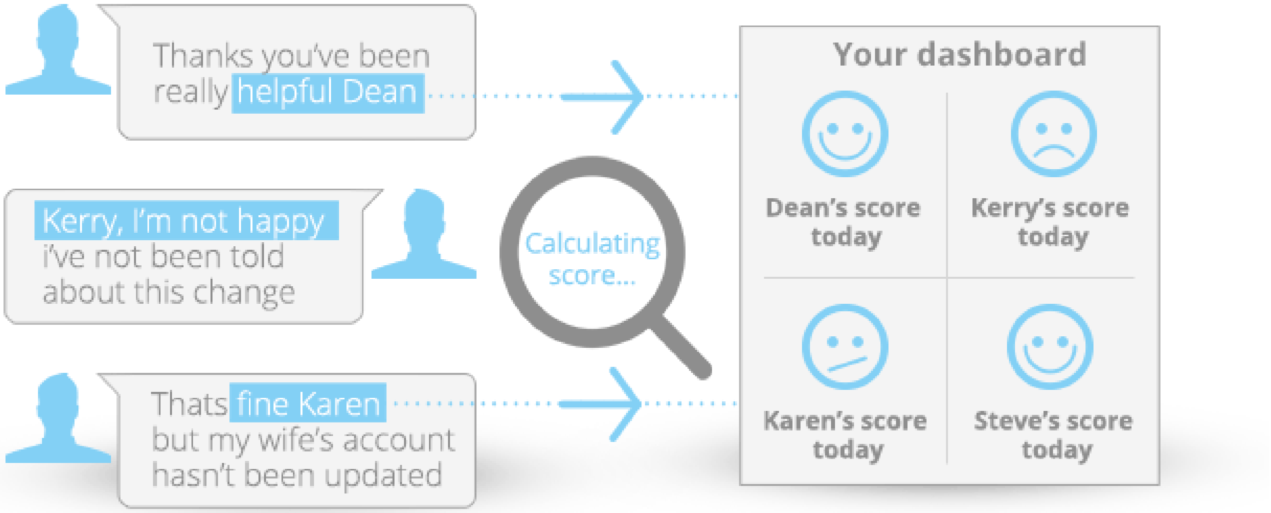


Figure 2: Using Sentiment Analysis for Agent Monitoring - Sentiment analysis calculates agent

satisfaction scores

1.4. Motivation

Since humans express their thoughts and feelings more openly than ever before, sentiment analysis is fast becoming an essential tool to monitor and understand the sentiment in all types of data.

Automatically analyzing customer feedback, such as opinions in survey responses and social media conversations, allows brands to learn what makes customers happy or frustrated so that they can tailor products and services to meet their customers' needs.

For example, using sentiment analysis to automatically analyze 4,000+ open-ended responses in your customer satisfaction surveys could help you discover why customers are happy or unhappy at each stage of the customer journey.

Maybe you want to track brand sentiment so you can detect disgruntled customers immediately and respond as soon as possible. Maybe you want to compare sentiment from one quarter to the next to see if you need to take action. Then you could dig deeper into your qualitative data to see why sentiment is falling or rising.

1.5. Existing Illustrations

To understand the goal and challenges of sentiment analysis, here are some examples:

Basic examples of sentiment analysis data

- Netflix has the best selection of films
- Hulu has a great UI
- I dislike the new crime series
- I hate waiting for the next series to come out

More challenging examples of sentiment analysis

- I do not dislike horror movies. (a phrase with negation)
- Disliking horror movies is not uncommon. (negation, inverted word order)
- Sometimes I really hate the show. (adverbial modifies the sentiment)
- I love having to wait two months for the next series to come out! (sarcasm)
- The final episode was surprising with a terrible twist at the end (a negative term used in a positive way)
- The film was easy to watch but I would not recommend it to my friends. (difficult to categorize)

1.6. Specifications

The following are the system specifications.

1.6.1. Hardware Specifications

- **Hardware Requirements**. Intel Core i7 Octa Core Edition. 96 GB DDR4 RAM. Nvidia 1080 Ti

1.6.2. Software Specifications

- **Software Requirements**: Operating system - Windows 11, Windows 10 or Linux including Ubuntu, Kali etc

- **System architecture:** Windows- 64-bit x86, 32-bit x86; macOS- 64-bit x86 & Apple M1 (ARM64); Linux- 64-bit x86, 64-bit aarch64 etc.

- **Programming Language** - Python 3.6.

- **Platform used** - Jupyter , Tensorboard.

1.7. Our Approach

We have worked on modeling two models with different approaches in order to find the algorithm which provides maximum precision and accuracy. The two models are similar in working but one focuses on Detecting Depression or negativity in the input message from a user after training and testing the model using a dataset. This model gives an accuracy of 95-96%. The other focuses on mainly the sentimental analysis of texts/tweets by segregating them into categories of positive, negative, or neutral, ultimately giving us data that can be put to use to enhance further models already in use. The precision of this model is 0.9456 and the accuracy of this model is 94%

CHAPTER 2: LITERATURE REVIEW

CHAPTER 2: LITERATURE REVIEW

2.1. Background Study/Literature Survey

Depression is one of the most serious and quite commonly diagnosed mental disorders. It affects not only the sufferers but also their families, friends, and even society in general. With rapid advancements in Artificial Intelligence and Machine Learning, there have been some recent developments that aim to predict the severity of depression in an individual by analyzing certain parameters extracted from their video sample.

With it becoming prominent and an even more pressing issue measures need to be taken to identify depressed individuals in their early stages and then, use measures to treat the individual. Also, there are certain measures that can prevent depression altogether. There has been a rapid upsurge in Artificial Intelligence technologies and their use in various domains of recent times. One such domain is seeing changes due to AI advancements in healthcare. AI techniques focus on obtaining detailed information for classification purposes. This can be useful in various things such as mental health research, to accurately characterize the different psychiatric disorders research has been conducted in the field o detection and analysis.

With a rise in mental health issues and cases throughout the world, it has now become a matter of major concern. The effects of depression are tremendous, both to the individual suffering from it and to the entire society as well. With a recent rise in Artificial Intelligence (AI) and Deep Learning technologies, it can be put to good use in the field of healthcare - to better detect and predict mental health issues such as depression early on and treat them before they can cause much harm.

The comparison of the two politicians was made on the basis of real-time Twitter data, extracted from Twitter by using the Twitter-streaming application programming interface (API). Twitter streaming API was also used to gather data by the authors for the prediction of the Indonesian presidential elections. The aim was to use Twitter data to understand public opinion. For this purpose, after the collection of data, the study performed automatic buzzer detection to remove unnecessary tweets and then analyzed the tweets sentimentally by breaking each tweet into several sub-tweets. After that, it calculated sentiment polarity and, to predict the election outcome, used positive tweets associated with each candidate, and then used mean absolute error (MAE) in order to measure the performance of the prediction and make the claim that this Twitter-based prediction was 0.61% better than the same type of surveys conducted traditionally. To forecast a Swedish election outcome, other than sentiment analysis a link structure was analyzed using Twitter involving politicians' conversations. For this purpose, this used a link-prediction algorithm and showed that account popularity known by structural links has more similarities with outcomes of the vote.

A methodology was created to test Brazilian municipal elections in 6 cities. In this methodology, sentiment analysis was taken into consideration along with a stratified sample of users in order to compare the characteristics of the findings with actual voters.

To show that Twitter trends play an important role in electoral sentiments, the authors collected hashtag-based tweets covering the candidates of the Indian elections in 2014. They did not include neutral tweets for the analysis because they found these kinds of tweets problematic for sentiment analyses that are in favor of more than one party. Two lexicons were combined for the sentiment analysis of tweets. This bipolar lexicon was best in the case of the analysis of two parties, but for the classification of multiple parties they created variables and their approach was not sufficiently state-of-the-art to calculate sentiment scores when more parties were involved in the analysis. They reported the highest accuracy of Naïve Bayes with a 65.2% rate, which was 5.1% more than the SVM accuracy rate.

These kinds of analyzers are also used in other domains like health, disease, and personality prediction. To the best of our knowledge, this project's aim is to attempt to cater to tweets for sentiment analysis. In addition, our results from analyzers are hence validated by statistical machine-learning classifiers like Naïve Bayes.

Purpose

Social networks have been developed as a great point for its users to communicate with their interested friends and share their opinions, photos, and videos reflecting their moods, feelings, and sentiments. This creates an opportunity to analyze social network data for users' feelings and sentiments to investigate their moods and attitudes when they are communicating via these online tools.

Methods

Although the diagnosis of depression using social networks data has picked an established position globally, there are several dimensions that are yet to be detected. In this study, we aim to perform a depression analysis on Facebook data collected from an online public source. To investigate the effect of depression detection, we propose the machine learning technique as an efficient and scalable method.

Results

We report an implementation of the proposed method. We have evaluated the efficiency of our proposed method using a set of various psycholinguistic features. We show that our proposed method can significantly improve the accuracy and classification error rate. In addition, the result shows that in different experiments Decision Tree (DT) gives the highest accuracy of other ML approaches to finding the depression.

2.2. METHODOLOGY

The workflow in this paper is divided into three phases

i. Pre-processing
ii. Training
iii. Testing

2.2.1. Pre-processing

The majority of the tweets usually can be divided into 3 parts, not specifically in the same order.

The first part contains the people to whom the tweet is intended or who is it referencing. This is usually denoted by “@”. ex: @Hickman.

The second part contains the actual message. This is what the user actually wants to convey.

The third part contains the hashtag denoted by “#”. This is usually to categorize the tweets. This can also be used by others to find tweets related to the particular content. ex: #SupportWHO.

The tweets are pre-processed to filter the first part and third part since they do not hold very less to zero significance in sentiment analysis, later, the remaining message part of the tweet is pre-processed further for obtaining useful keywords which account for much significance in identifying the emotions.

Emoji Extraction: Since Twitter users express their feelings along with emoticons, emoticons play a vital role in identifying the sentiment of the tweet.

Hyperlink Removal: Hyperlinks can be considered as a noise in the tweets whose presence degrades the quality of the data, thus, they are removed.

Slang substitution: The efficiency of the model can be increased with the substitution of full forms of abbreviations like LOL, BRB, etc., which provides more keywords for the model.

Timestamp removal: Timestamps can be in various forms like 10:30 AM, 10:30:22, etc., identifying and removing them is an important task.

Digits removal: Even digits do not hold much significance; thus they need to be removed.

Symbols removal: Unwanted symbols which do not form any meaningful emoticons need to be removed to increase the quality of the data.

Spelling correction: Most of the tweets contain a lot of misspelled words, hence to create an efficient model spelling correction is very important. Spelling correction is divided into 2 processes

1. Shortening
2. Correction

- ➢ **a. Shortening:**
  Sometimes, to emphasize feelings Twitter users repeat characters, which results in the misspelling of the word, these words are shortened by reducing the repetition of every character to a maximum of 2 letters. Ex: Haaaappppyyyyy is reduced to Haappyy

- ➢ **b. Correction:**
  This process substitutes correct spellings for misspelled words. It is not only effective in correcting misspelled words by users but also to correct words that were obtained in the shortening process. Thus, Haappyy is substituted with Happy.

Proper nouns removal: Proper nouns are the name of specific people, places, things, or ideas like Shobha, Kanchi, Pacific Ocean, etc., These are removed since the names do not contribute much to the sentiment analysis.

Lemmatization: Lemmatization is the process of mapping words to their lemma, i.e., the root word based on its meaning. It is better than stemming since stemming simply removes the suffixes of the words. Ex: The word better is substituted with the word good after lemmatization thus process of extracting keywords becomes efficient

Stop words removal: Stop words are the most commonly used words that are intended to be ignored does not increase efficiency but have the potential to decrease it. Words like “is”, “are”, “want” etc., are considered stop words. Stop words are not universal, the context decides the stop words thus identifying a set of effective stop words is an important task.

2.2.2. Training

The training process has two tasks

1. Creation of a bag of words model
2. Creation of predictive model

1. **Creation of bag of words model:**

The bag-of-words method is a simplifying representation used in natural language processing and information retrieval. In this model, a text like tweets is represented as the bag of its words, disregarding grammar and even word order but keeping multiplicity. The bag-of-words model is commonly used in methods of document classification

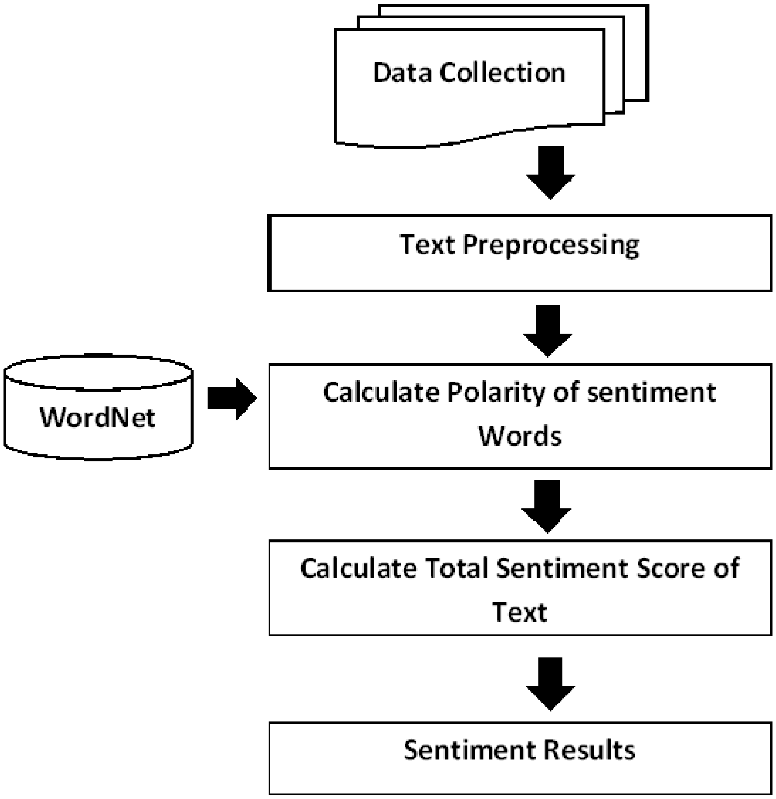
2. **The Bag-of-words model:**

The Bag-of-words model is mainly used as a tool for feature generation. After transforming the text into a "bag of words", we can calculate various measures to characterize the text. The most common type of characteristic or feature calculated from the Bag-of-words model is term frequency, namely, the number of times a term appears in the text. b. Creation of Predictive model Two Predictive models are created using naïve Bayes and logistic regression which were discussed afore. The data file will have two columns with one column representing whether the tweet is positive or negative and the other will contain the actual tweet. The column with tweets is pre-processed. The pre-processed data is used to create a bag words model which will be used as training data to build the predictive models.

2.2.3. Testing

In the testing phase, the 30% of data that was split randomly from the dataset is tested on the predictive model. The test data is pre-processed and classified as either positive or negative.

Figure 3: Sentimental Analysis Methodology


Data Collection
Text Preprocessing
WordNet
Calculate Polarity of sentiment Words
Calculate Total Sentiment Score of Text
Sentiment Results

2.3 Machine learning techniques in sentiment analysis

Machine learning is an application of AI that enables systems to learn and improve from experience without being explicitly programmed. Machine learning focuses on developing computer programs that can access data and use it to learn for themselves.

Similar to how the human brain gains knowledge and understanding, machine learning relies on input, such as training data or knowledge graphs, to understand entities, domains and the connections between them. With entities defined, deep learning can begin.

The machine learning process begins with observations or data, such as examples, direct experience, or instruction. It looks for patterns in data so it can later make inferences based on the examples provided. The primary aim of ML is to allow computers to learn autonomously without human intervention or assistance and adjust actions accordingly.

Machine learning techniques in sentiment analysis:

Machine learning techniques in the recent era are very useful to make automating classification, clustering, and predictions. Machine learning techniques in most cases have data sets for training and data sets for testing. With the help of training data sets, the system learns how to classify the test data sets, by analyzing its classification one can make future decisions.

2.3.1 Supervised Learning

Supervised learning gets training from existing labeled data and is used for the classification of data. In supervised learning algorithms map the function’s input to respective outputs. If the outputs belong to a particular class, it is known as classification otherwise it will be called a regression problem.

2.3.2 Unsupervised learning

Unsupervised learning doesn’t have labeled data sets for training for clustering. In unsupervised learning, the training task is done through the inputs directly. Structure and relation between the inputs are automatically identified by the learning algorithm. It is appropriate for the clustering of inputs to put them into the appropriate clusters.

2.3.3 Semi-supervised learning

Semi-Supervised learning uses both labeled and unlabeled data for learning. Semi-supervised learning is the branch of supervised learning techniques. As mentioned these techniques shall use labeled and unlabeled data. In these techniques minimum, labeled data is used as training data sets. Hence it overcome the problem of preparing the large training data sets with labels that are required in the case of supervised learning techniques

2.3.4 Reinforcement learning

Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation. Reinforcement learning differs from supervised learning in the way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience. Semi-supervised learning takes a middle ground. It uses a small amount of labeled data bolstering a larger set of unlabeled data. And reinforcement learning trains an algorithm with a reward system, providing feedback when an artificial intelligence agent performs the best action in a particular situation.

2.4. Machine Learning Algorithms Specification

Classifiers SVM, Naive Bayes (NB), and Decision Tree (DT) are some of the widely used algorithms in natural language processing tasks. Of these, the SVM-linear classifier demonstrates the best performance. As there is no one algorithm suited for all tasks, researchers tend to try various algorithms and enhance them for the problem of their interest.

2.4.1. Naive Bayes

Naive Bayes is based on the “Bayes’ theorem” in probability. As a requirement of this theorem, Naive Bayes can be applied only if the features are independent of each other.

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$

It is a prediction model that breaks the posterior possibilities of each class and the possible circumstances of the class for each feature. It is generally used in machine learning owing to its ability to efficiently merge the evidence from several features. Often, we know how frequently some evidence is observed, given a known outcome. The knowledge that certain evidence is observed provides us with a conclusion.

Although the Naive Bayes classifier is considered the most straightforward method in the machine learning field, it is still competitive with SVM

2.4.2. Decision Tree

The decision Tree classifies instances by sorting them based on the feature values. Each node in a Decision Tree represents a feature, and each division represents a value that the node can undertake.

A Decision Tree can structure complicated nonlinear decision borders Decision Tree. The extensive trees will be structured, and then excluded to reduce the cost-complexity criterion. The resulting tree would be easily explicable and can provide perception into the data structure which is claimed to be the main advantage of tree algorithms. Decision Trees simply pose a series of carefully constructed questions to classify a task that makes them straightforward in nature, for which they are extensively employed within the machine learning field.

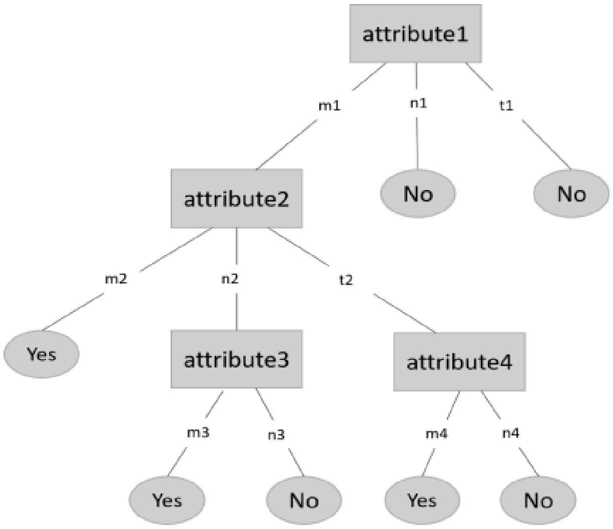


Figure 1: DT example

2.4.3. SVM

SVM is a supervised learning model that underlines two different classes in a high-dimensional space. It can adjust several features while balancing the excellent performance, to reduce the possibility of overfitting. SVM is famous for its powerful capability, specifically when working on real-world data, which includes a decisive theoretical basis and its insensitivity to high-dimensional data. SVM is a type of algorithm with a set of labeled training examples for a binary classification problem. The training algorithm in SVM creates a potential hyperplane, which divides the cases into two classes. It escalates the distance between the divided hyperplane and the training examples closest to the hyperplane. SVM can provide the prediction and determine which side of the hyperplane an object inclines.

2.5. Evaluation Measures:

Critique and cross-validation of the feasibility of this automated prediction will be conducted through standard accuracy (Acc), precision (P), recall (R), and F1 scores, as well as confusion matrix (CM), and recipient operating classification curves (ROCs), which are defined as follows:

2.5.1. Accuracy:

Accuracy is the simplest and most used measure to evaluate a classifier. It is defined as the degree of right predictions of a model (or contrarily, the percentage of misclassification errors).

$$Acc = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negative}$$

2.5.2. Precision:

Precision is defined as the fraction of correctly classified positives to the total predicted positives. Under our condition, it aims to find how many of the users identified as depressed are actually depressed.

$$P = \frac{true\ positives}{true\ postives + false\ postives}$$

2.5.3. Recall:

Recall is defined as the fraction of correctly classified positives to total positives. Within our situation, it aims to determine that of all depressed users, how many are properly detected.

$$R = \frac{true\ positives}{true\ positives + false\ negatives}$$

The trade-off between recall (false negatives) and precision (false positive) is compromised by considering the F1 measure:

2.5.4. F1 Score (F-measure):

F1 Score is the harmonic mean of precision and recall; it weighs each metric evenly, and therefore, is commonly utilized as a classification evaluation metric. Hence, it is important to achieve both high recall and high precision.

$$F1 = \frac{2 * P * R}{P + R}$$

CHAPTER 3: SYSTEM ANALYSIS AND DESIGN

CHAPTER 3: SYSTEM ANALYSIS AND DESIGN

3.1. Requirement Specification

- Software Requirements: Operating system - Windows 11, Windows 10, or Linux including Ubuntu, Kali, etc
- System architecture: Windows- 64-bit x86, 32-bit x86; macOS- 64-bit x86 & Apple M1 (ARM64); Linux- 64-bit x86, 64-bit aarch64, etc.
- Programming Language - Python 3.6.
- Platform used - Jupyter, Tensorboard.
- Hardware Requirements. Intel Core i7 Octa-Core Edition. 96 GB DDR4 RAM. Nvidia 1080 Ti

3.2 Dataset Information

We generated a new dataset, combining part of the Sentiment140 (8,000 positive tweets), and another one for depressive tweets (2,314 tweets), with a total of 10,314 tweets. You can find this dataset in our repo. **Sentiment140**.

The Sentiment140 dataset contains 1,600,000 tweets extracted using the Twitter API. The tweets have been annotated (0 = negative, 2 = neutral, 4 = positive) and they can be used to detect sentiment. It contains the following 6 fields:

target: the polarity of the tweet (0 = negative, 2 = neutral, 4 = positive)
ids: The id of the tweet (2087)
date: the date of the tweet (Sat May 16 23:58:44 UTC 2009)
flag: The query (lyx). If there is no query, then this value is NO_QUERY.
user: the user that tweeted (robotickilldozr)
text: the text of the tweet (Lyx is cool)

You can find the dataset here: **https://www.kaggle.com/kazanova/sentiment140.**
For our experiment, **we just took a sample of 8,000 tweets with a polarity of 4, the positive ones.**

3.3. Library Specification

PANDAS:

Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. Pandas allow us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets and make them readable and relevant.

NUMPY:

NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices. The array object in NumPy is called ND array, it provides a lot of supporting functions that make working with ND array very easy.

MATPLOTLIB:

Matplotlib is a low-level graph plotting library in python that serves as a visualization utility. Matplotlib was created by John D. Hunter. Matplotlib is open source, and we can use it freely. It is mostly written in python, a few segments are written in C, Objective-C, and JavaScript for Platform compatibility.

SEABORN:

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and is also closely integrated to the data structures from pandas.

RE:

Python has a built-in package called re, which can be used to work with Regular Expressions. Regex, or Regular Expression, is a sequence of characters that forms a search pattern. Regex can be used to check if a string contains the specified search pattern.

NTLK:

NLTK (Natural Language Toolkit) Library is a suite that contains libraries and programs for statistical language processing. It is one of the most powerful NLP libraries, which contains packages to make machines understand human language and reply to it with an appropriate response.

FROM NTLK.STEM IMPORT PORTER STEMMER:

Martin Porter invented the Porter Stemmer or Porter algorithm in 1980. Five steps of word reduction are used in the method, each with its own set of mapping rules. Porter Stemmer is the original stemmer and is renowned for its ease of use and rapidity. Frequently, the resultant stem is a shorter word with the same root meaning.

WORDCLOUD:

It is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. Word clouds are widely used for analyzing data from social network websites.

COUNT VECTORIZER:

Count Vectorizer is a great tool provided by the scikit-learn library in Python. It is used to transform a given text into a vector-based on the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple such texts, and we wish to convert each word in each text into vectors (for use in further text analysis).

TRAIN SPLIT TEST:

It is a function in Sklearn model selection for splitting data arrays into two subsets: for training data and for testing data. With this function, you don't need to divide the dataset manually. By default, Sklearn train_test_split will make random partitions for the two subsets. However, you can also specify a random state for the operation.

FROM NTLK CORPUS IMPORT STOPWORDS:

A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. We would not want these words to take up space in our database or take up valuable processing.

FROM NTLK.STEM IMPORT WORDNETLEMMATIZER:

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. Lemmatization is similar to stemming but it brings context to the words. So it links words with similar meanings to one word.

3.4. SYSTEM ARCHITECTURE

3.4.1. Supervised learning:

In supervised learning, models are trained using labeled datasets, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.

Steps Involved in Supervised Learning:

- First, determine the type of training dataset
- Collect/Gather the labeled training data.
- Split the training dataset into a training dataset, test dataset, and validation dataset.
- Determine the input features of the training dataset, which should have enough knowledge so that the model can accurately predict the output.
- Determine the suitable algorithm for the model, such as support vector machine, decision tree, etc.
- Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of training datasets.
- Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

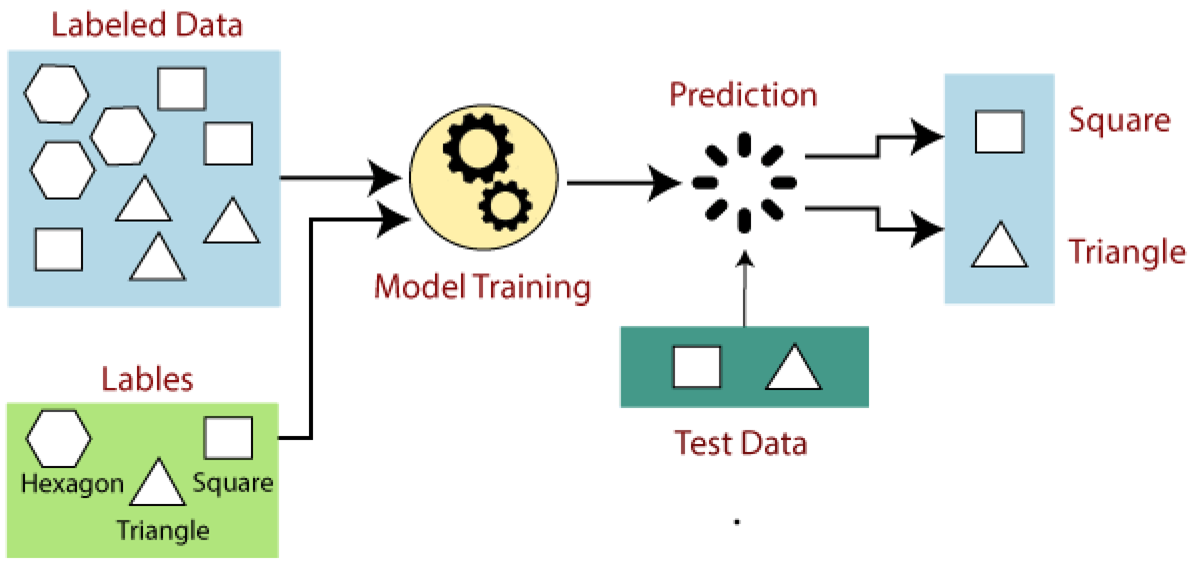


Figure 4: Working of Supervised Learning

3.4.2 Steps of Naïve Bayes Classifier (With Training Set Scheme):

Step 1: Create data files for the classifier.
(1.1) Create a file of tweets with the sentiment of each sentiment analyzer (test set).
(1.2) Create a file of negative and positive labeled tweets of each sentiment analyzer (training set).
(1.3) Convert all csv files to arff format files for Weka compatibility. Math. Comput. Appl. 2018, 23, 11 9 of 15

Step 2: Build Naïve Bayes classifier model on Weka.
(2.1) Create a model of each analyzer by providing the training set file.

Step 3: Execution of model on the test set.
(3.1) Load the test set file.
(3.2) Apply the StringToWordVector filter with following parameters: IDFTransform: true, TFTransform: true, stemmer: SnowballStemmer, stopwordsHandler: rainbow, tokenizer: WordTokenizer.
(3.3) Execute the model on the test set.
(3.4) Save results in the output file.

3.4.3 NAIVE BAYES CLASSIFIER ALGORITHM:

Naïve Bayes algorithm is a supervised learning algorithm, which is based on the Bayes theorem and is used for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset. Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of the Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles. The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as

Naïve: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identifying that it is an apple without depending on each other.

Bayes: It is called Bayes because it depends on the principle of Bayes' Theorem.

Advantages of Naïve Bayes Classifier:

- Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
- It can be used for Binary as well as Multi-class Classifications.
- It performs well in Multi-class predictions as compared to the other Algorithms.
- It is the most popular choice for text classification problems.

Disadvantages of Naïve Bayes Classifier:

- Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

Applications of Naïve Bayes Classifier:

- It is used for Credit Scoring.
- It is used in medical data classification.
- It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
- It is used in Text classification such as Spam filtering and Sentiment analysis.

Steps to implement:

- Data Pre-processing step
- Fitting Naive Bayes to the Training set
- Predicting the test result
- Test accuracy of the result(Creation of Confusion matrix)
- Visualizing the test set result.

3.4.4. LOGISTIC REGRESSION:

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1). Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:

Assumptions for Logistic Regression:

1) The dependent variable must be categorical in nature.
2) The independent variable should not have multi-collinearity.

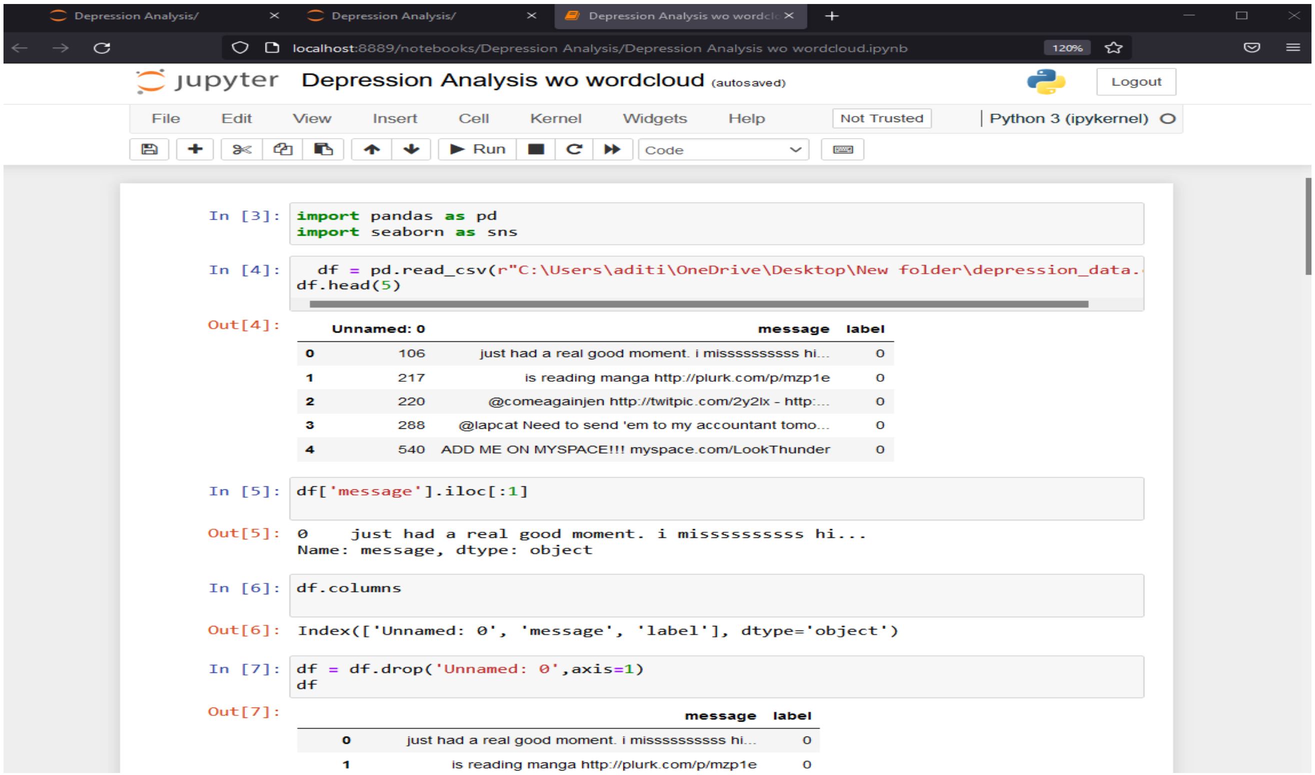
Steps in Logistic Regression:

To implement the Logistic Regression using Python, we will use the same steps as we have done in previous topics of Regression. Below are the steps:

- Data Pre-processing steps
- Fitting Logistic Regression to the Training set
- Predicting the test result
- Test accuracy of the result(Creation of Confusion matrix)
- Visualizing the test set result.

3.4.5. JUPYTER NOTEBOOK

The Jupyter Notebook is an open-source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is convenient for the initial development of code. It allows you to segment your code (and re-run segments of your code) while storing the values of variables from segments you've already run.


Depression Analysis/
Depression Analysis/
Depression Analysis wo wordcl
localhost:8889/notebooks/Depression Analysis/Depression Analysis wo wordcloud.ipynb
120%
jupyter
Depression Analysis wo wordcloud (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code

```
In [3]: import pandas as pd
        import seaborn as sns

In [4]:   df = pd.read_csv(r"C:\Users\aditi\OneDrive\Desktop\New folder\depression_data.
        df.head(5)

Out[4]:
```

	Unnamed: 0	message	label
0	106	just had a real good moment. i missssssssss hi...	0
1	217	is reading manga http://plurk.com/p/mzp1e	0
2	220	@comeagainjen http://twitpic.com/2y2lx - http:...	0
3	288	@lapcat Need to send 'em to my accountant tomo...	0
4	540	ADD ME ON MYSPACE!!! myspace.com/LookThunder	0

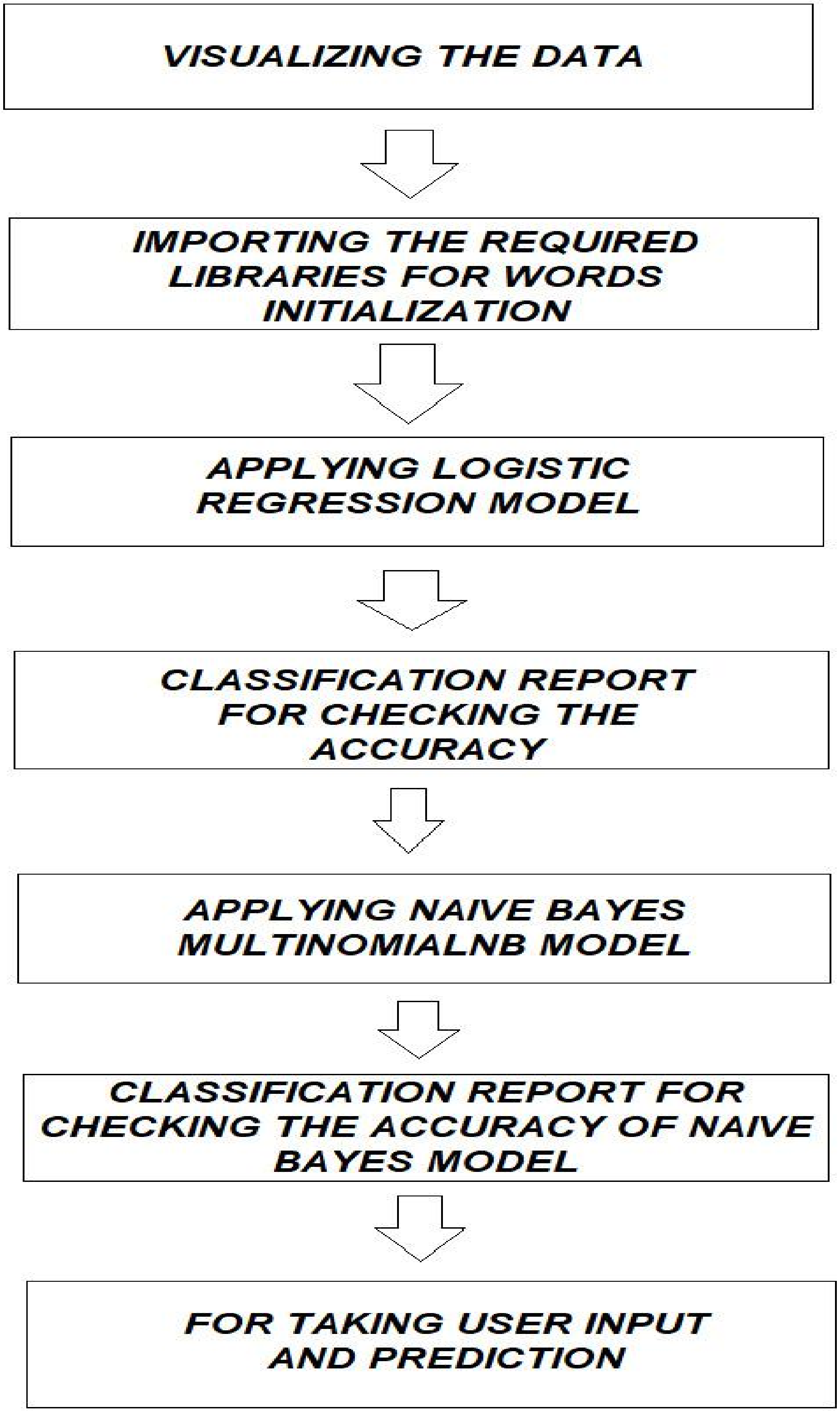
```
In [5]: df['message'].iloc[:1]

Out[5]: 0     just had a real good moment. i missssssssss hi...
        Name: message, dtype: object

In [6]: df.columns

Out[6]: Index(['Unnamed: 0', 'message', 'label'], dtype='object')

In [7]: df = df.drop('Unnamed: 0',axis=1)
        df

Out[7]:
```

	message	label
0	just had a real good moment. i missssssssss hi...	0
1	is reading manga http://plurk.com/p/mzp1e	0

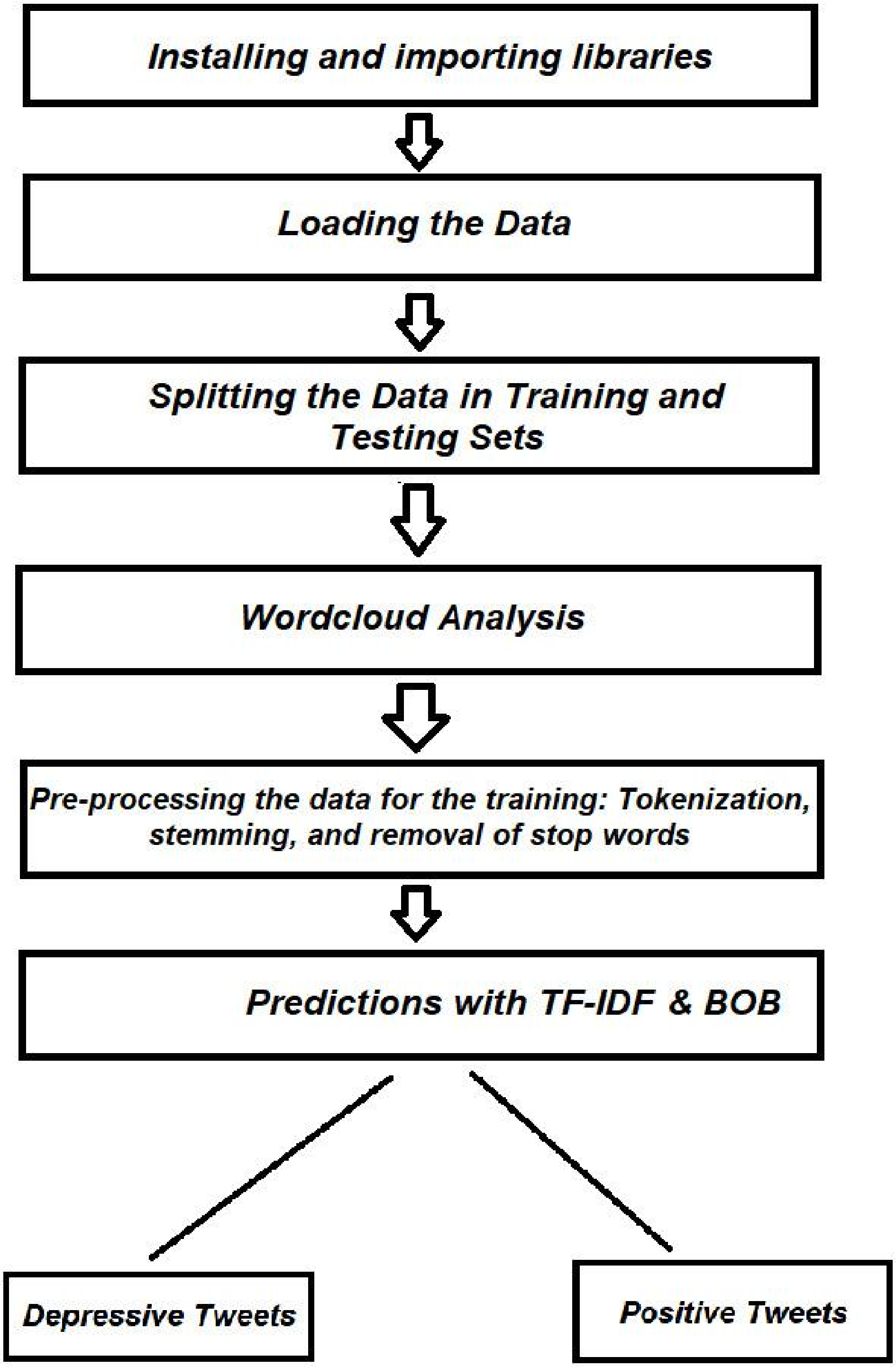
Figure 5: Interface of Jupyter Notebook

3.5. FLOWCHARTS OF THE PROGRAMS

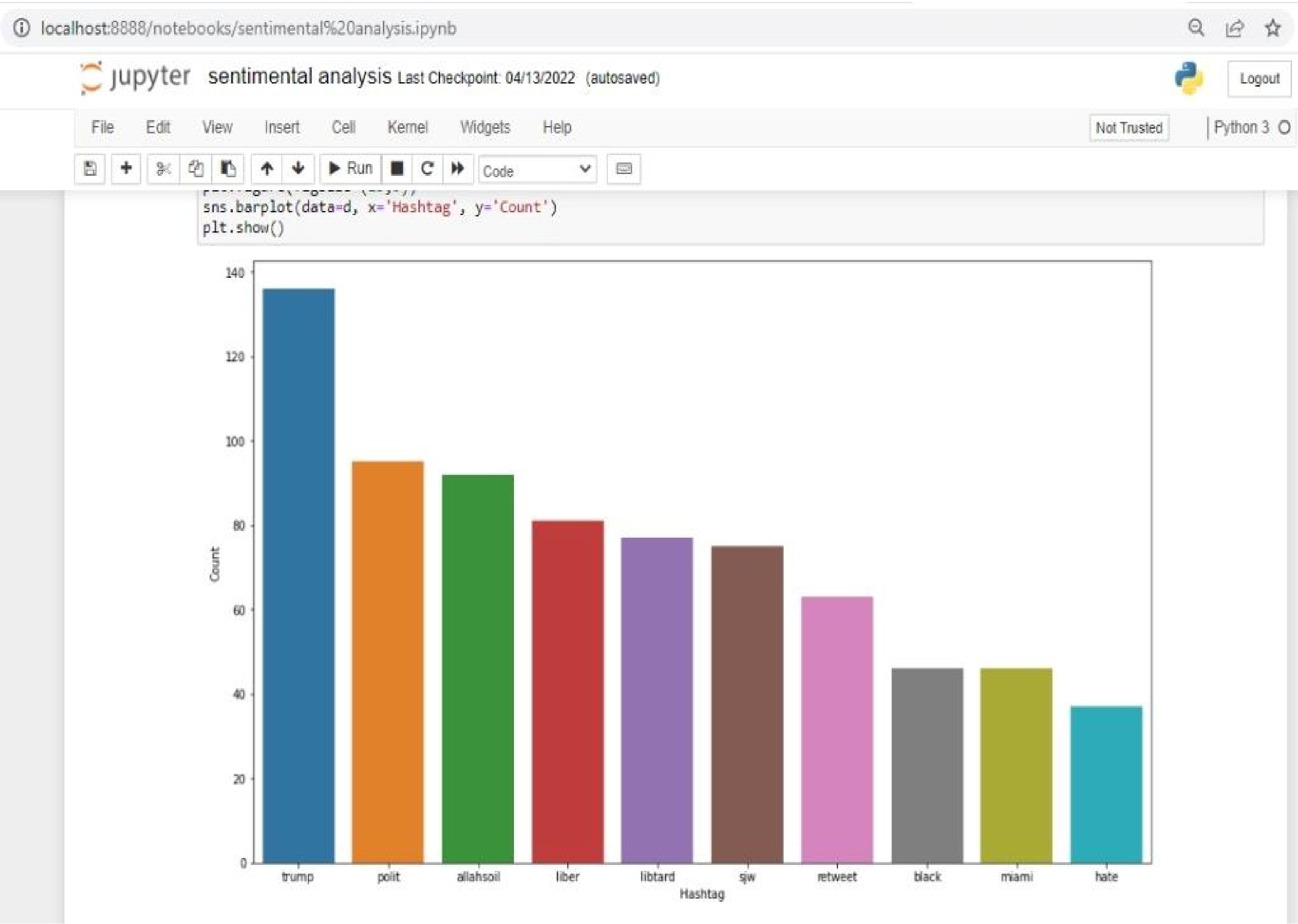
Flowchart 3.5.1 for Model 1 - Detecting Depression or negativity in the input message from a user


VISUALIZING THE DATA
IMPORTING THE REQUIRED LIBRARIES FOR WORDS INITIALIZATION
APPLYING LOGISTIC REGRESSION MODEL
CLASSIFICATION REPORT FOR CHECKING THE ACCURACY
APPLYING NAIVE BAYES MULTINOMIALNB MODEL
CLASSIFICATION REPORT FOR CHECKING THE ACCURACY OF NAIVE BAYES MODEL
FOR TAKING USER INPUT AND PREDICTION

Flowchart 3.5.2 for Model 2 : Training a model to segregate into categories (positive, negative, neutral) the dataset messages on the basis of the sentiment of the tweet or text

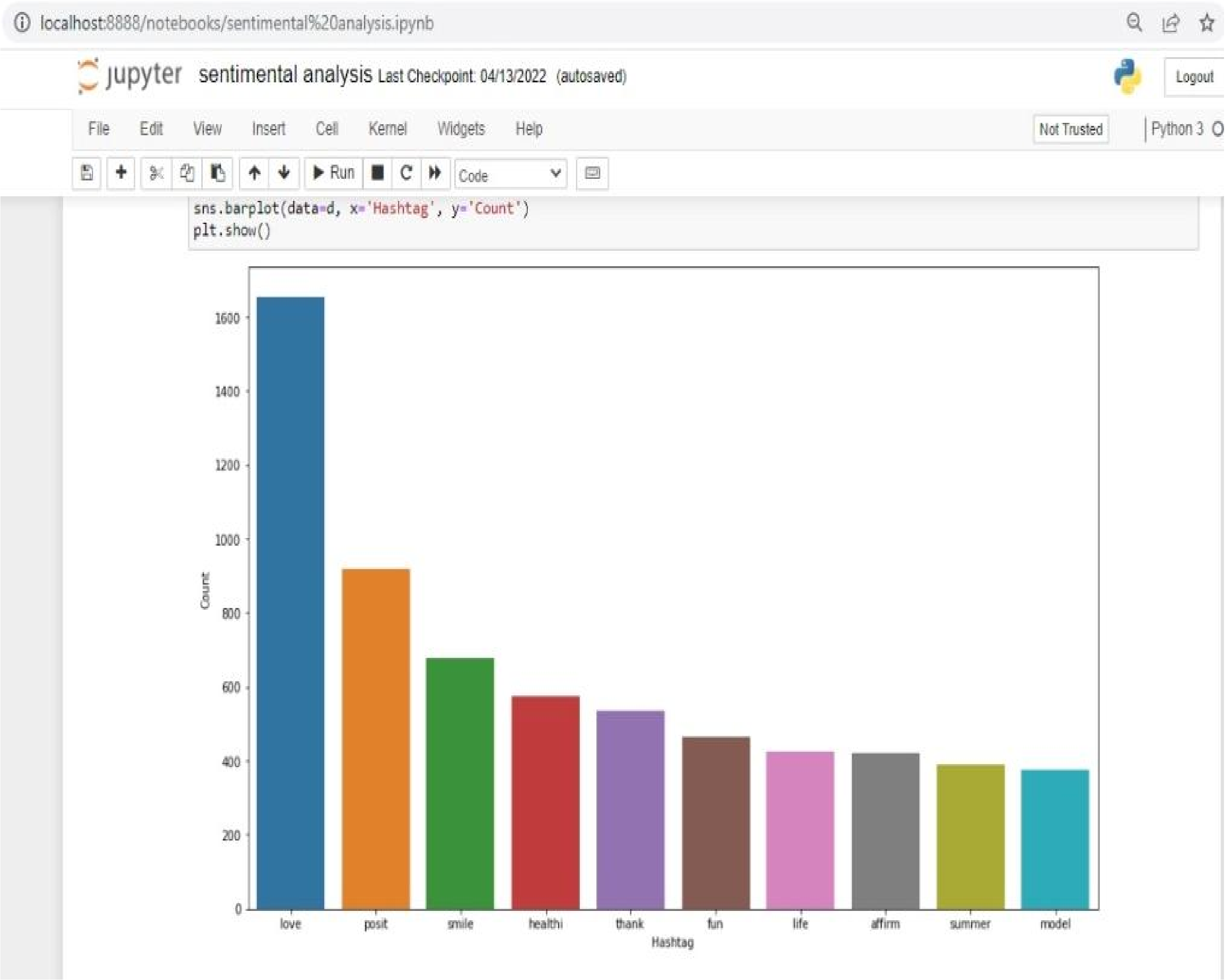

Installing and importing libraries
Loading the Data
Splitting the Data in Training and Testing Sets
Wordcloud Analysis
Pre-processing the data for the training: Tokenization, stemming, and removal of stop words
Predictions with TF-IDF & BOB
Depressive Tweets
Positive Tweets

3,6. GRAPH:



```
sns.barplot(data=d, x='Hashtag', y='Count')
plt.show()
```

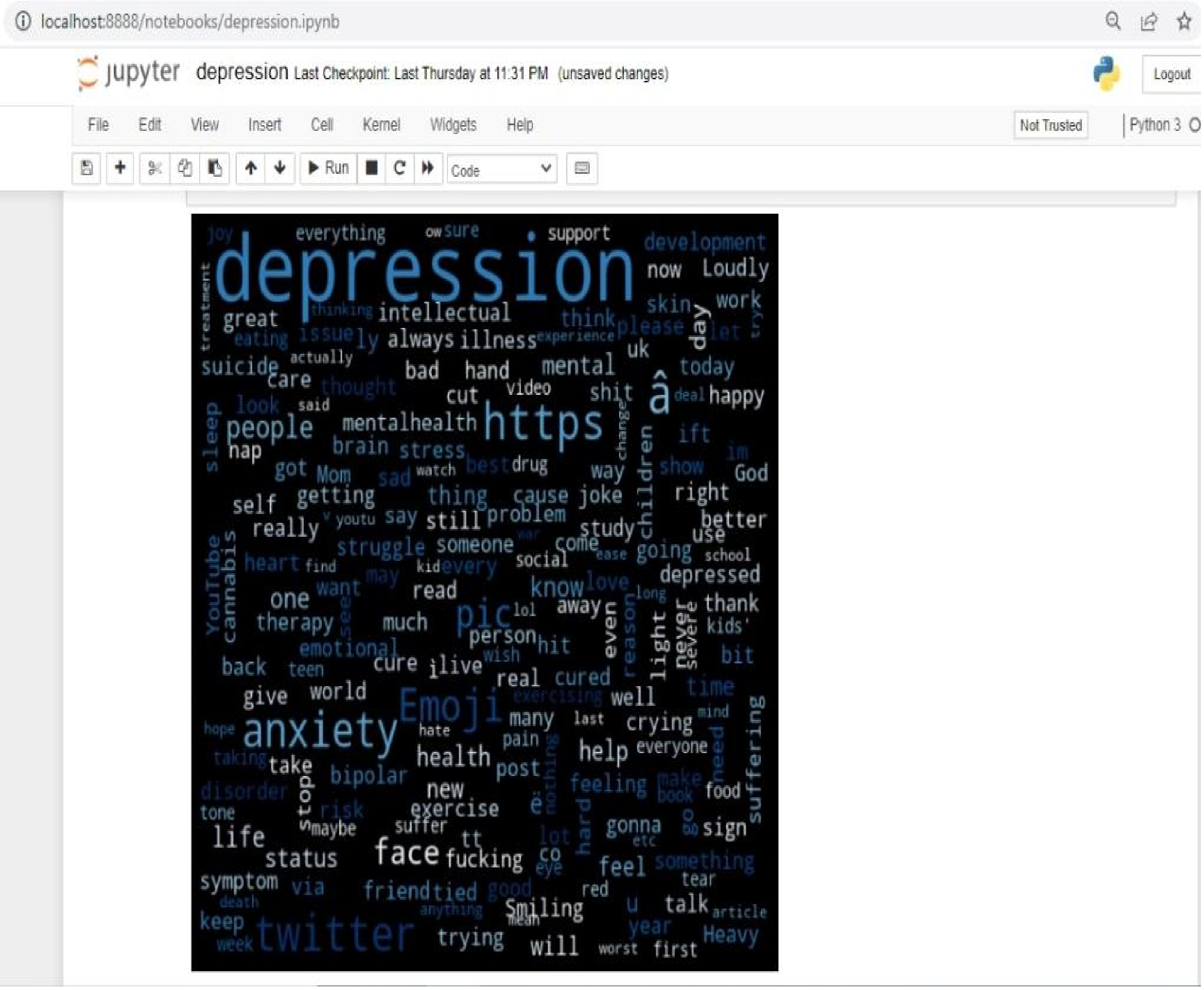
Graph 10.1.: Graph for negative or depressive tweets (MODEL 1)



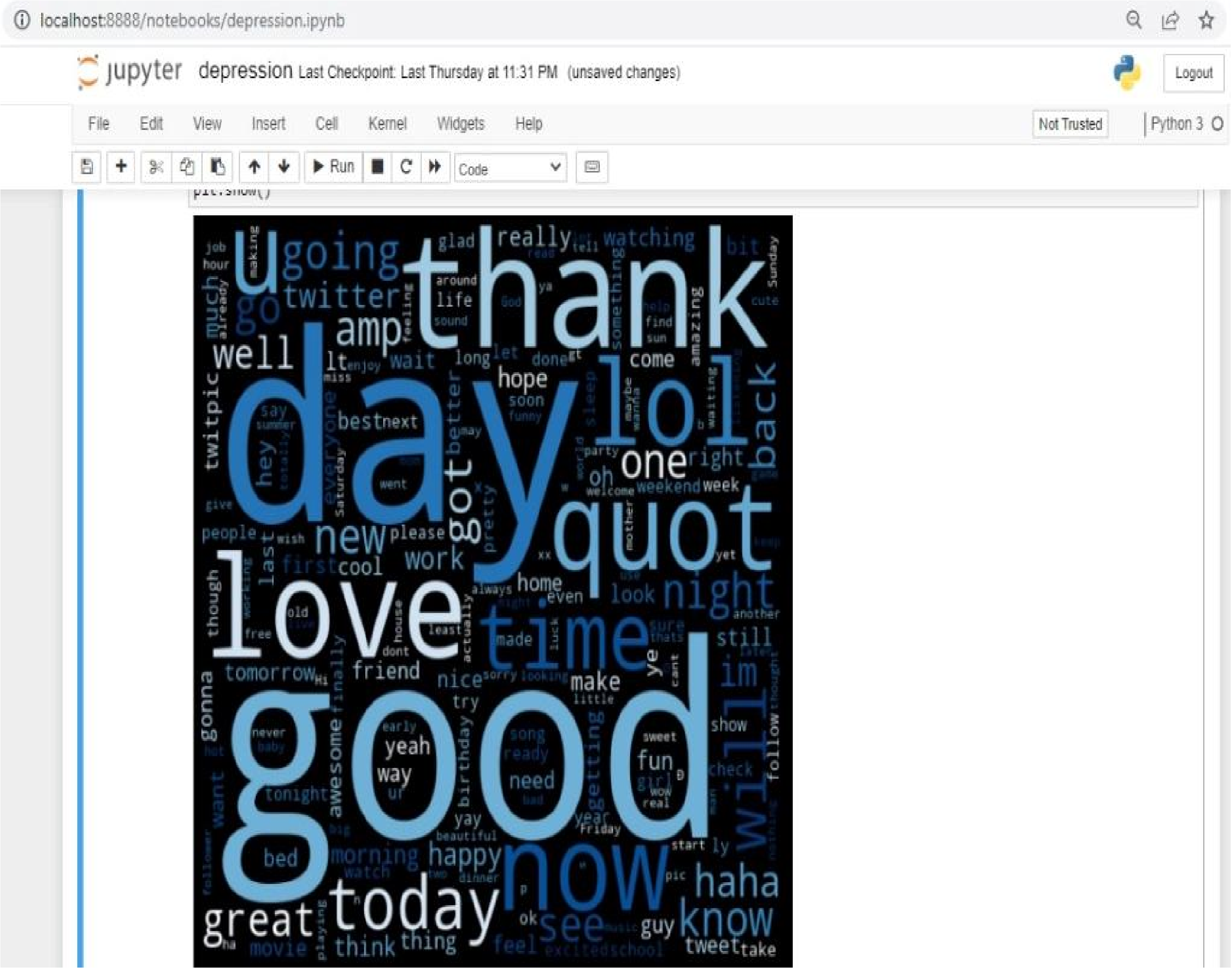
```
sns.barplot(data=d, x='Hashtag', y='Count')
plt.show()
```

Graph 10.2: Graph for positive or non-depressive tweets (MODEL 2)

3.7. WORDCLOUD:



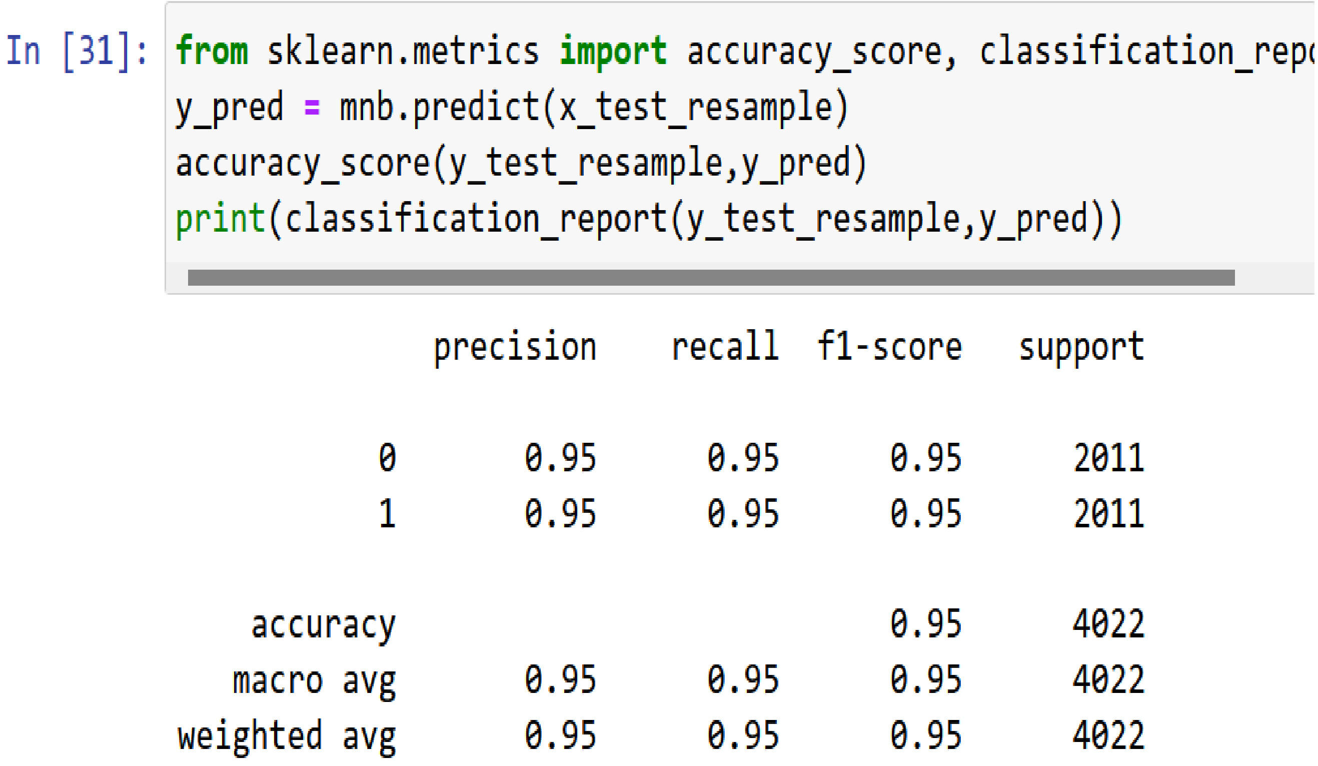
Wordcloud 1: Negative/Depressive messages wordcloud



```
plt.show()
```

Wordcloud 2: Positive/Non-Depressive messages wordcloud

3.8. PRECISION AND ACCURACY OF DEPRESSION DETECTION MODEL (MODEL1):



```
In [31]: from sklearn.metrics import accuracy_score, classification_rep
         y_pred = mnb.predict(x_test_resample)
         accuracy_score(y_test_resample,y_pred)
         print(classification_report(y_test_resample,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.95      0.95      0.95      2011
           1       0.95      0.95      0.95      2011

    accuracy                           0.95      4022
   macro avg       0.95      0.95      0.95      4022
weighted avg       0.95      0.95      0.95      4022
```

Figure 6: A precision of 0.95 and an accuracy of 95% are achieved

CHAPTER 4: IMPLEMENTATION AND RESULTS

CHAPTER 4: IMPLEMENTATION AND RESULTS

4.1. Implemented Code for Model 1: Detecting Depression or negativity in the input message from a user after training and testing the model using a dataset, while generating graph plots, wordcloud of data, and an accuracy and precision table

4.2. Implemented Code for Model 2: Training a model to segregate into categories (positive, negative, neutral) the dataset messages on the basis of the sentiment of the tweet or text and then illustrating the same by the means of graph plots and wordclouds

CHAPTER 5: CONCLUSION AND FUTURE ENHANCEMENTS

CHAPTER 5: CONCLUSION AND FUTURE ENHANCEMENTS

5.1. Summary of work done

We have worked on modeling two models with different approaches in order to find the algorithm which provides maximum precision and accuracy. The two models are similar in working but one focuses on Detecting Depression or negativity in the input message from a user after training and testing the model using a dataset. This model gives an accuracy of 95-96%. The other focuses on mainly the sentimental analysis of texts/tweets by segregating them into categories of positive, negative, or neutral, ultimately giving us data that can be put to use to enhance further models already in use. The precision of this model is 0.9456 and the accuracy of this model is 94%

Present scope of Sentimental Analysis:

Identifying and Predicting Market Trends
It enables you to analyze large amounts of market research data in order to spot emerging trends and better understand consumer buying habits. This type of practice can help you navigate the complicated world of stock market trading and make decisions based on market sentiment.

Keeping an eye on the brand's image
Sentiment analysis is frequently used to investigate user perceptions of a product or topic. You can also use it to conduct a product analysis and provide all relevant data to the development teams.
Examining public opinion polls and political polls

To predict the outcome of an election
To predict the outcome of an election, anyone can use sentiment analysis to compile and analyze large amounts of text data, such as news, social media, opinions, and suggestions. It takes into account how the general public feels about both candidates.

Data from customer feedback is being analyzed.
Data from customer feedback can be used to identify areas for improvement. Sentiment analysis can help you extract value and insights from customer feedback data, as well as develop effective customer satisfaction strategies.

Observing and analyzing conversations on social media

Conversations on social media are a gold mine of information. Look at conversations about your brand on social media to see what your customers are saying with sentiment analysis; this can help any company plan its future strategies much more effectively.

Employee Turnover Reduction
Analyze large amounts of employee feedback data to determine employee satisfaction levels. The insights are used by the sentiment analysis tool to boost morale and productivity while also informing you of how your employees are feeling.

5.2. Proposal/scope of future enhancement

Applying the sentimental analysis to extract the sentiment became an important work for many organizations and even individuals. Sentiment analysis is an emerging field in the decision-making process and is developing fast. The goal of our project is to analyze the sentiments on a topic that are extracted from Twitter and determine the nature (positive/negative/neutral) of the defined topics. The development of techniques for document-level sentiment analysis is one of the significant components of this area. Recently, people have started expressing their opinions on the Web which increases the need of analyzing the opinionated online content for various real-world applications. A lot of research is present in the literature to detect the sentiment of the text. Still, there is a huge scope for improvement of these existing sentiment analysis models. Existing sentiment analysis models can be improved further with more semantic and commonsense knowledge.

The most important bit for sentiment analysis in the future has less to do with improving the accuracy of the algorithms but instead lies in the area of determining where you can correlate sentiment with behavior.

The future of sentiment analysis is going to continue to dig deeper, far past the surface of the number of likes, comments, and shares, and aim to reach, and truly understand, the significance of social media interactions and what they tell us about the consumers behind the screens. This forecast also predicts broader applications for sentiment analysis – brands will continue to leverage this tool, but so will individuals in the public eye, governments, nonprofits, education centers, and many other organizations.

It will have a lot to do with social forums/platforms where people express a free opinion. Presently tweets are one such open medium, then if Facebook at some point chooses to make the timeline updates/status messages open to search (I think it will someday do that through a minuscule sounding update in "privacy policy") it will be a gold mine of real-time sentiments.

5.3.Limitations

Sentiment analysis tools can identify and analyze many pieces of text automatically and quickly. But computer programs have problems recognizing things like sarcasm and irony, negations, jokes, and exaggerations - the sorts of things a person would have little trouble identifying. And failing to recognize these can skew the results.

The patterns a machine learning system trained on review data has learned to recognize as evidence for predicting sentiment in one domain will generally not be useful for predicting sentiment in other domains. Even more problematically, most online review data is in English. For global organizations, successful reputation management requires monitoring media sources in many languages. In order to use sentiment analysis systems trained on English data exclusively, special steps must be taken that either involve costly translation of all relevant news articles and social media posts or complex, state-of-the-art methods that allow the trained system to transfer what it has learned from one language to another.

5.4. Conclusion

This project defines a binary classification problem as identifying whether a person is depressed, based on his tweets, messages, and social profile activity. Different machine learning algorithms are exploited and different feature datasets are explored. Many preprocessing steps are performed, including data preparation and aligning, data labeling, and feature extraction and selection. The Naive Bayes model has achieved optimal accuracy metric combinations.

This project can be considered as a step toward building a complete social media-based platform for analyzing and predicting mental and psychological issues and recommending solutions for these users. The main contribution of this study lies in exploiting a rich, diverse, and discriminating feature set that contains both tweet text and behavioral trends of different users. This study can be extended in the future by considering more ML models that are highly unlikely to over-fit the used data and find a more dependable way to measure the features' impact.

REFERENCES:

[1] VarshaSahayak, VijayaShete, ApashabiPathan, “Sentiment Analysis on Twitter Data, International Journal of Innovative Resea Advanced Engineering”, (IJIRAE) ISSN: 2349-2163, Issue 1, Volume 2, January 2015

[2] Anjali Gupta, Amita Dhankar, Surayansh Dabas, “Sentiment Analysis using Machine Learning”, SSRN Electronic Journal, 5(2):923-937, February 2018

[3] MS. Purude Vaishali Narayanrao, Dr. P. Lalitha Surya Kumari, “Analysis of Machine Learning Algorithms for Predicting Depression”, in IEEE Int. Conf. Commun., Feb 2020

[4] “Depression Detection and Analysis Using Deep Learning: Study and Comparative Analysis”, Conference: 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT 2021), July 2021

[5] Ms.Sumathi M.R., Dr. B. Poorna, “Prediction of Mental Health Problems among Children using Machine Learning Techniques “, IJACSA Int. Conf, Sept 2016

[6] Marcel Trotzek, Sven Koitka, and Christoph M. Friedrich “Utilizing Neural Networks and Lingustic Metadata for Early detection of depression Indications in Text Sequences” , IEEE TKDE Conf., Dec 2018.

[7] Shen, G., Jia, J., Nie, L., Feng, F., Zhang, C., “Depression detection via harvesting social media: A multimodal dictionary learning solution”, IJCAI 3838-3844, Aug 2017

[8] Khari, M., & Kumar, P. (2018). “Evolutionary computation-based techniques over multiple data sets: an empirical assessment”, Arabian Journal for Science and Engineering, 43(8), 3875-3885, Jan 2019

[9] Md Zia Uddin, Kim Kristoffer Dysthe, Asbjørn Følstad, Petter Bae Brandtzaeg, “ Deep learning for prediction of depressive symptoms in a large textual dataset ”, IEEE Int. Conf. Commun. Workshops (IEEE ICC 2021) , 27 August 2021

[10] Wheidima Carneiro de Melo, Eric Granger†, Abdenour Hadid “Depression detection based on Deep Distribution Learning” , IEEE ICIP 2019, Sep 2019

[11] Petra Hoffmannová "Text-Based Detection of the Risk of Depression, Frontiers in Psychology", IEEE Int. Conf. Commun. Workshops (IEEE ICC 2019), Paris, March 2019.