Depression Detection in tweets

Akshay Kumar Thota[1]

Swinburne University of Technology,Melbourne Australia
akshaykumarthota@gmail.com

1 Introduction

Depression is a mental illness that is not taken seriously.We are not taught about the seriousness of mental illness , neither at home not at school. We might know that someone might be suffering from depression and we might feel bad for them but we do not know what measure we can take in that state. We do not know what to do to get them out of depression state. Mental illness is generally seen as a taboo and not many people want to talk about it.The people who might be suffering from depression might not consider visiting doctor or even take medicine and continue living the way they live and their state keeps on getting worse.

1.1 Problem Statement

Social media platforms can be used to know about people's personal life as they are becoming integral part of their life. They share their personal feelings whether happy or sad and by analysing these tweets we can find if a person's mental state.

2 Related Work

Ahmed Husseini Orabi proposed a system for detecting depression of twitter users in 2018.

Link to their solution is https://www.aclweb.org/anthology/people/a/ahmed-husseini-orabi/

We have used Naive Bayes algorithm of supervised learning and to understand our dataset better we have shown different kinds of visualizations as well.

3 Proposed Methodology

3.1 Dataset Description

The dataset is obtained from kaggle[1] The data is collected using web scrapping.The tool used is twint which is an advanced Twitter scraping tool that is

Table 1. Details of the dataset.

Details	**Count**
Number of instances	1600000
Number of attributes	6
whether labelled or unlabelled	labelled
Type of label information	Numerical(target)

written in Python that doesn't use Twitter's API. It allows to scrape a user's followers, following, tweets and more while avoiding most API limitations.

Every dataset also comprises of data attributes. Table 2 describes attributes of data. In case of supervised learning, clearly mention which attribute(s) would be considered as the *labels*.

Table 2. Details of Data Attributes.

Data Attributes	**Explanation**
Target	It tells the sentiment of the tweet-numerical
ID	It tells the id-numerical
Date	It tells the date when tweet was made-categorical
Flag	It tells the query (lyx). When there is no query,the value is NO Query-categorical
User	It tells the name of the user who tweeted-categorical
Text	The text that tweet contains-categorical

Target -If 0 then negative, if 2 then neutral,if 4 then positive

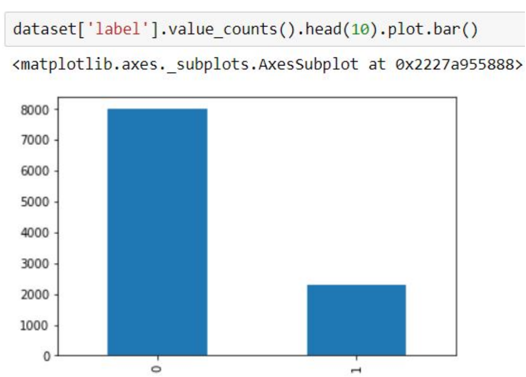

Fig. 1. The graph depicts Total number of negative and positive tweets.

[1] https://www.kaggle.com/kazanova/sentiment140

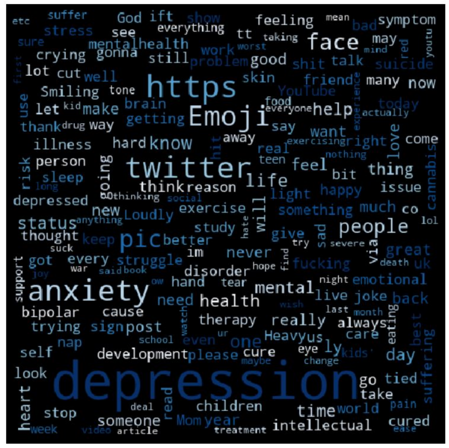

Fig. 2. The graph depicts the most common words that were present in the tweets

Fig. 3. The graph depicts most frequent used words.

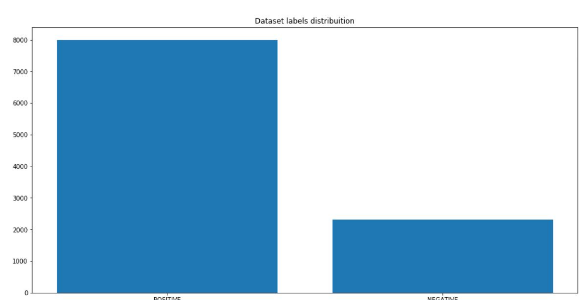

Fig. 4. The graph depicts negative frequency vs positive frequency.

3.2 Data Pre-processing

The data for any NULL values was checked.Our data do not contain any missing values.The attributes containing text was normalised.The words are split by white spaces and all are converted into lower case.Text paragraphs are broken down into smaller words or sentence also known as tokens. Stopwords are removed generally by creating a list of stopwords and filter out the list of tokens from these words. Finally, Stemming is done that reduces the words to their word root word or removes off the derivational affixes.

4 Proposed Approach

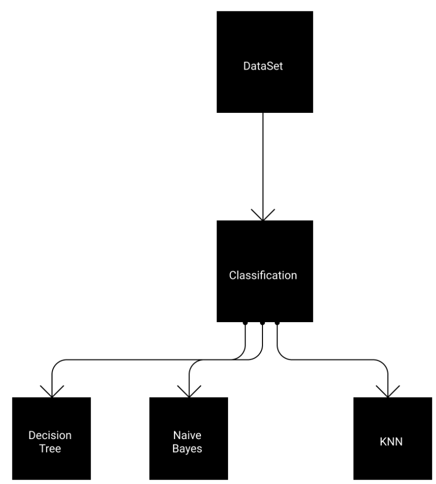

Fig. 5. The flowchart depictes about our process.

There through this flowchart we are explaining our process. After collecting dataset we decided to perform Classification Machine Learnig task.We decided upon Naive Bayes, KNN and decision tree. We have choosen Naive Bayes after using TF-IDF and bag of words the accuracy of TF-IDF was more- 0.86 so we have used this methodology

4.1 Depression Detection

Input:Tweets
Output:Depressive or not

4.1.1 Naive Bayes We have used Naive Bayes machine learning algorithm.Naive Bayes is based on Bayes Theorem that is a statistical classification technique .Naive Bayes classifier is the simplest , accurate , fast, and reliable algorithm. Naive Bayes classifiers can be used on large datasets and have high accuracy.In this algorithm the effect of a particular feature in a class is assumed to be independent of other features. The features are considered independently even though if the features are interdependent.Because of this assumption the computations are simplified, and that's why it is considered as naive. This assumption is known as class conditional independence.

$$P(h|D) = (P(D|h)P(h))/P(D) \tag{1}$$

P(h):It is known as the prior probability of h.
P(D):It is known as the prior probability.
P(h/D):This is known as posterior probability.
P(D/h):This is known as posterior probability.

4.1.2 Decision Tree Decision tree has a structure like a flowchart, here each of the internal nodes are some test on an attribute whereas each of the branch is some output of those tests and each of the leaf nodes represents a label class. For learning of the tree it is splitted into subsets. And these subsets are based on the attribute values. In the tree the instances are classified by starting from the root node and then moving towards the leaf of the tree.

4.1.3 K-Nearest Neighbor This algorithm is used to find the K nearest neighbors from the testing data.Euclidean distance is used to find the distance between the training data set and the test data. Then the k number of smallest distances are choosen and their labels are compared. The label which is present more number of times is given as label for the test data.

5 Algorithm

Naive Bayes based implementation

Algorithm 1 Naive Bayes algorithm

procedure NaiveBayes
  $tweets \leftarrow sentimentdataset$
  $totalTweets \leftarrow 8000 + 2314$
  $trainIndex \leftarrow list()$
  $testIndex \leftarrow list()$
  $randomnumber \leftarrow rand()$
  while $i \geq range(tweets.shape[0])$ **do**
    if $randomnumber \leq 0.98$ **then**
      $trainIndex \leftarrow trainIndex + 1$
    else
      $testIndex \leftarrow testIndex + 1$
    end if
  end while
  $trainData \leftarrow tweets.iloc[trainIndex]$
  $testData \leftarrow tweets.iloc[testIndex]$
  $gnb = GaussianNB()$
  $gnb.fit(trainData)$
  $ypred = gnb.predict(testData)$
  $precision = truePos/(truePos + falsePos)$
  $recall = truePos/(truePos + falseNeg)$
  $Fscore = 2 * precision * recall/(precision + recall)$
  $accuracy = (truePos + trueNeg)/(truePos + trueNeg + falsePos + falseNeg)$
end procedure

6 Data Exploration

6.1 Visualizations

Frequencies of words Table 3 depicts the frequencies of words in the negative and positive tweets.It shows what type of words are used in the text, and how many times it was used in the entire text. we are using count vectorizer to calculate the term frequencies.

Table 3. Number of positive and negative tweets.

words	negative	positive	total
to	1322	2564	3886
the	1162	2656	3818
and	1324	1503	2827
you	710	2018	2728
depression	2505	0	2505
it	675	1512	2187
my	653	1273	1926
of	959	915	1874
is	709	1071	1780
for	465	1215	1680

Top 50 tokens in negative tweets Fig.1 shows the bar graph plot for the word frequencies in negative tweets.On x-axis we have the frequencies and on y-axis the tokens, each bar represents the frequency for a particular word used in negative tweets.

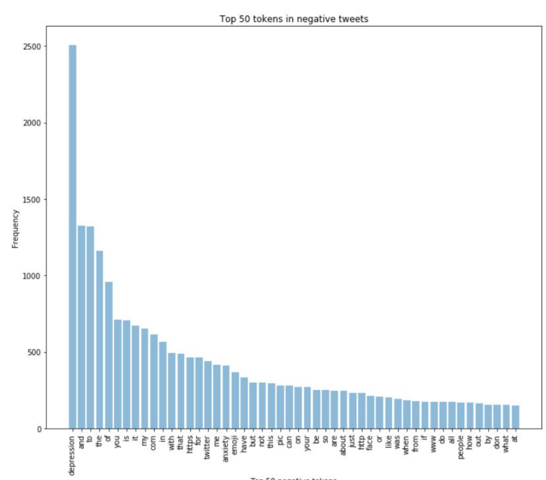

Fig. 6. The graph depicts the top 50 words used in negative tweets.

50 Top tokens in positive tweets Fig.2 shows the bar graph plot for the word frequencies in positive tweets.On x-axis we have the frequencies and on y-axis the tokens, each bar represents the frequency for a particular word used in positive tweets.

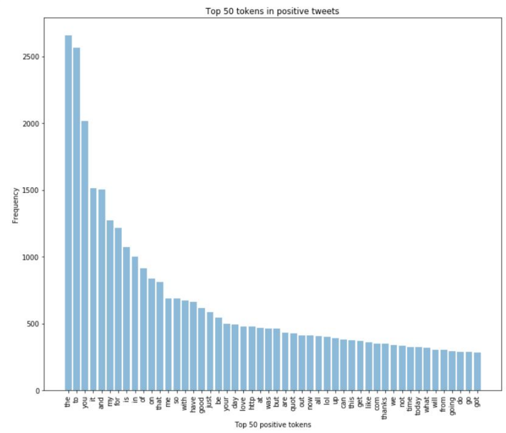

Fig. 7. The graph depicts the top 50 words used in positive tweets.

Negative frequency vs Positive frequency Fig.3 shows the seaborn plot for the word frequencies in positive tweets and negative tweets.On x-axis we have the negative frequencies and on y-axis the positive frequencies The graph is used to compare their frequencies.

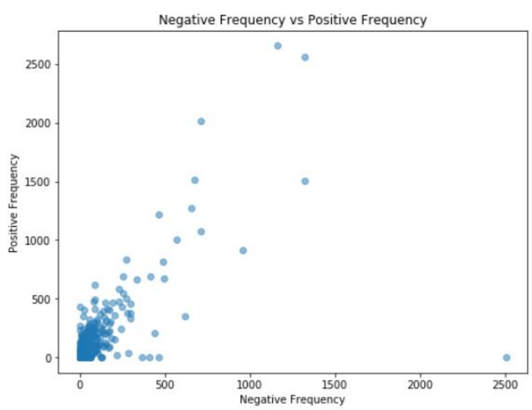

Fig. 8. The graph depicts negative frequency vs positive frequency.

Number of tweets with a given length Fig.4 shows the histogram plot for the tweets length, that is it depicts the number of tweets with a particular length represented on x-axis of the plot.

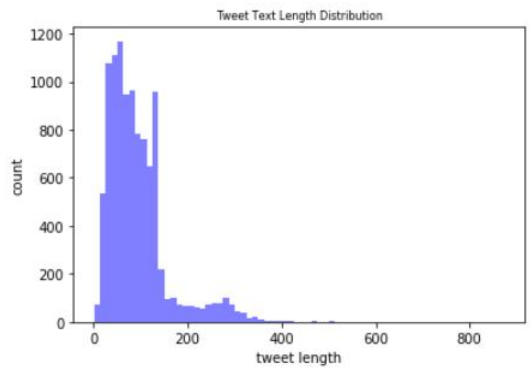

Fig. 9. Histogram depicting length of tweets

Distribution of tweet sentiment polarity score Fig.5 shows the histogram plot for the tweets sentiment polarity score, as vast majority of the sentiment polarity scores are greater than zero, means most of them are pretty positive.

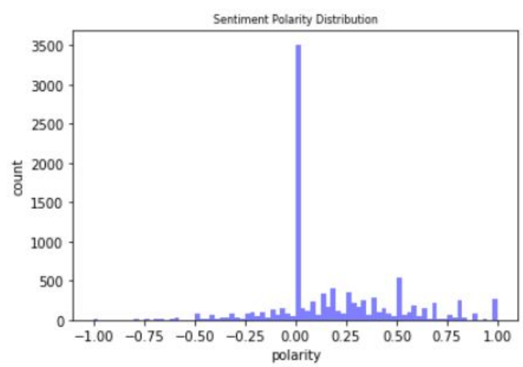

Fig. 10. Histogram depicting sentiment polarity score of tweets

Distribution of tweets word count Fig.6 shows the histogram plot for the tweets word count. There were quite number of people like to leave long reviews.For categorical features, we simply use bar graph to present the frequency.

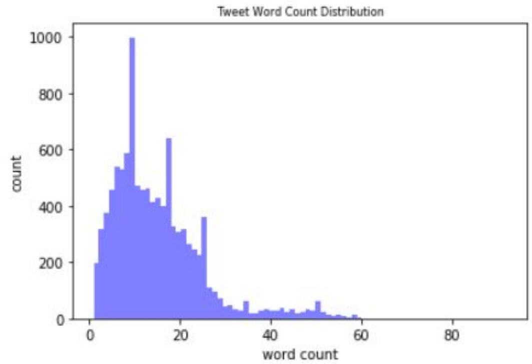

Fig. 11. Histogram depicting tweets word count

7 Results

The result obtained after applying the bag of words and TF-IDF model on the data set are represented using the following metrics:

1.Accuracy:Accuracy is a ratio of correctly predicted values to the total values present.

$$Accuracy = TruePositives+TrueNegatives/TruePositives+FalsePositives+FalseNegatives+TrueNegat \tag{2}$$

2.Precision:Precision is the ratio of correctly predicted positive values to the total predicted positive values.

$$Precision = TruePositives/TruePositives + FalsePositives \tag{3}$$

3.Recall (Sensitivity) - Recall is the ratio of correctly predicted positive values to the all values in actual class.

$$Recall = TruePositives/TruePositives + FalseNegatives \tag{4}$$

4.F1 score - F1 Score is the weighted average of Precision and Recall. It is a score that takes both false positives and false negatives values into account.

$$F1Score = 2 * (Recall * Precision)/(Recall + Precision) \tag{5}$$

So reference to table 4. depicts the various performance measures for comparing the bag of words and TF-IDF model to predict the depression in the tweets.

Table 4. Performance measures for the applied approach.

Metrics	Bag of words	TF-IDF
Precision	0.9285714285714286	0.9615384615384616
Accuracy	0.7967914438502673	0.8609625668449198
Recall	0.26	0.5
F1 Score	0.40625	0.6578947368421052

8 Future Work and Conclusion

We have implemented naive Bayes algorithm. In future we will implement various other machine learning algorithms like decision trees and support vector machine as well. Also,for future , we would want to study depression among groups of people based on their gender,age, locations and other attributes. We would also like to find other mental disorders as well, and detecting suicidal thoughts if any.

In conclusion, we proposed a model for tweet-level classification and used it to classify the depressed tweets for each person who uses this social media. We also performed various visualization on our dataset for better understanding of data.

References

1. World Health Organization. The ICD-10 Classification of Mental and Behavioural Disorders , 1978.
2. Ahmed Husseini Orabi, Prasadith Buddhitha, Mahmoud Husseini Orabi, Diana Inkpen *Deep Learning for Depression Detection of Twitter Users.*
3. Dataset,
`https://www.kaggle.com/kazanova/sentiment140.html`