**University of Ibadan**
**Journal of Science and Logics in ICT Research**

# Classification of Depression through Social Media Posts Using Machine Learning Techniques

**[1]✉ WOODS, N. C. and [2] ADEDEJI, A. M.**
*University of Ibadan, Ibadan, Nigeria*
*[1] chyn.woods@gmail.com, [2] plentymary@gmail.com*

***Abstract***

Machine Learning has been applied to solve several problems in various areas of life such as medicine, sciences and industries. Depression is a major problem across the globe and is becoming a serious challenge in the health sector. Millions of people suffer from depression, at different levels, but only few take preventive measures and get appropriate treatment, due mainly to the fact that early detection of depression may be cumbersome. A deep study of an individual's behaviour could led to early detection and some of these behaviours can be gotten through social media platforms. This study seeks to analyse users' tweets gotten from twitter and classify depressive contents into four levels, rather than the usual two-tier depression classification. Users' tweets were extracted using twitter API and a web scrapping tool called 'Twint'. Bag of words model, Term Frequency-Inverse Document Frequency and a text pre-processing tool provided by Keras framework, were used to quantify and comparatively evaluate how different models influenced the classification of tweets. Three machine learning algorithms; Naïve Bayes, Random Forest and Decision Tree were used for the classification. The result reveals that Random Forest best classifies the tweets into the four categories of depression.

***Keywords****: Depression, Tweets, Social media, Bag of Words (BOW), Term Frequency-Inverse Document Frequency (TF-IDF), Tokenizer, Naïve Bayes, Random Forest, Decision Tree.*

## 1. INTRODUCTION

Depression is determined by persistent low mood, fatigue, poor concentration, loss of interest in normally enjoyable activities and it often comes with a feeling that life isn't worth living. It is a serious and common mental disorder that affects one's feelings, thoughts and behaviours, and it greatly contributes to the economic, social and physical burden of people around the globe. Alongside other mental disorders, it has been related to early termination of education, unstable marriages, teenage pregnancy, role impairment, heart disease, suicide and other negative outcomes [1, 2].

Social media has become part of nearly everyone's daily routine, where people are connected almost all the time performing several activities on Facebook, twitter, WhatsApp, Instagram, snapchats etc. Social media is seen as a platform where people express and share their feelings, opinions, experiences, beliefs and almost all of their daily activities. All these uploads and updates by users contain information about their demographics, likes and dislikes, which can be collected and analyzed through various techniques. These techniques include machine learning and classical statistics such as neural networks, natural language processing (NLP) and sentiment analysis.

An estimate of one in 15 adults (6.7%) in any given year is affected by depression and one in six persons (16.6%) will experience depression at some time in their life. Depression can strike at any time, but on average, it first appears during the late teens to mid-20s, women are more likely to be depressed than men as studies showed that one-third of women will experience a major depressive episode in their lifetime [3]. Therefore, with the daily increase in the use of social media, users' posts and

Woods, N. C. and Adedeji, A. M. (2021). Classification of Depression through Social Media Posts Using Machine Learning Techniques. *University of Ibadan Journal of Science and Logics in ICT Research* (UIJSLICTR), Vol. 7 No. 1, pp. 19 -28
**©UIJSLICTR Vol. 7, No. 1, December 2021**

updates can be accessed, collected and analysed.

In this study, users' tweets were collected from their twitter accounts, the choice of twitter over other social media platforms boils down to the fact that it has more of text-based posts and text mining is crucial for this work. A model was introduced to read through the features in the texts and thereby determining if a user is depressed or not. The model further classified the level of depression using ICD-10 and DSM-TR classification using three machine learning techniques; Naïve Bayes, Random Forest and Decision tree.

## 2. RELATED WORKS

The study by Ahmed *et. al*., [4] identified some effective deep neural network among a few selected architectures that were successfully used in Natural Language Processing (NLP) tasks. The chosen architecture was used to detect users with signs of depression, given limited unstructured text data extracted from twitter. Four models were developed and built on top of word embeddings; three of the models used Convolutional Neural Network (CNN) and the last used Recurrent Neural Network (RNN). Support Vector Machine (SVM) linear classifier with Term Frequency- Inverse Document Frequency (TF-IDF) was used to initiate a baseline for the binary classification task. The experiment showed that CNN based model performed better than RNN based model with accuracy of 87.957%, F1 = 86.967%, precision= 87.435 and recall= 87.029.

Depression analysis on Facebook data collected from an online public source to investigate the effect of depression detection was done by Rafiqul *et. al*., [5]. They focused on four types of factors; emotional process, temporal process, linguistic style and all (emotional, temporal, linguistic style) features together for the detection of depression. Supervised machine learning approaches were applied to study each factor type independently. The classification techniques such as Decision tree, k-Nearest Neighbour, SVM and Ensemble were used for each type and it was observed that decision tree gave the best accuracy of 72%.

Another study carried out by Sharon Babu [6], was aimed at predicting if a user is at risk of depression using their Facebook status updates as the predictors. Algorithms such as logistic regression, SVM and random forest were used to solve the classification problem. Improvements were recorded from the baseline accuracy, using a model that employed both regression and TF-IDF which gave a better accuracy of 88%.

An automated system that can identify at-risk users from their public social media activity, specifically through tweets was proposed by Zunaira *et.al.* [7]. In their work, a user level classifier was trained and a tweet level classifier that predicts if a tweet indicates depression was also trained. They achieved a precision of 0.1237, recall of 0.8020 and F1 of 0.2144.

The study conducted by Recee *et. al*., [8] extracted predictive features measuring linguistic style and context from twitter data to build models with supervised learning algorithms in order to predict emergence of depression and post-traumatic stress disorder in twitter users.

Tsugawa *et. al*., [9] in their own study used survey responses and status updates from 28,749 Facebook users to develop a regression model that predicts users' degree of depression across seasons. They discovered that the degree of depression increases from summer to winter and showed potential factors driving individual's level of depression. They achieved an accuracy of (r=.386), when a model was trained over all messages from user in the training set and then applied this model to all messages in the test set.

In another research by Schwartz et. al., [10], crowd sourcing was used to collect data of twitter users with clinical depression and they measured behavioural attributes to build a classifier in order to identify depression in a person.

In this study, a model that will classify depression into four categories using Naïve Bayes, Random Forest and Decision Tree was developed.

## 3. METHODOLOGY

The flow diagram in Figure1 summarizes the basic level of how our model was developed. The input data, which were tweets, were pre-processed to remove noise, after which, some features were extracted using Bag of Words (BOW), TF-IDF and Tokenizer. K-means clustering was used for relabeling target variables, the data was then normalized. Our model was then built and trained using three machine learning algorithms (Naïve Bayes, Decision Tree and Random Forest). The trained model was then used to classify new tweet text data into four different categories.

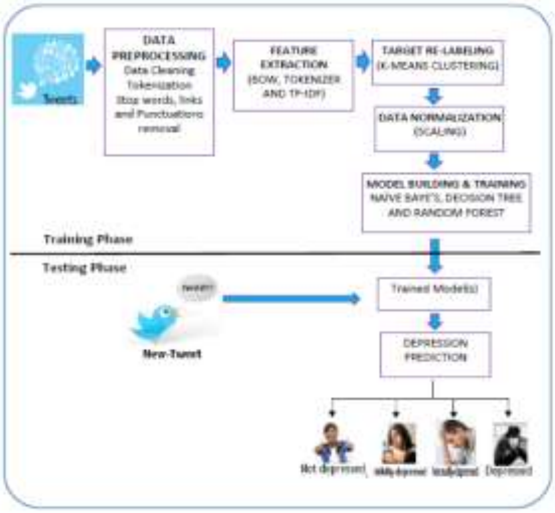


Figure 1: The architecture of the methodology.

This work was implemented using Python programming language on Google colaboratory, "COLAB", an online research platform with GPU and TPU support for Machine and Deep Learning project development. The study was run and tested on a 1xTesla K80, 3.7GHz computer, having 2496 CUDA cores, 12GB GDDR5 VRAM, with ~12.6 GB Available RAM size and ~33 GB Available Disk size. It was implemented using keras framework, a tool provided for implementing various machine and deep learning algorithms.

### 3.1 Dataset

Data was collected through twitter API, kaggle and using a web scrapping tool, "Twint". Hash tags were used with keywords to generate the needed texts. The dataset is made-up of 800,000 negative tweets, and 800,000 positive tweets. The negative tweets were annotated as '0', while positive tweets as "1". The tweets were extracted between the months of April 2019 and February 2020, using the basic keywords: positive emotion words, negative emotion words, sad words, angry words and anxiety words.

In order to understand the data better, we generated tokens of positive tweets and negative tweets, thereby producing two corpuses, one of depressive tweets and another of non-depressive tweets. These were plotted out on a wordcloud based on frequency of word occurrence using 'wordcloud' and "matplotlib" tools in python.

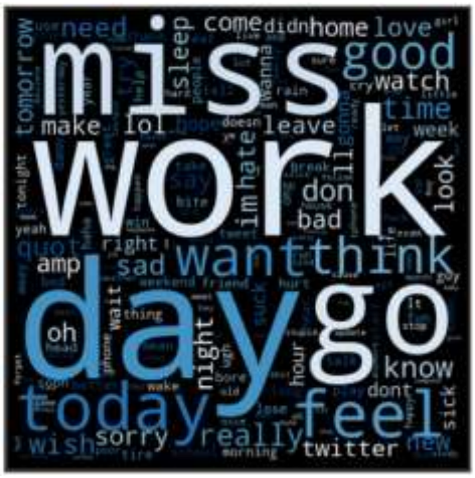


Figure 2: Word Cloud plot of Depressive Tweets

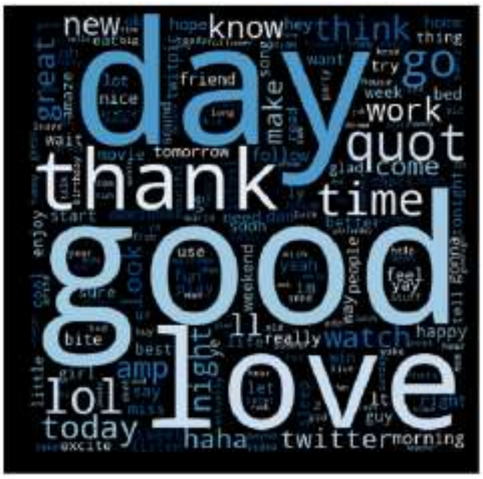


Figure 3: Word Cloud plot of Non-Depressive Tweets

As shown in Figures 2 and 3, the tweets classified as depressive ("0" = negative tweets) contained too many overlapping tokens with the non-depressive tweets, such that distinguishing between the depressive and non-depressive tweets might be hard for our machine learning model. To tackle this problem, we decided to source for more depressive tweets. Using "Twint", 16,467 depressive tweets were scrapped from Twitter. And these data constituted our second dataset. In order to ensure user privacy, we did not include users' personal or identifying information when scrapping the tweets.

### 3.2 Text Pre-processing

For each tweet in the datasets, the following text cleaning procedures were carried out:

**Removal of Non-alphabetic Characters:** All punctuations, html tags, hashtags, urls, special characters, quotations and numeric values in the tweets in the dataset were removed, as this will not be useful in the classification process, as well as all non-alphabetic characters in the tweets.

**Decapitalization:** After the first step, all tweets now contained only alphabetic characters. These tweets were then converted into their lower-case representation.

**Removal of stopwords:** After decapitalizing all tweets, all words that would not contribute to classification of depressive or non-depressive tweets were removed. These included words such as prepositions (in, above, over, on, from, at, over, between), conjunctions (and, or, with), articles (a, the, an).

**Stemming and Lemmatization:** The next text cleaning step was stemming and lemmatization of polymorphic words. **"Stemming"** is the process of extracting the root word of words that can take different forms. For example; loving is stemmed as "love". This was achieved using PorterStemmer. **Lemmatization** is a method used to group different inflected forms of words into the root form. For example; love, loved, loving, are lemmatized as "love" using WordNetLemmatizer tools of nltk library.

### 3.3 Feature Extraction

Machine learning algorithms cannot work with raw text directly; the text must be converted into numeric values of some sort, usually vectors of numbers. In order to feed the tweets into a machine learning algorithm for classification, there was need to build a numeric model of each tweet as vector representation of fixed length, suitable for machine learning modelling, training and classification purposes. In this work, the Bag-Of-Words (BOW), Tokenizer (a text-pre-processing tool provided by 'Keras' framework), and TF-IDF models were used. Our choice of three models is to comparatively evaluate how different models influenced the classification of depressive tweets.

#### 3.3.1 Bag-Of-Word (BOW) Model

BOW model was chosen, because of its simplicity, flexibility for customization, and ease of implementation and representation. The model is only concerned with whether known words occur in a document, not necessarily where in the document [11]. The process, involves two (2) major steps: generating vocabulary of known words and scoring of words. To generate the vocabulary of known words as well as cater for the scalability problem of BOW, a series of text cleaning was carried out. This then generated the corpus and vocabulary of the cleaned tweets containing 564,181 words. A sample of the tweets are shown in the appendixes.

After creating a vocabulary, the occurrence of words in example documents was scored. This process involved generating numeric values (as vector) of fixed length for each text (tweet) under consideration. In order not too loose too much data nor introduce too much bias if we arbitrarily choose a vector size, we calculated the mean of the length of all cleaned tweets which gave us 42, and used that as the fixed length size for all vectors. The result (matrix of vectorized tweets) of this phase was the input to our machine learning models.

#### 3.3.2 Term Frequency – Inverse Document Frequency (TF-IDF) Model

A problem with scoring word frequency is that highly frequent words start to dominate in the document (e.g. larger score), but may not contain as much "informational content" to the model as rarer but perhaps domain specific words. One approach is to rescale the frequency of words by how often they appear in all documents, so that the scores for frequent words that are common across all documents is penalized. This approach to scoring is called

Term Frequency – Inverse Document Frequency (TF-IDF) [12].

TF is calculated using

$$tf(t_{i,j}) = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

The *tf*, for term, $t_{i,j}$, in document, j, is calculated as the number of occurrences, n, of term, i, in document, j, divided by the total number of all terms, $n_{i\,..}\,n_k$ in j (where k is the number of terms in document j).

The Inverse Document Frequency is a scoring of how rare the word is across all documents (tweets) and can be calculated using:

$$idf(t_{i,j}) = \log\left(\frac{N}{df_t}\right)$$

TF-IDF is an information retrieval technique that weighs a term's frequency (TF) and its inverse document frequency (IDF). The TF-IDF weight, of term $t_{i,j,}$ is the product of the TF and IDF scores of a term.

$$w(t_{i,j}) = tf(t_{i,j}) * idf(t_{i,j})$$

TF-IDF was chosen so as to get the words in vocab that are relevant and of high importance rather than having words with least importance taking the highest number of counts.

TF-IDF was used to rescale the frequency of the common words based on how they often appear in all documents in a corpus, such that the scores for frequent words across all documents are penalized. The result of this rescaling was the input to our machine learning models.

### 3.3.3 Tokenizer Model

This Keras tool generates dictionary of index for each word from a list of texts (tweets) it is fitted on, which can later be used to generate sequences for consequent texts to be encoded. It uses the index generated during fitting to identify and provide a sequence containing a list of the numeric index value for every word found in the index dictionary corresponding to the words that make up the text (tweet).

For this research, we calculated the average number of words that made-up all tweets and used that as the value for *maxlen* argument of pad_sequences() method. Fitting the Tokenizer on the cleaned tweets, our vocabulary size was 45827 and average tweet length was 53, we had to remove the first ten (10) columns because they contained no single value, as such, we only used 43 columns or features.

### 3.3.4 Re-Labelling of Target Variable with K-Means Clustering

The focus of this research is to classify tweets into non-depressed, mildly-depressed, moderately-depressed and severely depressed. Considering the datasets available, which has only two (2) classes, non-depressed and depressed, there was need to re-label the target variable. This was accomplished using K-Means Clustering Algorithm to re-label the cleaned and encoded dataset.

With K-Means, we clustered the non-depressed datasets into two with the intent to use one class as non-depressed and the other as mildly-depressed, and also the depressed tweets into two clusters, one as moderately-depressed and the other cluster as severely-depressed classes respectively. These clusters were then re-labelled into four (4) classes (0: non-depressed, 1: mildly-depressed, 2: moderately-depressed, and 4: severely-depressed) and then assigned as our target variable.

The labels obtained through K-Means was highly skewed towards the two extremes. To tackle this problem, we extracted 80,000 observations from the entire datasets, constituting 20,000 carefully sampled observations for each class. This approach eliminated the skewness that could have introduced bias into the models during classification, in favour of the majority.

### 3.3.5 Data Normalization or Scaling

The encoded datasets was normalized before using them for training, because of the computational requirements of working with high-dimensional data, a situation peculiar to our research work. The encoded dataset was then scaled using min-max scaling as shown below:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Where x' is the scaled form of value x belonging to a particular observation of a given feature, $x_{min}$ is the minimum value and $x_{max}$ is the maximum value of the feature. Min-max scaling was applied to all features generated by each of our text encoding models. The scaled features were fed into our respective algorithms for training and evaluation. Finally, each text model was then split into 80% and 20% for training and testing sets respectively.

### 3.4 Building and Training the Machine Learning Models

The three machine learning algorithms used in this work were Decision Tree Classifier (DTree), Naïve Baye's (NB), and Random Forest Classifier (RF). From each machine learning algorithm, we built three models, one taking the vector generated from Tokenizer model, another taking the vector generated from TF-IDF word models and the third, taking vector generated from BOW model as inputs respectively for training and evaluation.

## 4. RESULTS AND DISCUSSION

### 4.1 Results

A model was built that classified users into various depression categories (Not depressed, Mildly depressed, Moderately depressed and Severely depressed) based on three machine learning techniques; Naïve Bayes, Decision Tree and Random Forest.

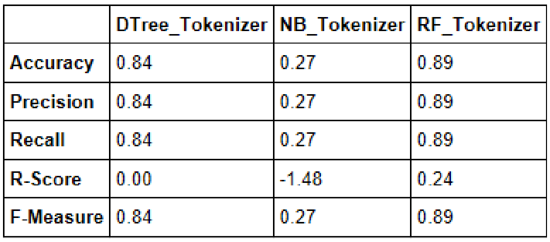
Each of the models was built and evaluated using five metrics; precision, recall, R-score, accuracy and F-measure and for each of the algorithms, comparisons were done.

### Comparison of All Models Results

***Comparing results of the three models taking vector from tokenizer.***

Table 1 shows the result of comparison of all models taking vector from tokenizer. From Table 1 it can be seen that DTree had an accuracy of 0.84, NB gave an accuracy of 0.27 and RF had an accuracy of 0.89. From this result, we observed that the model using RF and taking vector from tokenizer best classified the tweets.

Table 1 Results of all models using Tokenizer

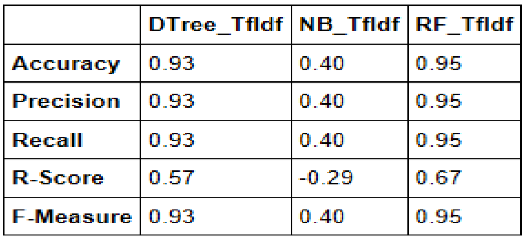


| | DTree_Tokenizer | NB_Tokenizer | RF_Tokenizer |
|---|---|---|---|
| Accuracy | 0.84 | 0.27 | 0.89 |
| Precision | 0.84 | 0.27 | 0.89 |
| Recall | 0.84 | 0.27 | 0.89 |
| R-Score | 0.00 | -1.48 | 0.24 |
| F-Measure | 0.84 | 0.27 | 0.89 |

***Comparing results of the three models taking vector from TF-IDF***

Table 2 shows the result of comparison of all models using TF-IDF. From this table it can be seen that DTree with TF-IDF had an accuracy of 0.93, NB with TF-IDF had an accuracy of 0.40 while RF gave an accuracy of 0.95. From this result, we observed that RF taking vector from TF-IDF performed best in this category.

Table 2: Result of models using TF-IDF

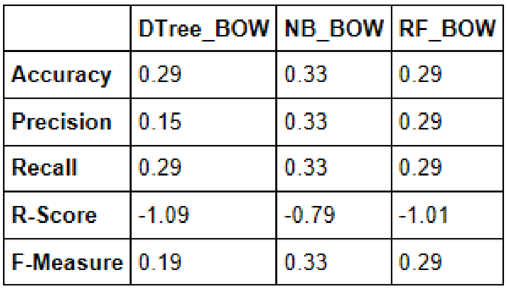


| | DTree_TfIdf | NB_TfIdf | RF_TfIdf |
|---|---|---|---|
| Accuracy | 0.93 | 0.40 | 0.95 |
| Precision | 0.93 | 0.40 | 0.95 |
| Recall | 0.93 | 0.40 | 0.95 |
| R-Score | 0.57 | -0.29 | 0.67 |
| F-Measure | 0.93 | 0.40 | 0.95 |

***Comparing results of the three models taking vector from BOW.***

Table 3 shows the result of the comparison of all models using BOW, while Figure 4 shows the histogram of all models using BOW. From Table 3, we see that DTree had an accuracy of 0.29, NB had an accuracy of 0.33 and RF gave an accuracy of 0.29. From this result, we observed that the models taking vector from BOW poorly classified the tweets.

Table 3: Result of models using BOW



| | DTree_BOW | NB_BOW | RF_BOW |
|---|---|---|---|
| Accuracy | 0.29 | 0.33 | 0.29 |
| Precision | 0.15 | 0.33 | 0.29 |
| Recall | 0.29 | 0.33 | 0.29 |
| R-Score | -1.09 | -0.79 | -1.01 |
| F-Measure | 0.19 | 0.33 | 0.29 |

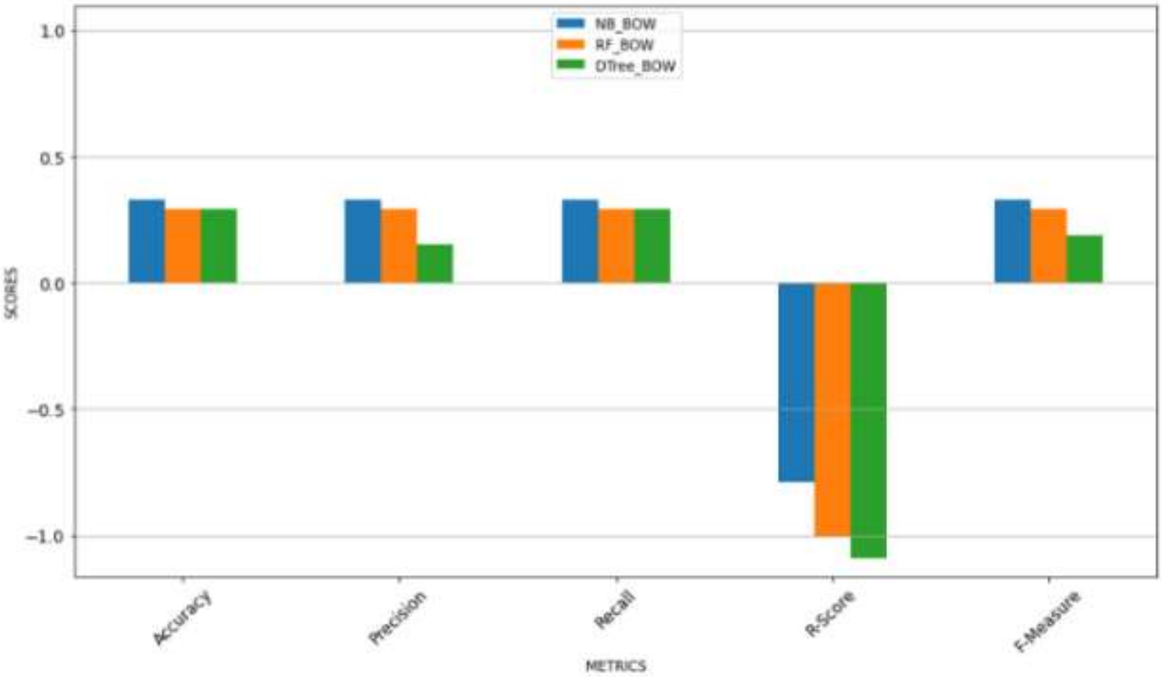


Figure 4: Histogram of all models using BOW

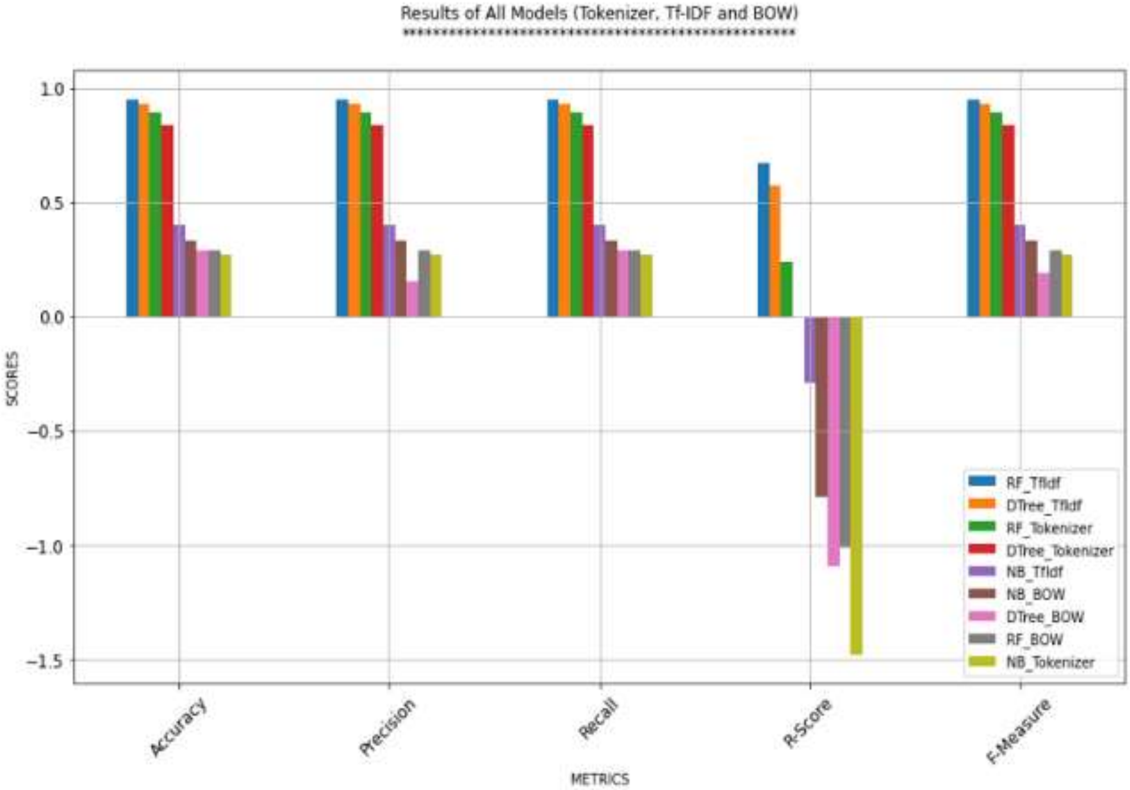


Figure 5: Histogram of all models

Figure 5 depicts the combined Histogram of all the models. From this figure we see that the model developed with RF and taking vector from TF-IDF gave the best results in all categories.

**4.1 Discussion**

The accuracy of the models built with Random Forest algorithm were higher, taking vectors from Tokenizer and TF-IDF, but was very low taking vector from BOW. The Decision Tree algorithm models also classified the tweets better taking vector from Tokenizer and TF-IDF but not as good, taking vector from BOW. Lastly, the models built with Naïve Bayes poorly classified the tweets, taking vector from Tokenizer, TF-IDF and BOW with low accuracy. In comparison with reviewed related works, Random Forest algorithm best classified the tweets.

## 5. CONCLUSION

This research proposed models for classifying depression levels using textual data representation of tweets, such that tweets belonging to the same cluster label can be classified without user interaction or input. The models classified depression into a four-tier taxonomy. RF, DTree and NB algorithms were used for the classification, each taking vector from Tokenizer, TF-IDF and BOW. From the results obtained, it is therefore notable to state that the model developed with RF and taking vector from TF-IDF gave the best results.

## References

[1] Kessler, R.C. and Bromet, E.J. (2013). The Epidemiology of Depression Across Cultures. *Annu Rev Public Health*. 2013;34:119-38. doi:10.1146/annurev-publhealth-031912-114409.

[2] Lichtman, J.H., Froelicher, E.S., Blumenthal, J.A., Carney, R.M., Doering, L.V and Frasure-Smith, N. (2014). Depression as a risk factor for poor prognosis among patients with acute coronary syndrome:Systematic Review&Recommendation *A scientific statement from the American Heart Association*.

[3] American Psychiatric Association. (2013). Diagnostic and statistical manual of mental disorders *(5th ed.), 5 editions. American Psychiatric Publishing, Washington*.

[4] Ahmed Husseini Orabi, Prasadith Buddhitha, Mahmoud Husseini Orabi and Diana Inkpen. (2018). Deep Learning for Depression Detection of Twitter Users. *In Proceedings of the fifth workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*,pp 88-97,doi:10.18653/v1/W18-0609, http://www.aclweb.org/anthoogy/W180609.

[5] Md. Rafiqul Islam, Muhammad Ashad Kabir, Ashir Ahmed, Abu Raihan M. Kamal, Hua Wang and Anwaar Ulhaq. (2018). Depression Detection from Social Network Data Using Machine Learning Techniques. *Department of Computer Science and Engineering, Islamic University of Technology(IUT), Dhaka, Bangladesh* Health Inf Sci Syst 6:8. https://doi.org/10.1007/s13755-018-0046-0.

[6] Sharon Babu (2018). "Predicting Depression from Social Media Updates" *Donald Bren School of Information and Computer Sciences*.

[7] Zunaira Jamil, Diana Inkpen, Prasadith Buddhitha and Kenton White. (2017). Monitoring Tweets for Depression to Detect At-risk Users. School of Electrical Engineering and Computer Science, University of Ottawa, *in proceedings of the fourth workshop on computational linguistics and clinical Psychology*. January, 2017. doi: 10.18653/v1/W17-3104.

[8] Andrew G. Recee, Andrew J. Reagan, Katharina L.M. Lix and Peter Dodds. (2017). Forecasting the onset and course of mental illness with twitter data. *Scientific Reports 7(1)*, doi:10.1038/s41598-017-12961-9.

[9] Sho Tsugawa., Yusuke Kikuchi., Fumio KIshino., Kosuke Nakajima., Yuichi Itoh and Hiroyuki Ohsaki. (2015). Recognizing Depression from Twitter Activity. *In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* – CHI '15. Pages 3187-3196. https://doi.org/10.1145/2702123.2702280.

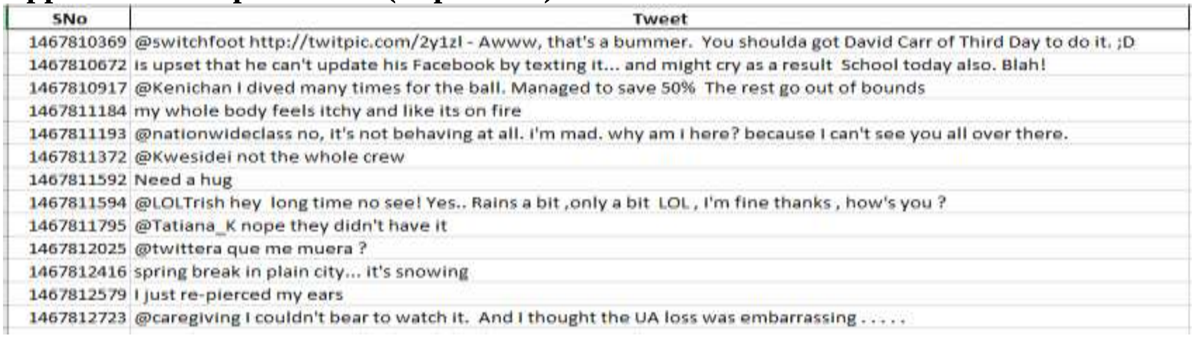
[10] H Andrew Schwartz, Johannes Eichstaedt, Margaret L Kern, Gregory Park, Maarten Sap, David Stillwell, Michal Kosinski, and Lyle Ungar. (2014). Towards Accessing Changes in Degree of Depression through Facebook. *In proceedings of the workshop on computational linguistics and clinical Psychology: From Linguistics Signal to Clinical Reality*, pages 118-125.

[11] Jason Brownlee. (2017). A Gentle Introduction to the Bag-of-Words Model. Deep Learning for Natural Language Processing. Machine Learning Mastery. https://www.machinelearningmastery.com/gentle-introduction-bag-words-model/

[12] William Scot. (2019). TF-IDF for scratch in python on a real-world dataset. *Towards datascience.*https://towardsdatascience.com/tf-idf-fordocument-ranking-from-scratch-in-python-on-real-world-dataset- 796d339a4089.

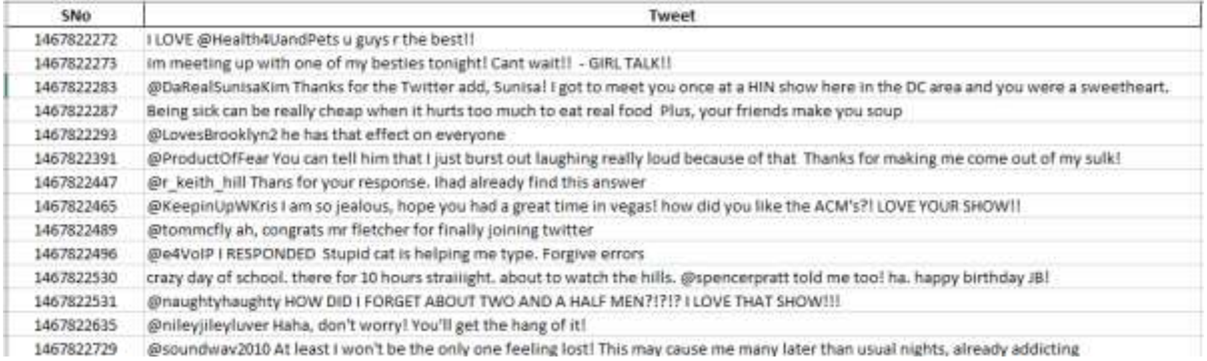
# Appendix

## Appendix i: Sample Tweets (Depressive)



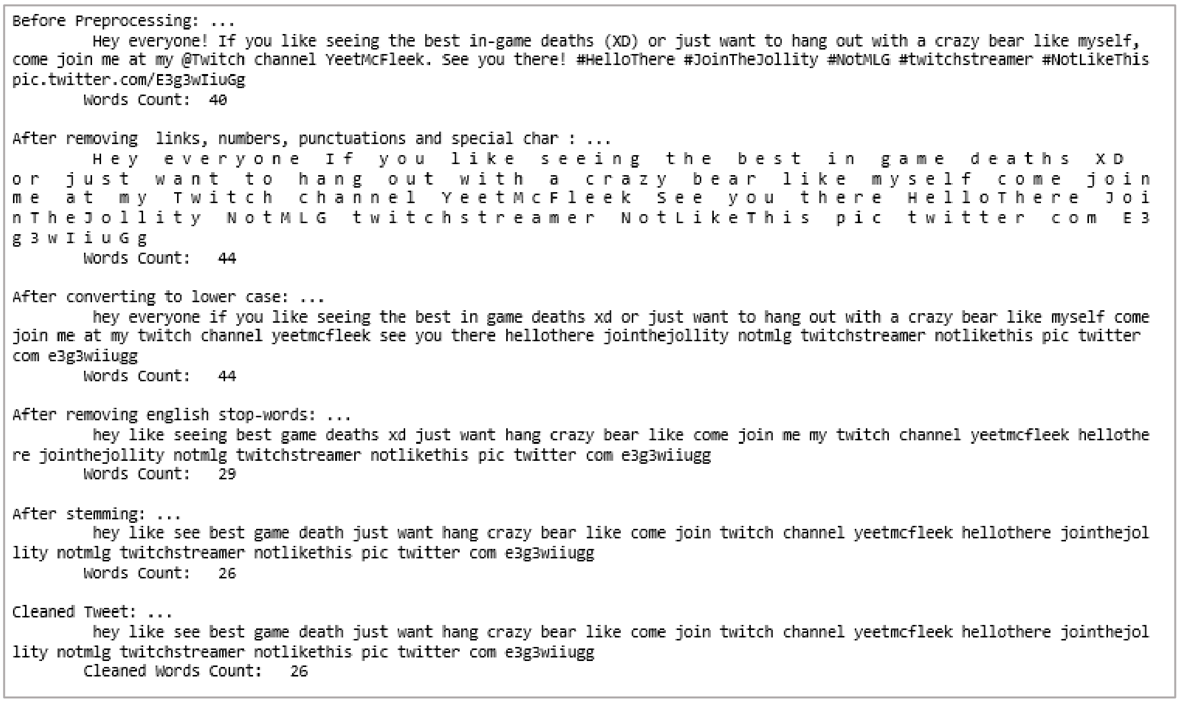
| SNo | Tweet |
|---|---|
| 1467810369 | @switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer.  You shoulda got David Carr of Third Day to do it. ;D |
| 1467810672 | is upset that he can't update his Facebook by texting it... and might cry as a result  School today also. Blah! |
| 1467810917 | @Kenichan I dived many times for the ball. Managed to save 50%  The rest go out of bounds |
| 1467811184 | my whole body feels itchy and like its on fire |
| 1467811193 | @nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't see you all over there. |
| 1467811372 | @Kwesidei not the whole crew |
| 1467811592 | Need a hug |
| 1467811594 | @LOLTrish hey  long time no see! Yes.. Rains a bit ,only a bit  LOL , I'm fine thanks , how's you ? |
| 1467811795 | @Tatiana_K nope they didn't have it |
| 1467812025 | @twittera que me muera ? |
| 1467812416 | spring break in plain city... it's snowing |
| 1467812579 | I just re-pierced my ears |
| 1467812723 | @caregiving I couldn't bear to watch it.  And I thought the UA loss was embarrassing . . . . . |

## Appendix ii: Sample Tweets (Non-Depressive)



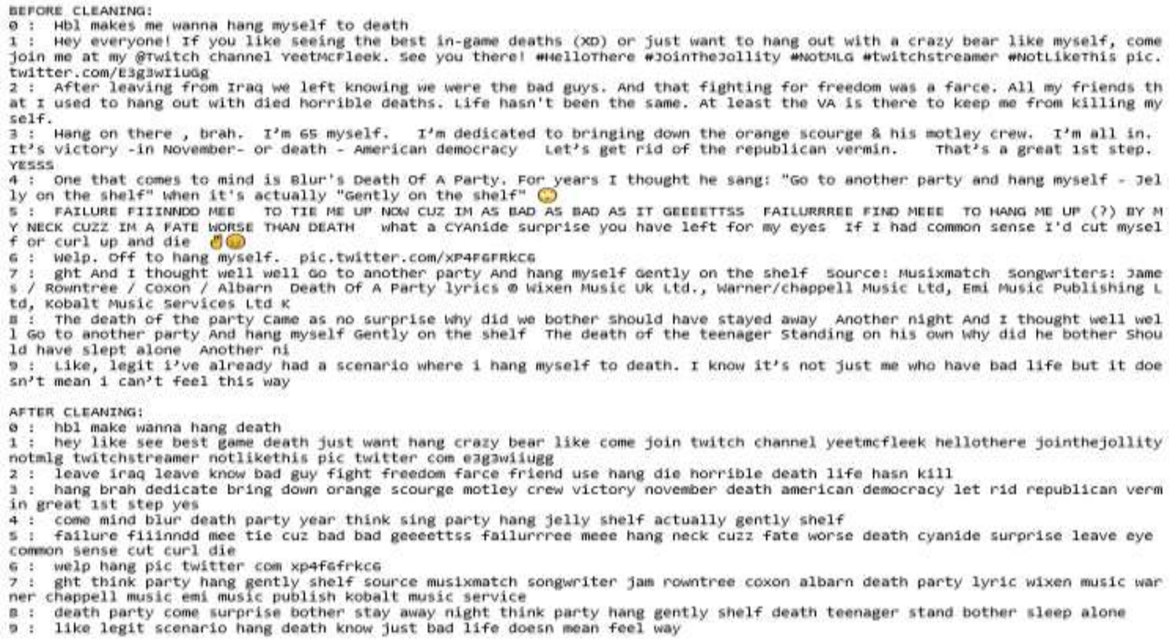
| SNo | Tweet |
|---|---|
| 1467822272 | I LOVE @Health4UandPets u guys r the best!! |
| 1467822273 | im meeting up with one of my besties tonight! Cant wait!!  - GIRL TALK!! |
| 1467822283 | @DaRealSunisaKim Thanks for the Twitter add, Sunisa! I got to meet you once at a HIN show here in the DC area and you were a sweetheart. |
| 1467822287 | Being sick can be really cheap when it hurts too much to eat real food  Plus, your friends make you soup |
| 1467822293 | @LovesBrooklyn2 he has that effect on everyone |
| 1467822391 | @ProductOfFear You can tell him that I just burst out laughing really loud because of that  Thanks for making me come out of my sulk! |
| 1467822447 | @r_keith_hill Thans for your response. Ihad already find this answer |
| 1467822465 | @KeepinUpWKris I am so jealous, hope you had a great time in vegas! how did you like the ACM's?! LOVE YOUR SHOW!! |
| 1467822489 | @tommcfly ah, congrats mr fletcher for finally joining twitter |
| 1467822496 | @e4VoIP I RESPONDED  Stupid cat is helping me type. Forgive errors |
| 1467822530 | crazy day of school. there for 10 hours straiiight. about to watch the hills. @spencerpratt told me too! ha. happy birthday JB! |
| 1467822531 | @naughtyhaughty HOW DID I FORGET ABOUT TWO AND A HALF MEN?!?!? I LOVE THAT SHOW!!! |
| 1467822635 | @nileyjileyluver Haha, don't worry! You'll get the hang of it! |
| 1467822729 | @soundwav2010 At least I won't be the only one feeling lost! This may cause me many later than usual nights, already addicting |

## Appendix iii: Result of Cleaning a Sample Tweet



```
Before Preprocessing: ...
        Hey everyone! If you like seeing the best in-game deaths (XD) or just want to hang out with a crazy bear like myself,
come join me at my @Twitch channel YeetMcFleek. See you there! #HelloThere #JoinTheJollity #NotMLG #twitchstreamer #NotLikeThis
pic.twitter.com/E3g3wIiuGg
       Words Count:  40

After removing  links, numbers, punctuations and special char : ...
        H e y   e v e r y o n e   I f   y o u   l i k e   s e e i n g   t h e   b e s t   i n   g a m e   d e a t h s   X D
o r   j u s t   w a n t   t o   h a n g   o u t   w i t h   a   c r a z y   b e a r   l i k e   m y s e l f   c o m e   j o i n
m e   a t   m y   T w i t c h   c h a n n e l   Y e e t M c F l e e k   S e e   y o u   t h e r e   H e l l o T h e r e   J o i
n T h e J o l l i t y   N o t M L G   t w i t c h s t r e a m e r   N o t L i k e T h i s   p i c   t w i t t e r   c o m   E 3
g 3 w I i u G g
       Words Count:   44

After converting to lower case: ...
        hey everyone if you like seeing the best in game deaths xd or just want to hang out with a crazy bear like myself come
join me at my twitch channel yeetmcfleek see you there hellothere jointhejollity notmlg twitchstreamer notlikethis pic twitter
com e3g3wiiugg
       Words Count:   44

After removing english stop-words: ...
        hey like seeing best game deaths xd just want hang crazy bear like come join me my twitch channel yeetmcfleek hellothe
re jointhejollity notmlg twitchstreamer notlikethis pic twitter com e3g3wiiugg
       Words Count:   29

After stemming: ...
        hey like see best game death just want hang crazy bear like come join twitch channel yeetmcfleek hellothere jointhejol
lity notmlg twitchstreamer notlikethis pic twitter com e3g3wiiugg
       Words Count:   26

Cleaned Tweet: ...
        hey like see best game death just want hang crazy bear like come join twitch channel yeetmcfleek hellothere jointhejol
lity notmlg twitchstreamer notlikethis pic twitter com e3g3wiiugg
       Cleaned Words Count:   26
```

## Appendix iv: Sample Tweets Before and After Cleaning



```
BEFORE CLEANING:
0 :  Hbl makes me wanna hang myself to death
1 :  Hey everyone! If you like seeing the best in-game deaths (XD) or just want to hang out with a crazy bear like myself, come
join me at my @Twitch channel YeetMcFleek. See you there! #HelloThere #JoinTheJollity #NotMLG #twitchstreamer #NotLikeThis pic.
twitter.com/E3g3wIiuGg
2 :  After leaving from Iraq we left knowing we were the bad guys. And that fighting for freedom was a farce. All my friends th
at I used to hang out with died horrible deaths. Life hasn't been the same. At least the VA is there to keep me from killing my
self.
3 :  Hang on there , brah.  I’m 65 myself.   I’m dedicated to bringing down the orange scourge & his motley crew.  I’m all in.
It’s victory -in November- or death - American democracy   Let’s get rid of the republican vermin.    That’s a great 1st step.
YESSS
4 :  One that comes to mind is Blur's Death Of A Party. For years I thought he sang: "Go to another party and hang myself - Jel
ly on the shelf" When it's actually "Gently on the shelf" 😅
5 :  FAILURE FIIINNDD MEE   TO TIE ME UP NOW CUZ IM AS BAD AS BAD AS IT GEEEETTSS  FAILURRREE FIND MEEE  TO HANG ME UP (?) BY M
Y NECK CUZZ IM A FATE WORSE THAN DEATH   what a CYAnide surprise you have left for my eyes  If I had common sense I'd cut mysel
f or curl up and die  🤞😭
6 :  Welp. Off to hang myself.  pic.twitter.com/xP4F6FRkC6
7 :  ght And I thought well well Go to another party And hang myself Gently on the shelf  Source: Musixmatch  Songwriters: Jame
s / Rowntree / Coxon / Albarn  Death Of A Party lyrics © Wixen Music Uk Ltd., Warner/chappell Music Ltd, Emi Music Publishing L
td, Kobalt Music Services Ltd K
8 :  The death of the party Came as no surprise Why did we bother Should have stayed away  Another night And I thought well wel
l Go to another party And hang myself Gently on the shelf  The death of the teenager Standing on his own Why did he bother Shou
ld have slept alone  Another ni
9 :  Like, legit i’ve already had a scenario where i hang myself to death. I know it’s not just me who have bad life but it doe
sn’t mean i can’t feel this way

AFTER CLEANING:
0 :  hbl make wanna hang death
1 :  hey like see best game death just want hang crazy bear like come join twitch channel yeetmcfleek hellothere jointhejollity
notmlg twitchstreamer notlikethis pic twitter com e3g3wiiugg
2 :  leave iraq leave know bad guy fight freedom farce friend use hang die horrible death life hasn kill
3 :  hang brah dedicate bring down orange scourge motley crew victory november death american democracy let rid republican verm
in great 1st step yes
4 :  come mind blur death party year think sing party hang jelly shelf actually gently shelf
5 :  failure fiiinndd mee tie cuz bad bad geeeettss failurrree meee hang neck cuzz fate worse death cyanide surprise leave eye
common sense cut curl die
6 :  welp hang pic twitter com xp4f6frkc6
7 :  ght think party hang gently shelf source musixmatch songwriter jam rowntree coxon albarn death party lyric wixen music war
ner chappell music emi music publish kobalt music service
8 :  death party come surprise bother stay away night think party hang gently shelf death teenager stand bother sleep alone
9 :  like legit scenario hang death know just bad life doesn mean feel way
```