Depression Detection on Twitter Text Based on Negative Emotion Score and Level of Depressed

Houjun Ji*

St.John's Preparatory High School, Danvers, MA, United States

* Corresponding Author Email: hji24@stjohnsprep.org

Abstract. This essay presents a method for detecting depression level detection based on the negative emotion score of social media text. The score is calculated by the NRC Word-Emotion Association Lexicon (EmoLex). The dataset will first be processed by the EmoLex model and then based on certain scores given by the model, three levels of depression will be generated for further classification. The advantage of this model is that the detected depression is classified into three levels, which can help people take different levels of response. The experiment is performed in Multi Labeled Depression Corpus of 60,000 English tweets. Three models are trained in this experiment: Random Forest and Logistic Regression. The result shows that Random Forest achieves 65.9% accuracy; Logistic Regression achieve 72.2% accuracy.

Keywords: Depression Detection; Twitter Data; Emotion score; Level of Depressed.

1. Introduction

Depression is a serious mental disorder that causes damage to individuals. Moreover, such mental illness can have an impact on people around them as well since they always reveal negative emotions to the public [1]. According to the World Health Organization, approximately 3.8% of the world population, which is about 280 million people, are suffering from such mental disorders. It is considered as the fourth leading cause of death in 15 to 29-year-olds [2]. It can eventually lead to suicide for those who are considered as severe. After the COVID-19 pandemic, the number of people who got depression went up quickly. People who experienced COVID-19 infection or were forced to experience quarantine have greater chance of demonstrating depressed and somatic mood [3]. Thus, it is an important task to set up procedures to help those depression patients and prevent others from experiencing the same mental disorder.

At present time, people's social life usually takes place on social media, such as Twitter, Reddit, and Instagram. People can express their emotions and feelings through texts on these platforms. Thus, it can be used as a tool for detecting depression and identifying signs of depression through words from those who are mentally disordered. Among other social media platforms, Twitter posts are treated as a trustworthy source for detecting and analyzing depression. Twitter users tend to share their depressed feelings in a broader sense and express negative emotions with negative words [4]. Because of all the above reasons, this experiment considers using Twitter text data for training and testing models for depression detection.

Depression detection using artificial intelligence based on social media data has not been a new topic. More than 1500 thousands of papers have been published online. Within these thousands of papers, more than ten different models have been used and multiple datasets from different social media platforms have been used for training and studying [5]. However, under close examination, most of these studies used only two labels for classification, such as depressed or non-depressed and simply positive or negative. Although some of the papers try to use multiple labels for detection, the label does not help classify different levels of depression. Thus, these models have relatively limited impacts on assisting patients [6].

Based on the above ideas, this essay proposed a process of classifying depression levels using an emotion classification tool. The process proposed to extract negative emotion score from texts from Twitter and based on the overall score, these data are further divided into three level of depression.

Then, models are trained to detect depression levels from text words. The main contribution of this essay is the mullite-level classification for depression.

2. Literature Review

A similar approach has been proposed by R. Safa et al. This paper proposes that a range can be set for positive and negative emotion from 1 which represents positive and -1 as negative. If emotion falls in between, which is zero, it is considered neutral. The paper then used an algorithm to calculate the polarity of word emotion. This algorithm includes a modified version of EmoLex, which adjusts the weight of depression and sadness to let more data fall in the range of the negative polarity. The result shown in the paper suggests such an approach can help better detect accuracy. The accuracy reached 69% [7].

The tool used in this essay for the calculation of the negative emotion score is the NRC Emotion Lexicon which can detect eight basic emotions, such as anger, sadness, and happiness. The model contains a list of words and an emotion score, to which this emotional word corresponds. Based on the total score of words in a given sentence, the score of one of the emotions is calculated. This model can help provide an overview of how depressed this sentence is expressed to people reading it, which can help classify levels of depression [8].

The processing of the Twitter text referenced to the Bag-of-Word function. The Bag-of-word treats the sentence as a vector with the occurrences of words within the overall dataset. Each sentence is being treated as a single document in the bag of word sentences. The document records the occurrence of a featured word in the given sentence and formulates it into a vector. The step is repeated until every sentence is being processed [9]. The advantage of the Bag-of-Word function is that it is easy to process, and the model can understand it easily. The disadvantage is that the feature of the sentence is not being considered. Each word might have a different weight when considering the detection of certain emotions.

This paper combines the idea proposed by R. Safa et al. and the Lexicon model. The dataset is first preprocessed by stemming and dropping out all the stopping words. Then each text data is inputted into the Lexicon model to get a series of emotion scores. After that, based on the combined score of negative motion, three levels of depression can be classified and be used for training. The improvement is getting a combined emotion score and multi-labeled dataset for further depression detection and response.

3. Methodology

3.1. Dataset Processing

3.1.1 The Dataset

The dataset used in this paper is called the Multi Labeled Depression Corpus of 60,000 English tweets. The dataset contains 60,000 tweets from Twitter. It does not contain the date posted, tags of this tweet, and other user information. The dataset also contains three different labels with pre-classified labels. These labels show emotions revealed by this tweet the labels range from -1 to 1. -1 corresponds to positive. 1 corresponds to negative, and 0 corresponds to neutral. Since this paper mainly focuses on the detection of the depression level of tweets, non-negative text information is deleted from the dataset. Only 17668 tweets are being used as data for the classification and detection process.

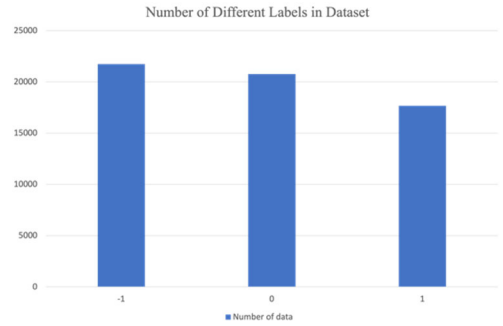


Fig. 1 Description of components of dataset

3.1.2 Word Stemming

Word stemming reduces words to their root state. Such a method can significantly reduce a total number of different words in a document, which can save computational time and space. This experiment uses a Snowball stemmer for stemming. It performs well on dropping out affixes and suffixes. However, some words are processed incorrectly, which cause some problem in later training and testing.

3.1.3 Drop Out Stop Words

Stop words are words that are commonly used in the English language. Most of them are insignificant since they do not have meanings when trying to extract information from sentences. Because of this, in this experiment, all the stop words are removed from text data so that the model can focus on important words that express strong emotion.

Table 1. Sample text data

Text	Label
going through depression open this thread	Neagtive
know handle anxiety depress panic attack	Sadness
probably should just stay because this topic	Negative

3.1.4 Relabel the Dataset

To classify each text data into three level of depression, it is necessary to determine how negative or depressed this text reveal. To achieve this purpose, this experiment used the model NRC Emotion Lexicon, A model that can detect six different emotions based on scores calculated using words. The model contains a list of words that correspond to a given emotion, such as happiness. Each emotion word has a score that represents how strong the emotion is revealed by the word. The score does not have a limitation. It can range from 0 to 23 in this experiment. Since this experiment focuses on determining the value of depression of a Twitter user, only three of the total emotions are extracted as a measurement. They are anger, sadness, and negative.

Table 2. Sample scores generated from EmoLex.

Text	Sadness	Negative	Anger
sorry cant talk right doing girl	456	456	123
employees back stressful unpaid	789	213	644
they kill amanda suicide squad	213	654	649

Some of the scores of these three emotions are too low to be counted as negative. For example, text data might have a high anger and sadness emotion score, but it has a negative emotion score of zero. This is not considered valid data since this sentence does not reveal a sense of depression. For better detection results, these outliers are dropped out of the dataset. The total number of text data was reduced to 10040.

After a final cleaning of the overall dataset, it is ready to be relabeled. Since the sum of the three-emotion score of the text ranges from 2 to 18, every number below 6 will be counted as mild. Every

number between 6 to 11 will be considered as moderate and every number above 11 will be considered as severe. Here, this experiment uses numbers ranging from one to three to represent three levels of depression. As a result, this experiment gets a dataset with 4614 text data with mild depression, 4397 text data with moderate depression, and 1029 text data with severe depression. This is the final dataset that will be used for training and testing.

3.2. Converting Text Data

To let text data be better processed and used by the constructed model, it is necessary to convert the actual text data to vectors constructed by the occurrence of words in a given sentence. In this experiment, the Count Vectorizer is used for transforming text data into word vectors. It is very efficient when handling big datasets compared to other methods. After fitting the text data into the count vectorizer, 10040 texts are processed, and 21962 unique words are extracted from the overall dataset.

3.3. Constructing and Training Models

After all processes of preparations have been done, it is time to test the performance of the models this experiment chose when receiving these text data. The whole dataset is split into the train data and a test data. Only 30% of the data are used as test data to avoid overfitting the training models. In this experiment, two models are chosen to evaluate performance when given this dataset: Random Forest and Logistic Regression.

3.3.1 Random Forest Model

Random Forest is a great method when dealing with classification problems. It is fed with training data and training labels. Based on the given instances, the model makes predictions for the test data. The Random Forest model contains multiple decision trees. It acts as a base classifier for the data fed in. Based on the decision given by the formal decision tree, the next prediction is made by the next decision tree. The depth of the decision tree can be specified by the constructer of the model. In this experiment, the maximum depth is set as zero as too much depth of the decision tree will affect the overall efficiency for real-time predictions and might cause the problem of overfitting.

3.3.2 Logistic Regression Model

Logistic Regression works basically the same as Random Forest. It generates predictions based on the given training datasets. However, it is not exactly the same as Random Forest. In this model, each data is considered as a point in the logistic function. With the shape of the overall model, the given test data are predicted based on the trained logistic function. It performs best when encountering binomial scenarios, but it can still handle multinomial situations. The advantage of the Logistic Regression model is that it is easier to construct, and train compared to Random Forest.

4. Literature Review

After the model is constructed and predictions being made by models, charts are generated to help analyzing the overall performance of the two models.

4.1. Random Forest

The Random Forest achieved an accuracy of 65.9%, which is a considerably acceptable result when encountering multivariable situations. According to the confusion matrix, 1565 mild text data are successfully classified, 179 mild texts are classified as moderate, and surprisingly, none of the mild data are classified as severe. Only 3 moderate texts are being classified as severe, which is also surprising. The disappointing part of this matrix is that only a small proportion of severe text data are successfully classified.

Table 3. Confusion matrix of Random Forest

Labels	Mild	Moderate	Sever
Mild	1565	179	0
Moderate	759	1059	3
Severe	63	259	57

According to the classification report, the precision of the severe depression reaches the highest and the mild depression has the lowest accuracy. What is surprising from this report is that the macro average reached 75% for precision, which means that if the dataset is better balanced, the model could achieve a better result.

Table 4. Classification report for Random Forest

Labels	Precision	Recall	F1-Score
1	0.65	0.86	0.74
2	0.67	0.54	0.62
3	0.94	0.37	0.27
Macro Avg	0.75	0.53	0.53
Weighted Avg	0.69	0.66	0.63

4.2. Logistic Regression

The Logistic Regression model achieved an accuracy of 72.2%, which is a great result given such a situation. According to the confusion matrix, just like the Random Forest model, none of the mild texts are classified as severe, which is a piece of great news. The Logistic Regression model performs better on the severe part as 158 severe texts are correctly labeled. The moderate label also has some improvement as 1351 moderate labels are correctly classified by the Logistic Regression model.

Table 5. Confusion matrix of Logistic Regression Classifier

Labels	Mild	Moderate	Severe
Mild	1425	319	0
Moderate	459	1351	47
Severe	14	243	158

According to the classification report, the mild label has the best precision, and the moderate label has the least. What is surprising from this report is that the macro average only reached 72.2%, which is less than the Random Forest model. On the other hand, when it comes to the f1 score, each label has a higher score compared to the Random Forest model.

Table 6. Classification report for Logistic Regression Classifier

Labels	Precision	Recall	F1-Score
1	0.75	0.81	0.78
2	0.70	0.72	0.71
3	0.74	0.34	0.49
Macro Avg	0.73	0.63	0.66
Weighted Avg	0.72	0.72	0.72

5. Discussion

The overall performance of these two trained models is better than expected since the dataset is not very balanced as the mild depression takes up about half of the overall dataset. The other problem is that these two models, Random Forest and Logistic Regression does not perform very well when it comes to multinomial situation, especially for the Logistic Regression. The reason why an

imbalance dataset occurs is that the NRC Emotion Lexicon is not adjusted for this particular situation. In [7], they adjust the weight of some words when getting scores from the Emotion Lexicon, which makes more negative data counted and used for training and testing. In this experiment, the weight of negative words is not adjusted well. As a result, over half of the text data are dropped out of the dataset and left only one-third of the original data being used in the two models. Thus, it is surprising to see that these two trained models have such performance, which is better than the 69% achieved by [7].

6. Conclusion

This paper focuses on using the NRC Emotion Lexicon to give scores to text data from the Multi Labeled Depression Corpus of 60,000 English tweets. Based on the score calculated, new labels are generated to classify different levels of depression from users using the Random Forest model and the Logistic Regression model. These two reaches 65.9% and 72.2% accuracy. The future work will focus on adjusting the Emotion Lexicon model for better score calculation, more models being used such as convolutional neural network, and using other Twitter text as a dataset for testing the performance of this methodology.

References

[1] Ronald Kessler Evelyn Bromet, Peter Jonge, Victoria Shahly, and Marsha Wilcox. 2017. The burden of depressive illness. *Public Health Perspectives on Depressive Disorders*.

[2] World Health Organization: WHO & World Health Organization: WHO. (2023). Depressive disorder (depression). *www.who.int*. https://www.who.int/news-room/fact-sheets/detail/depression

[3] Zhang, J., Lu, H., Zeng, H., Zhang, S., Du, Q., Jiang, T., & Du, B. (2020). The differential psychological distress of populations affected by the COVID-19 pandemic. *Brain, behavior, and immunity*, *87*, 49.

[4] Park S, Lee SW, Kwak J, Cha M, Jeong B. Activities on facebook reveal the depressive state of users. J Med Internet Res 2013;15(10): e217.

[5] William, D., & Suhartono, D. (2021). Text-based depression detection on social media posts: A systematic literature review. *Procedia Computer Science*, *179*, 582-589.

[6] Babu, N. V., & Kanaga, E. G. M. (2022). Sentiment analysis in social media data for depression detection using artificial intelligence: a review. *SN Computer Science*, *3*, 1-20.

[7] Safa, R., Bayat, P., & Moghtader, L. (2022). Automatic detection of depression symptoms in twitter using multimodal analysis. *The Journal of supercomputing*, *78*(4), 4709–4744. https://doi.org/10.1007/s11227-021-04040-8

[8] Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a word–emotion association lexicon. *Computational intelligence*, *29*(3), 436-465.

[9] Qader, W. A., Ameen, M. M., & Ahmed, B. I. (2019, June). An overview of bag of words; importance, implementation, applications, and challenges. In *2019 international engineering conference (IEC)* (pp. 200-204). IEEE.