Opinion Mining in Social Networks for Depression Detection

Vishwas Shivakumar
Luddy School of Informatics, Computing and Engineering
Indiana University Bloomington
Bloomington, Indiana
vshivak@iu.edu

Saurav Subash Prasad
Luddy School of Informatics, Computing and Engineering
Indiana University Bloomington
Bloomington, Indiana
sasuba@iu.edu

Sushant Menon
Luddy School of Informatics, Computing and Engineering
Indiana University Bloomington
Bloomington, Indiana
ssmenon@iu.edu

Keywords—NLTK, Sentiment Analysis, depression, tokenization, lemmatization, bag of words

I. Introduction

Depression is a serious medical disorder that affects your feelings, thoughts, and actions. Depression creates melancholy or a loss of interest in previously appreciated activities. It can lead to various emotional and physical problems and decrease your function at work and home. Fortunately, depression is treatable with medication, psychotherapy, or electroconvulsive therapy. However, treatment for depression is often delayed, imprecise, or missed entirely.

We propose a system that will scrutinize millions of tweets by analyzing linguistic markers in social media posts. It creates a deep learning model that can give a unique insight into their mental health far earlier than traditional approaches. Individuals, parents, guardians, and medical professionals can evaluate social media posts for linguistic signals that reflect declining mental health using the system's algorithm, which can scan tweets that exhibit self-assessed depression traits.

II. Dataset

The dataset is available on Kaggle. It consists of 15 million tweets that are labeled as tweets that indicate depression or tweets that do not indicate depression. The dataset is unbiased. We have roughly the same number of tweets in each of the two classes.

```
In [116]: messages.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1578612 entries, 0 to 1578611
Data columns (total 4 columns):
 #   Column           Non-Null Count    Dtype
---  ------           --------------    -----
 0   ItemID           1578612 non-null  int64
 1   Sentiment        1578612 non-null  int64
 2   SentimentSource  1578612 non-null  object
 3   SentimentText    1578612 non-null  object
dtypes: int64(2), object(2)
memory usage: 48.2+ MB
```

Fig A. Analyzing Twitter dataset

```
In [110]: sns.countplot(messages['Sentiment'])
C:\Users\Sushant\anaconda3\lib\site-packages\seaborn\_decorators.py:36
arg: x. From version 0.12, the only valid positional argument will be
eyword will result in an error or misinterpretation.
  warnings.warn(
Out[110]: <AxesSubplot:xlabel='Sentiment', ylabel='count'>
```

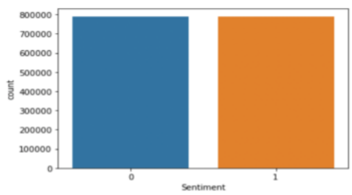

Fig B. Checking for class Biasness

III. Proposed Methodology

We propose building the system on binary classification technique of sentiment classification techniques. In binary classification technique each document di in D where D = {d1, d2, d3, …. dn} are classified into category C where C = {Positive, Negative} i.e. {Depressed, Not Depressed}. In our proposed methodology, there are four main components that are preprocessing, feature extraction, meta learning and training data.

A. Preprocessing

We've implemented the below preprocessing techniques to clean and treat the data:

1) Remove all URLs, hashtags, target words like (@username)
2) Correct the spellings; sequence of repeated characters is to be handled
3) Remove all the emoticons
4) Remove all punctuations, symbols, numbers
5) Remove Stop Words
6) Remove Non-English

Tweets *B. Data Wrangling*

1. *Tokenization:-* It is the process of breaking a stream of text up into words, symbols and other meaningful elements called "tokens". It's done this way so we can examine the tokens as independent components of a tweet.
2. *Lemmatization:-* Lemmatization usually refers to executing things correctly using a vocabulary and morphological analysis of words, with the goal of removing inflectional endings solely and returning the base or dictionary form of a word. For example, in our case a word such as beautiful will be converted to its root form which is beauty.

3. *Stop Word Removal:-* Stop words are a group of extremely common words that, when employed in a text, provide no extra information and are thus considered useless. "A," "an," "the," "he," "she," "by," "on," and so on are examples.
4. *Text Vectorization:-* Text Vectorization is the process of converting text into numerical representation. To feed in the data to a Machine Learning or Deep learning Model, we have to convert our text data to a numerical representation for our Model to train on.
5. *Bag of Words:-* The bag-of-words model is a simplifying representation used in natural language processing and information retrieval (IR).

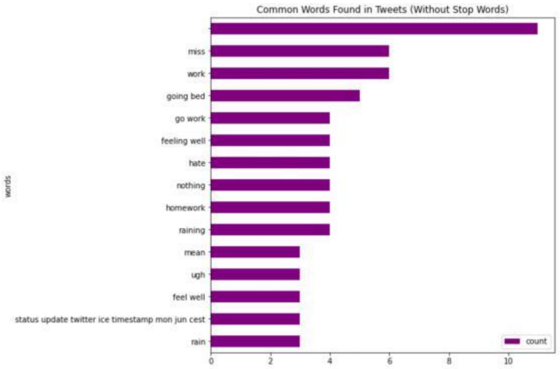

Fig C. Count of common words found in tweets

C. Model Selection

a) Multinomial Naïve Bayes – The classification algortithm is a Bayesian learning approach popular in Natural Langauage Processing. The algorithm predicts the tag of the name i.e {Depresed or Not Depressed} using the Bayes theorem. It calculates the likelihood of each tag in the sample dataset and maps it to the class with the greatest probability. When using the Naïve Bayes classifier, each feature being classed is unrelated to the other features. A feature's existence or absence has no bearing on the inclusion or exclusion of another feature.

b) Logistic Regression **-** Logistic regression is a statistical model that uses a logistic function to represent a binary dependent variable in its most basic form, though there are many more advanced variants. It's used to find out the correlation between one dependent binary variable and one or more nominal, ordinal, interval, or ratio-level independent variables by describing the data.

c) Artificial Neural Network – This algorithm has been implemented on the Twitter dataset with two dense hidden layers. The first layer is massaged with the Rectified Linear Unit (ReLU) activation function and the second layer with sigmoid activation function. The piecewise linear activation function of ReLU is a piecewise linear function. If the input is positive, it will be output directly; otherwise, it will be output as zero. It is the most commonly used and defualt activation function for most neural networks because the model that makes use of the ReLU activation function is in most cases, easier to train and often achieves better model accuracy. The vanishing gradient problem is overcome with the rectified linear activation function, allowing models to train faster and perform better.

We utilize the sigmoid function in the second layer since it exists between the first and second layers (0 to 1). As a result, it's especially useful in models that require the probability to be predicted as an output.Because for the likelihood of anything only occuring between 0 and 1, sigmoid is the best option. It is possible to differentiate the function. That is, the slope of the sigmoid curve may be found at any two places. Although the function is monotonic, the derivative is not. The hyperbolic tangent activation function cannot be used in networks with many layers due to the vanishing gradient problem.Thus we make use of a combination of ReLU and sigmoid activation functions.

The neural network uses the adam optimization algorithm which is basically used instead of the classic stochastic gradient descent procedure to update network weights iteratively based on the training data.

D. Data Visualization

On performing the intensive preprocessing and wrangling techniques on the dataset, it makes it easier to run the machine learning models and the accuracy increases by multifolds. On visualizing the dataset, split over tweets that indicate depression and those that do not indicate depression it is evident that the tweets that indicate depression has words like "work", "miss", "want", etc that do indicate some negative feelings.

Fig. D Word Cloud Visualization for tweets that **do not** indicate depression

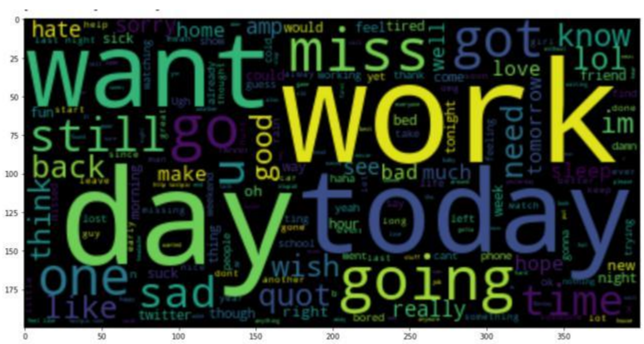

Fig. E Word Cloud Visualization for tweets that indicate depression

E. Accuracy Metrics

To test the performance of the algorithms we make use of four accuracy metrics like precision, recall, f1-score, and support. We've also closely monitored the rate at which the loss reduces which has assisted in selecting the best algorithm.

1) *Accuracy* – The most intuitive performance metric is accuracy, which is defined as the ratio of accurately predicted observations to total observations.

2) *Precision – It is a proportion of positive identifications was actually correct.*

3) *Recall – Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes. The question recall answers is: Of all the tweets that truly indicated depression, how many did we label correctly?*

4) *F1- score* – The weighted average of Precision and Recall is the F1 Score. As a result, this score considers both false positives and false negatives. Although it is not as intuitive as accuracy, F1 is frequently more useful than accuracy, especially if the class distribution is unequal. When false positives and false negatives have equivalent costs, accuracy works well. It's best to look at both Precision and Recall if the cost of false positives and false negatives is considerably different.

Out[34]: <AxesSubplot:>

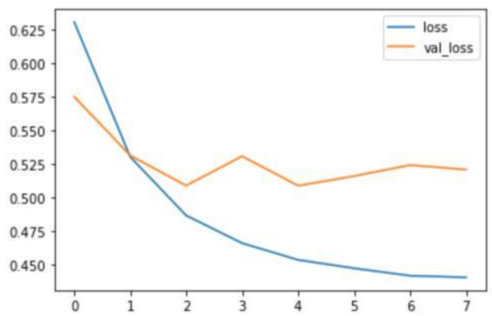

Fig F. Plotting the gradual decrease in the training data loss and validation loss

```
[[3637 1348]
 [1129 3886]]
              precision    recall  f1-score   support

           0       0.76      0.73      0.75      4985
           1       0.74      0.77      0.76      5015

    accuracy                           0.75     10000
   macro avg       0.75      0.75      0.75     10000
weighted avg       0.75      0.75      0.75     10000
```

Fig G. Confusion Matrix and Classification Report for Logistic Regression.

```
[[3676 1309]
 [1235 3780]]
              precision    recall  f1-score   support

           0       0.75      0.74      0.74      4985
           1       0.74      0.75      0.75      5015

    accuracy                           0.75     10000
   macro avg       0.75      0.75      0.75     10000
weighted avg       0.75      0.75      0.75     10000
```

Fig H. Confusion Matrix and Classification Report for Multinomial Naïve Bayes.

```
[[3541 1444]
 [1605 3410]]
              precision    recall  f1-score   support

           0       0.69      0.71      0.70      4985
           1       0.70      0.68      0.69      5015

    accuracy                           0.70     10000
   macro avg       0.70      0.70      0.70     10000
weighted avg       0.70      0.70      0.70     10000
```

Fig I. Confusion Matrix and Classification Report for Artificial Neural Network.

	Model	Accuracy
0	LR	0.7523
1	NB	0.7456
2	ANN	0.6951

Fig I. Comparing different algorithms and their accuracy

CONCLUSION

We set out to find words that we use to predict the negative sentiment, which can signify depression. These tweets must be looked at holistically concerning patterns of behavior over time and whether positive sentiments are expressed as well as coexists with negative sentiment. By analyzing tweets and experimenting with model parameters such as thresholds, scores, number of tweets and themes, and user selection, we can generate a list of words with associated scores that are statistically more likely to be used by persons who are depressed when used in tweets. We want to experiment and examine the results using stress, positive sentiment, and relaxation in future work. An exciting adaptation is to analyze sentiment by processing emoticons.

REFERENCES

[1] Banitaan, Shadi, and Kevin Daimi. ”Using data mining to predict possible future depression cases.” International Journal of Public Health Science (IJPHS) 3.4 (2014): 231-240.

[2] Hussain, Jamil, Maqbool Ali, Hafiz Syed Muhammad Bilal, Muhammad Afzal, Hafiz Farooq Ahmad, Oresti Banos, and Sungyoung Lee. ”SNS based predictive model for depression.” In International Conference on Smart Homes and Health Telematics, pp. 349-354. Springer, Cham, 2015.

[3] T. Al-Moslmi, N. Omar, S. Abdullah and M. Albared, "Approaches to Cross-Domain Sentiment Analysis: A Systematic Literature Review," in IEEE Access, vol. 5, pp. 16173-16192, 2017.

[4] R. Mehra, M. K. Bedi, G. Singh, R. Arora, T. Bala and S. Saxena, "Sentimental analysis using fuzzy and naive bayes," In Proc. of the 2017 Int. Conf. on Computing Methodologies and Communication (ICCMC), Erode, 2017, pp. 945-950.

[5] S. Liu, X. Cheng, F. Li and F. Li, "TASC:Topic-Adaptive Sentiment Classification on Dynamic Tweets," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 6, pp. 1696-1709, 1 June 2015.

[6] Braithwaite, Scott R., et al. ”Validating machine learning algorithms for twitter data against established measures of suicidality.” JMIR mental health 3.2 (2016).

[7] Xiaohui Tao, Ravi Dharmalingam, Ji Zhang, Xujuan Zhou, Lin Li, and Raj Gururajan “Twitter Analysis for Depression on Social Networks based on Sentiment and Stress.”

[8] Anastasia Giachanou and Fabio Crestani. Like it or not: A survey of twitter sentiment analysis methods. ACM Comput. Surv., 49(2):28:1–28:41, June 2016.