Detecting Depression using Tweets

Anonymous ACL submission

Abstract

Detecting mental illness in the field of social media can be a very difficult task since the definition of these mental illness can very from person to person. Over 300 million people of all ages suffer from depression worldwide [WHO CITATION].WHO classifies depression broadly into 3 categories: mild, moderate and severe. Depression can be caused due to Stress, anxiety, past traumas and can cause serious physical illness which is detrimental in the long run. In this project we trained two types of classifiers to classify tweets as depressive or not. We used data from twitter pages related to depression and labeled them as 1(depressive tweet) and extract tweets from Sentiment 140 data set for the non depressive tweets. Using this data set, we trained our algorithms to predict if a given user has depression or not. We achieved an accuracy of 99% test set accuracy for classifier 1 which is used to predict if there is depression content in the tweet, and 80% test set accuracy for classifier 2 which is used to classify if the given user is at risk of depression.

1 Introduction

Nowadays, Twitter and other Online Social Media are being used by people suffering from depression to express their pain and sufferings. With the rise in usage of social media, more and more people tend to share information about their personal life and express their feeling openly on social media. This paper proposes a machine learning classification model to predict whether a user has depression by analyzing the tweet of the user.

2 Dataset

The dataset that has been collected for this project consists of Tweets extracted directly from Twitter with help of Tweep, a python library that allows us to retrieve tweets from a page. For the positive examples, we retrieved Tweets containing keywords related to depression from twitter pages that tweet such messages. (Eg: depression quotes, Depression Notes and Damn Depression) and filter out the negative examples. We also collected tweets from various github pages[1]. For negative examples, we extracted random tweets from Sentiment 140 dataset, which dont include any examples of depression. Hence, those tweets have been marked as 0. The tagging was done in a way such that the tweets where the user is under depression or the were tagged as 1 and others were tagged 0.

Tweet	Class
”Tbh I am feeling very depressed rn”	1
”I was depressed a few years back !”	0
”Depression is becoming serious nowadays and harming many lives”	0

Table1: Few tweets and their annotated tags.

3 Prepossessing

For cleaning the data, we used the underlisted methods:

1. Ekphrasis[2]: Ekphrasis is a tweet tokenizer library for python which was used since this Tokenizer aws specifically designed for tweets, and tokenizes them more accurately than any other tweet tokenizer.

2. Removed non utf characters from tweets as they were also irrelevant.

3. Tweets that contained words not belonging to English dictionary have also been removed

4. We used Tweeboparse[3] for POS- Tagging of our tweets since this POS Tagger has more accurate results while tagging tweets and also helps in handing slangs and emoticons.
5. links and images
6. Remove @ mentions and hashtags - Regex
7. Emoticons were converted to english descriptions (:) - ¡happy¿)
8. Remove stop words

   Normalize words by stemming/lemmatization (e.g. ran -¿ run)

   Remove url, emails, currencies, numbers, users, date and time - not useful. Also unpacked contractions (youll - you will) and corrects spelling mistakes in elongated words (sooooo).
9. Decision Trees
10. Multinomial Naive Bayes

4 Our Methods

4.1 Baseline Methods

We ran 2 classifiers for classification of our data, Comprising of classifier 1 and classifier 2, where we are first partitioning our data based on 'depression content' in the tweet, and in the later stages, we are predicting if the tweets' author has or is undergoing depression.

4.1.1 Classifier 1

The main objective of classifier 1 is to filter the tweets based on depression content, we used data set 1 for this task, where we have labelled the data based on the depression content of the tweet. The tweets marked as 1 are passed on for classification as depressive or non-Depressive in the classifier 2. We used the following models for classsifier 1:

1. Linear SVM
2. Decision Trees
3. Multinomial Naive Bayes

In these Classifiers, we have first segmented the tweets in tokens, and then ran the WordNet Lemmatizer present in the NLTK library on the tokenized tweets. We extracted the features from the tweets using TfIdf Vectorizer for unigram and bigram models and limited the number of maximum features to avoid overfitting. The optimal hyper parameters for the model are decided using GridSearchCV for optimal test Results.

4.1.2 Classifier 2

The main objective of classifier 2 is to classify a given tweet as depressive or non-depressive, based on the features extracted from the TfIdf Vectorizer with the n-gram range of uni-grams and bi-grams. In these baseline methods, we did not consider the sentiments of the emotes present in the tweets, and they were removed during the pre-processing stage of the tweet. We also did not consider the slang words present in the tweet as they were a noise for the wordnet and were removed during the pre-processing stage as well. The classifiers used for the task were:

1. Logistic Regression
2. MultiNomial Naive Bayes.
3. Linear SVM
4. RBF SVM
5. Decision Trees
6. Random Forests
7. K Nearest Neighbours
8. Gradient Boosting
9. AdaBoost

For the optimization of the classifier, we ran GridSearchCV for the values of parameters that maximize the accuracy on the test set.

4.2 Novelty Method

For the novelty method, in addition to the classifier 1, we also used multiple hand-crafted features in our classifier 2, Also we used a list of slang words, which are later used as one of the features in our classifier 2. In this novelty method, we also used a different POS Tagger which is specifically trained for Tagging tweets since tweets contain many words, emoticons and abbreviations (eg. IKR for I Know Right, and FB for FaceBook) which normal NLTK Tagger is not able to tag correctly. This is a java-based Classifier that works as a dependency parser. We used the following handcrafted features for our classification:

1. Sentiment of Slang Words: We classify the sentiment of the slang words as Positive, Negative, and Neutral (Eg. -1 for Negative(STFU- Shut The Freak Up), 0 for Neutral(FYI- For your Information) and 1 for Positive(LOL-Laugh Out Loud)). This feature can be a representative feature of the mood of the user, that is if the user has more depressive and harsh tone in his/her tweet or joyous and happy vibe.

2. Personal Pronoun: We created a list of personal pronouns, which we noticed occurred frequently in the tweets, for example, terms like 'I', 'me', 'my', etc occured very frequently when a user expressed his/her feelings while in depression, or told his/her activities and actions while they are in depression. The presence of this feature in the tweet is indicative of a depressive tone in the tweet.

3. Count of Second and Third person pronouns - We used the counts of words like 'he', 'she', 'our' and other third person pronouns as they can be indicative of the fact that the user is either addressing the fact that they have depressive tendencies, or they are talking about the people with them that have depressive tendencies, which can lead to their thought also being depressive.

4. Count of proper nouns- We used this feature for basically filtering out a tweet that a user has tweeted and just talks about someone else's depression. Since this is indicative of the fact that this person themselves do not have depression, high value of this parameter can lead the model to classify the tweet as negative.

5. Count of positive and negative present tense verbs in the tweet- These are 2 features, where we have created 2 lists containing count of positive present tense verbs and negative present tense verbs. The sentiment of these verbs were extracted using the Wordnet's Sentiwordnet library, which gives scores for the positive, negative and the neutral sentiment of the word. we use the sentiment which has the highest value. This feature was also used for present tense verbs. Since a negative verb can be indicative of the fact that if the activity mentioned in the tweet is negative, then most probably the tweet also has a negative sentiment, ehich helps us in confidently classifying the tweet. For example: activities such as suicide, cut, etc are indicative of a depressive tweet.

6. Similar to the count of positive and negative verbs, we have also stored the counts of Adjectives having positive and negative sentiments using Wordnet's Sentiwordnet. Since a positive or negative adjective is also a indicator of the mood of the user, positive adjectives such as 'beautiful', and 'good' are indicative of a non-depressive tweet, while adjectives like 'sad', 'unhappy' are representative of a depressed user.

7. Sentiment values of the past-tense-verbs, present-tense-verbs, adjectives and nouns: These values are used to indicate the degree of positivity or negativity in the tweet and helps in classification of the overall sentiment of the tweet.

8. Count and sentiment of positive and negative emoticons: This feature was again used to help in classification of the overall sentiment of the tweet. The emoticons were first converted to a text form, using which we can extract the sentiment of the emoticon. More positive emoticon counts and overall positive emoticon content are significant of a positive mood from the user, while a high count of negative emoticons, and higher negative emoticon sentiment is representative of a depressive tweet

9. Counts of tokens with all caps alphabets, emphasis words and elongated letters were also taken as possible features. Their occurence may indicate mood of a user.

We combined these 26 handcrafted features with the features extracted from our n-gram models, and used all the combined features for the classification task.

4.3 Results

Using handcrafted features, maximum increase of 8 percent was seen in LinearSVM, Decision Trees and Random Forests in classifier 2. Maximum accuracy of 81.09 was achieved in case of LinearSVM and Random Forests.

Table 1: Training and Testing Accuracy Results (in %) (For classifier 2 the bracket scores denote scores for novel classifier)

Models	Training	Testing	F Score	Classifier1	Classifier2
MultiNomial Naive Bayes.	97.81	97.41	96.50	Yes	
Linear SVM	99.76	99.64	99.52	Yes	
Random Forests	99.77	99.65	99.52	Yes	
Logistic Regression	99.78(83.59)	73.94(79.63)	73.51(74.54)		Yes
RBF SVM	86.35(88.88)	73.4(78.9)	73.30(77.7)		Yes
K Nearest Neighbours	78.75(85.14)	68.40(77.8)	67.80(0.7)		Yes
MultiNomial Naive Bayes.	99.52(96.44)	74.81(80.73)	74.28(76.89)		Yes
Linear SVM	96.13(99.72)	73.82(81.09)	73.73(78.27)		Yes
Decision Trees	99.85(99.90)	65.25(73.45)	65.50(70.43)		Yes
Random Forests	99.18(99.90)	73.18(81.09)	72.50(75.56)		Yes
Gradient Boosting	86.35(92.16)	73.39(80.36)	73.309(77.35)		Yes
AdaBoost	82.69(95.53)	70.79(75.27)	70.53(71.92)		Yes

4.4 Conclusions and further improvements

This paper presents a novel approach to diagnose depression in twitter users. The accuracy on the given models can be further improved by correcting some misclassifications in the dataset. Also using User level information can provide a wholesome idea about the user's depression. Another way the accuracy can improve is by taking multiple tweets from the same user for depression classification.

5 Acknowledgments

We would like to thank Prof. Tanmoy for supporting us to do this project.

6 References

[1] https://github.com/AshwanthRamji/Depression-Sentiment-Analysis-with-Twitter-Data
[2] https://github.com/cbaziotis/ekphrasis/tree/master/ekphrasis/utils
[3] http://www.cs.cmu.edu/ ark/TweetNLP/owoputi+etal.naacl13.pdf