Predicts Suicide Ideation Based on Tweets

Yee Zhian Liew, Sithara Krishna Murthy, Reetika Goel
Professor Vijay Eranti
Department of Software Engineering
San Jose State University, USA

Abstract—**Suiciding has become a major issue in our world. It is very hard to detect suicidal ideation from the real world. Online social network such as Twitter has become a platform for suicidal ideation detection. How do we detect suicidal intention of a user in Twitter? Text classification is a large research area with major challenges. The main point on how to detect suicide tweets are the words suicide people commonly used while expressing themselves. Words that implies the warning sign or key features when they have the thinking of take their life. In this paper, we are presenting a new approach that uses the twitter platform to detect for specific suiciding post which contains suicide-related-sentences or common words. We are trying to detect the sentiment of the tweets first and continue investigating for suicide-related post. To start with our model, we use 2 different dataset and uses 2 classification ways to improve our output. We use various machine learning algorithm with n-gram consideration to train with our dataset. Furthermore, we use deep learning model such as LSTM and CNN. Experiments show that our text-scoring approach effectively captures the sign of suicidal post in text compared to baseline machine learning classifiers approaches. Our CNN + LSTM model shows the best accuracy of 90%. Although it seems like 90% is good enough, but we have used validation method to check whether our model experiencing any under fitting or over fitting. Additionally, we use clustering algorithm to check out which area or which type of users are more likely to have suicide ideation. [1]**

Index Terms— **Twitter tweets, NLTK, Deep Learning, text classification, social network, Machine Learning**

I. Introduction

According to (WHO), close to 800,000 people die due to suicide every year. [2] This indicates that there is 1 person dying in every 40 seconds. Suicide is being the top10 highest leading cause of death occurs throughout globally. Followed by the suicide statistics from American Foundation of suicide foundation, suicide costing $69 billions in each year. [3] For the people between age 55 to 64, it shows that it has the highest rate of 19.72. Most people that attempt to suicide are white, followed by American Indian, Asian. As we can see, it is hard to predict that someone is trying to suicide or not. Understanding the ways on how individuals communicate their suicidality is key to preventing deaths. Suicidality is defined as any suicide-related behaviour, thoughts or intent, including completing or attempting suicide, suicidal ideation or communication. [4] Suicidal ideation is defined as thoughts about killing oneself, while suicidal behaviours involve acts of self-harm with the intention of causing death. While maybe not all the individuals expressing suicidal ideation, or will or planning to end their life, having this thought will definitely increase the possibility of suicidal. Some people might ask or talk to their friends about suiciding, mostly they will think that it will not happen. Recently, individuals have expressing their suicidal Ideation in Twitter by using sentences such as "I don't want to live anymore, I want to kill myself". [5] Suicidal tweet leads to negative sentiment. First, we need to figure out how to define negative tweets. Figure (a) shows the example of positive tweet whereas figure (b) shows the example of negative tweet.

(a)

(b)

II. BACKGROUND OF TWITTER

Twitter is a free online social network which registered user can post whatever ideas, thoughts on their news. Users can also use hashtag words or tagging people in their post for visibility. Twitter application can be used on any electronic devices such as laptop, cellphones, desktop. Twitter user basically can tweet anything, anywhere, anytime as long as there is internet connection. Approximately 25% of online adults use Twitter and over 500 million tweets are sent per day [6]. Furthermore, twitter post can be call from twitter API. Twitter API is an interface to retrieve Tweets in real time. We are using tweepy tool which can be used to call the post. By using tweepy, we need to get the authentication to login into Twitter developer platform. First, you will need a twitter account to start. After that, twitter will provide a specific token for user to call any API the users have set up. We can increase our features such as location, date, time and userID. Furthermore, we can also set the limits of post to be called. In our projects, we are just focusing on twitter text, user and location.

III. TEXT SENTIMENT ANALYSIS

Sentiment analysis or opinion mining is one of the famous research areas in computer sciences. The application of sentiment analysis is very broad and powerful especially in text mining. It can also be an essential part of your market research and customer service approach. For example, Amazon used sentiment analysis to analyze the comment in terms of text. It is very powerful as it can detect even emojis whether it is positive or negative. Based on sentiment analysis, sales not only can see what people think of your own products or services, you can see what they think about your competitors.

Sentiment analysis can be classified into three diverse as sentence level, document level, and entity-aspect level. In a sentence level, a supposition of specific sentence is considered as a priority for sentiment prediction. Whereas, document level is a more generalized feeling which considers the whole document for sentiment prediction. And if the focus is straightforwardly on the opinion itself then it can be termed as an entity-aspect level sentiment analysis. As databases is getting larger and vaster, without machine learning, it is very hard to extract the interesting or important information. Machine learning algorithm such as Naïve Bayes, Logic Regression, and Support vector machine for predicting class for sentiment problem. Model is train from the train dataset given and passing various of algorithm and calculating the model accuracy. Classification accuracy is the major issue. This gives a motivation for acquiring a good classification precision picking great feature determination, preprocessing along with order procedures.

IV. ARCHITECTURE

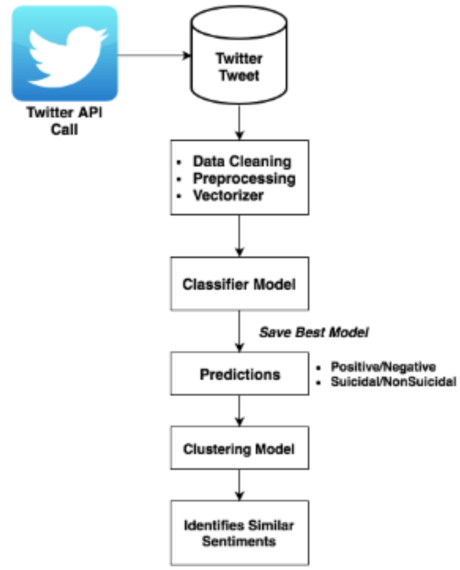

Fig.1. Flow Chart of sentiment analysis

V. DATASET

We have collected data from one of the Kaggle competition which contains 1.6 million tweets. [7] The larger the dataset, the better for training our model. This dataset contains the most important features that we wanted, Sentiment. It has 3 main sentiment, positive, negative and neutral. For our case, we suicide ideation will fall into negative categories. Beside of sentiment, we have collected 2000 dataset with suicide label on it. It is hard to collect such dataset and so collect latest data using Twitter API and find 3 to 10 people to comment whether the tweets are related to suicide post or not.

VI. FIRST CLASSIFICATION

For the first classification, we are using 1.6 million tweets with 2 different sentiments listed. In this dataset, we are using host the twitter tweet and the sentiment column.

VII. PREPROCESSING

Preprocessing of the data is a basic technique for data mining purposes. It uses to transform a raw data into structure format of data which can be understand by us. In real world, dataset can be obtained easily from Data warehouse to World Wide Web. It is often consisting of unstructured or non-consistent data type of dataset. Preprocessing is the first steps in data mining because without it, we cannot start our analyzing skill on data we do not understand. In our project, our data contains noises such as null values and columns that we do not need. Our data need to goes through data cleaning to fill up or removes all the null values. If null values are more than expected, we can remove it. If the null values are less, we can fill it up. Next, data might have different data type in the same columns. Some tools cannot perform well if the data contains various data type. Our data need to pass into data integration by converting to the data type that we want. After that, data needs to be normalized or generalized in order to be fair will doing the analysis.

Basic steps are done with our text dataset. With this dataset, sentiment analysis tools might perform very poorly because the text pre-processing step is not fully done. In order to get a good result, our raw text data needs to be clean up using method such as filtering and converting. The very first cleaning method would be converting upper case word into lower case word. After that, non-English word and stop word are basically needing to be remove from the text. The lesser the non-important words, the better our mode's prediction. Natural language processing toolkit (NLTK) has contains a well-prepared document of stop words such as "a", "an", "the" and non-English words to use for text cleaning. It is the most used tools for text mining problem. [8] Then, unwanted punctuations or special character removal is likewise executed as a piece of preprocessing strategy. From this step, most of the unwanted words is done with cleaning, such as "RT". Twitter has contained a lot of "Retweet or RT" post and it is not necessary to be in our consideration. The next step would be stemming or lemmatizing. Basically, stemming is removing word with suffixes, which ends with -ed, -ing, -ness. For example, 'created" will convert into "create". Word stem will reduce the number of unique word and thus reduce the dimensionality of data. Lemmatizing method is to convert word back to its root words. For example: "happily" will be convert into "happy". Also, we can apply n-gram function into it. Set of n-grams can make out of consecutive words. Negation words such as "no", "not" is attached to a word which follows or precedes it. [8]

VIII. TFIDF

Term frequency-inverse document frequency or text feature extraction is a statistic calculation on pinpointing how important a word is to a document. In another way it is using to judge the topic of an article by the words it contains. Relevance is the key measurement for TF-IDF but not frequency. First, TF calculates the total number of the same word that appears in a document. Since there will be word which is not important such as "and", "or", they are systematically discounted and that is what IDF does. TF-IDF intend to keep the important words for analyzing purpose and remove words that seems to be useless.

No doubt that this term is often used in text mining because it can extract each word as a string for easier calculation purposes. TFIDF vectorizer is used to transforms text to feature vectors that can be used as input to estimator. We can analyze it 4 for 2 ways: character and word. Basically, it will be comparing each characters or word in the document and find out which is appearing the most.

$$W_{i,j} \times log\frac{N}{df_i}$$

$tf_{i,j}$ = number of occurrences of i in j

df_i = number of documents containing i

N = total number of documents

Fig.2. Mathematical Calculation of TFIDF

IX. COUNT VECTORIZER

The Count Vectorizer provides a simple way to both tokenize a collection of text documents and build a vocabulary of known words, but also to encode new documents using that vocabulary. [9] First, we will create a count vectorizer class. Then, use the fit function to learn the vocabulary from various documents. Lastly, use the transform function to encode each word from various documents as a vector. While words transform becoming vectors, there will be a lot of zeros which we call them sparse. In this case, we will use to array function to transform them back to NumPy array, so we can understand easier.

X. SECOND CLASSIFICATION

When performing second classification, we are using the second dataset which contains tweets and 2 labels: potential suiciding post and non-potential suiciding post. We have applied the same preprocessing techniques and TFIDF vectorizer as the first classification.

XI. MACHINE LEARNING ALGORITHM

Machine learning and recurrent neural network deep learning are two the types of classification approaches. There are mainly 2 types of ML algorithm: Supervised learning and Unsupervised learning. Supervised learning means that the data is given a classify 'class' for it to refer and predict the output for other unseen data. Some of the useful supervised learning algorithm will be Decision Tree and Naive Bayes classifier. For Unsupervised Learning, the dataset does not contain any class label for the model to refer. In short, the algorithm is trying to find the interesting pattern itself according to what and numbers of features are given. Usually, unsupervised learning is related to clustering and association problem. Since we are dealing with classification problem, we are using algorithm such as random forest, KNN fit and test our dataset. To improves our model, we are trying to use Gradient Boosting and XGB classifier since they are claimed to be the best classifier compare to the others.

A. Random Forest

This algorithm is an ensemble learning method for classification, regression that operate by constructing multiple decision tree. For example: creating few random subsets for the problem statement. For each subset, it will design as each of the decision tree. Assuming there is a new subset need to be classified and the new subset will be throwing into the decision trees. [10] Each decision trees will provide an output and the highest output will be as the new subset's class.

	precision	recall	f1-score	support
0	0.61	0.64	0.63	41073
4	0.63	0.59	0.61	41427
avg / total	0.62	0.62	0.62	82500

Fig.3. RF Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.85	0.97	0.91	345
Potential Suicide post	0.92	0.67	0.78	183
micro avg	0.87	0.87	0.87	528
macro avg	0.89	0.82	0.84	528
weighted avg	0.87	0.87	0.86	528

Fig.4. RF Classification Report for Suicidal

B. Decision Tree

A decision tree will contain leaves and nodes which branch out. The internal nodes are tests and whose leaf nodes are categories. Basically, each of the attributes will be tested on each internal node. [10] Then, each branch from the internal node will select the best attribute. Best attribute is selected according to maximum information that can be extracted, known as information ratio.

	precision	recall	f1-score	support
0	0.61	0.59	0.60	151790
4	0.60	0.62	0.61	151810
avg / total	0.60	0.60	0.60	303600

Fig.5. DT Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.87	0.87	0.87	345
Potential Suicide post	0.75	0.75	0.75	183
micro avg	0.83	0.83	0.83	528
macro avg	0.81	0.81	0.81	528
weighted avg	0.83	0.83	0.83	528

Fig.6. DT Classification Report for Suicidal

C. Gaussian NB

A Gaussian Naive Bayes algorithm is one of the special types of NB algorithm. Specifically, feature with continuous value It's specifically used when the features have continuous values. For applying this method, all features are assuming to use a Gaussian distribution, for example: normal distribution. We are just using the basic Gaussian NB without adding any parameters. It's also assumed that all the features are following a gaussian distribution such as normal distribution.

	precision	recall	f1-score	support
0	0.50	1.00	0.67	151790
4	0.51	0.00	0.00	151810
avg / total	0.50	0.50	0.33	303600

Fig.7. Gaussian NB Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.68	1.00	0.81	345
Potential Suicide post	1.00	0.09	0.17	183
micro avg	0.69	0.69	0.69	528
macro avg	0.84	0.55	0.49	528
weighted avg	0.79	0.69	0.59	528

Fig.8. Gaussian NB Classification Report for Suicidal

D. Bernoulli NB

Bernoulli NB is one of the NB algorithms which split into multi variate or independent. The binomial model is useful if your feature vectors are binary is either 0 or 1. One application would be text classification with a bag of words model where the 0s 1s are "word occurs in the document" and "word does not occur in the document".

	precision	recall	f1-score	support
0	0.53	0.29	0.37	151790
4	0.51	0.74	0.60	151810
avg / total	0.52	0.51	0.49	303600

Fig.9. Bernoulli NB Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.68	1.00	0.81	345
Potential Suicide post	1.00	0.11	0.20	183
micro avg	0.69	0.69	0.69	528
macro avg	0.84	0.55	0.50	528
weighted avg	0.79	0.69	0.60	528

Fig.10. Bernoulli NB Classification Report for Suicidal

E. Gradient Boosting

For any supervised learning algorithm is to define what is the loss function and trying to minimize it to the minimum. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models. Basically, it learns a simple regression predictor of our data. Then, it computes the error residual in our prediction. Lastly, it will learn a new model to try to predict this error residual. This step can keep on going by adding more and more predictor in order to get a desire result.

	precision	recall	f1-score	support
0	0.72	0.68	0.70	151790
4	0.70	0.74	0.72	151810
avg / total	0.71	0.71	0.71	303600

Fig.11. Gradient Boosting Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.88	0.91	0.90	345
Potential Suicide post	0.82	0.77	0.79	183
micro avg	0.86	0.86	0.86	528
macro avg	0.85	0.84	0.85	528
weighted avg	0.86	0.86	0.86	528

Fig.12. Gradient Boosting Classification Report for Suicidal

F. XGB

XG Boost which is stand for extreme gradient boosting is an implementation of gradient boosted decision trees designed for speed and performance. XGB is a higher level of boosting algorithm which push the limit of computation on tree algorithm. Some key feature algorithm that we need to be aware: sparse aware, block structure, continued training. Sparse aware is the power of automatically handle missing values. Block structures is means that it supports tree construction parallelization. Continue training is it can further analyze or boost the already fitted model into a new data. [11]

	precision	recall	f1-score	support
0	0.70	0.66	0.68	151790
4	0.68	0.71	0.69	151810
avg / total	0.69	0.69	0.69	303600

Fig.13. XGB Classification Report for Sentiment

	precision	recall	f1-score	support
Not Suicide post	0.86	0.89	0.88	345
Potential Suicide post	0.78	0.73	0.75	183
micro avg	0.84	0.84	0.84	528
macro avg	0.82	0.81	0.81	528
weighted avg	0.83	0.84	0.83	528

Fig.14. XGB Classification Report for Suicidal

XII. DEEP LEARNING ALGORITHM

Deep Learning approach or Deep analysis learning is a every high-level machine learning to solve either supervised, semi-supervised or unsupervised problem. Recurrent Neural Network (RNN) is a class of AI which connects a bunch of neurons in a sequence. The basic idea is feeding it some input data, it will start processing using expensive algorithm to find out its predicted output. What RNN is different from another traditional machine learning is that, it can detect the hidden layer in the dataset. This is increasing the accuracy of model effectively since more data is feeding in. RNN is very useful in finding solution for sequences dataset such as video frame or image data. A very famous use cases would be word prediction. It is referring to the first input layer, then second and hidden layer and so on to predict what should be the next word. In every neurons layer, it contains input and its own specific weight value, which we call it as parameter. Weight is constantly updating to be larger or smaller when passing by gradient back-propagation phase. When the weight matrix is smaller than 1.0, it will be causing the gradient signal to become weak and learning model will either slow down or stop. Conversely, with larger weight matrix will cause high gradient signal and model learning path will be diverging. Thus, vanishing gradient problem and exploding gradient problem are the 2 main problems for RNN.

XIII. LSTM INTRODUCTION

This is where Long Short-Term Memory (LSTM) is introducing into it. LSTM is a unit of Recurrent Neural Network which can be added into the model to deal with the 2 main difficulty. [12] Behind LSTM model it has a memory cell structure. Basically, this cell contains 4 very crucial components: input gate, a neuron with a self-recurrent connection, forget gate and output gate. Self-recurrent connection neuron is to maintain the weight matrix to be 1 as this point. Every gate will be having interaction with the cell. For input gate, it will choose or block the input signal into the respective neurons in each state. Forget gate will have the power to adjust the cell to remember or forget the previous state. At last, the output gate can allow or block the input signal to have effect on the neuron in the next state.

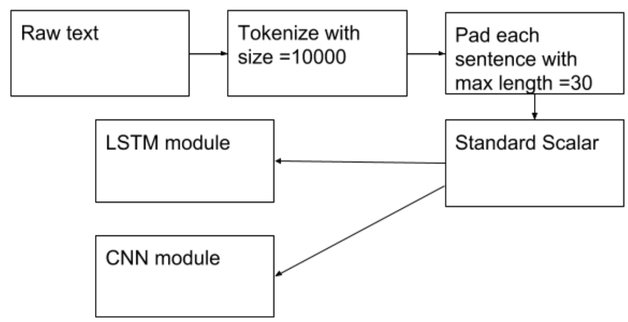

Fig.15. Structure of Deep Learning model with text classification

A. 1 layer of LSTM model

We use just the basic 1 layer of LSTM model with parameters of: 10 epochs, 100 batch size, SoftMax activation, LSTM = 32, categorical cross entropy.

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.54	0.52	0.52	0.25	0.52	0.51	0.51	0.51	0.50	0.50
accuracy	0.59	0.68	0.71	0.71	0.72	0.72	0.73	0.73	0.73	0.73

Fig.16. LSTM with categorical Cross Entropy for Sentiment

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.54	0.50	0.50	0.49	0.48	0.47	0.46	0.46	0.41	0.40
accuracy	0.71	0.74	0.75	0.76	0.76	0.77	0.77	0.76	0.78	0.80

Fig.17. LSTM with categorical Cross Entropy for Suicidal

B. 1 layer of LSTM model

We use just the basic 1 layer of LSTM model with parameters of: 10 epochs, 1000 batch size, SoftMax activations = 128, binary cross entropy. This time, we increase the batch size and change the loss into binary.

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.37	0.33	0.33	0.32	0.32	0.31	0.31	0.31	0.28	0.27
accuracy	0.84	0.78	0.79	0.80	0.81	0.82	0.82	0.82	0.83	0.83

Fig.18. LSTM with binary Cross Entropy for Sentiment

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.35	0.33	0.32	0.31	0.31	0.28	0.28	0.28	0.27	0.26
accuracy	0.60	0.68	0.68	0.68	0.77	0.78	0.80	0.80	0.80	0.81

Fig.19. LSTM with categorical Cross Entropy for Suicidal

C. 2 layer of LSTM model

We updated into 2 layer of LSTM model with parameters of: 10 epochs, 1000 batch size, SoftMax activation, both LSTM =128, binary cross entropy.

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0363	0.33	0.32	0.31	0.30	0.29	0.28	0.26	0.26	0.26
accuracy	0.81	0.81	0.82	0.83	0.85	0.85	0.85	0.86	0.86	0.87

Fig.20. 2-layer LSTM for Sentiment

Epoch	1	2	3	4	5	6	7	8	9	10
loss	054	0.49	0.47	0.45	0.44	0.39	0.37	0.35	0.35	0.34
accuracy	0.71	0.75	0.76	0.78	0.79	0.79	0.80	0.81	0.81	0.83

Fig.21. 2-layer LSTM for Suicidal

D. CNN + 2 LSTM

We updated into 2 layer of LSTM model and adding a CNN model into it. We set parameters of: 10 epochs, 1000 batch size, pool size =4, SoftMax activation, both LSTM =32, binary cross entropy.

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.40	0.37	0.35	0.33	0.33	0.29	0.25	0.23	0.21	0.20
accuracy	0.79	0.83	0.84	0.84	0.86	0.87	0.89	0.90	0.90	0.90

Fig.22. CNN + 2 layer of LSTM for Sentiment

Epoch	1	2	3	4	5	6	7	8	9	10
loss	0.57	0.49	0.47	0.47	0.33	0.29	0.25	0.23	0.21	0.20
accuracy	0.68	0.75	0.84	0.84	0.86	0.87	0.88	0.88	0.89	0.89

Fig.23. CNN + 2 layer of LSTM for Suicidal

Model	Accuracy	F1	Recall	Precision
Decision Tree	0.60	0.60	0.60	0.60
Random Forest	0.53	0.53	0.53	0.53
Gaussian	0.49	0.33	0.49	0.49
Bernoulli	0.51	0.48	0.51	0.51
Gradient Boosting	0.65	0.64	0.64	0.64
XGB	0.68	0.68	0.68	0.68
1-layer LSTM/cate	0.73	0.71	0.72	0.72
1-layer LSTM/binary	0.83	0.82	0.82	0.79
2 layers LSTM	0.87	0.85	0.87	0.87
CNN+ LSTM	0.90	0.89	0.89	0.90

Fig.24. Result from all baseline and NN model for Sentiment

Model	Accuracy	F1	Recall	Precision
Decision Tree	0.82	0.83	0.83	0.83
Random Forest	0.87	0.87	0.87	0.86
Gaussian	0.79	0.79	0.69	0.59
Bernoulli	0.78	0.79	0.69	0.60
Gradient Boosting	0.86	0.86	0.86	0.86
XGB	0.83	0.83	0.84	0.83
1-layer LSTM/cate	0.80	0.80	0.81	0.80
1-layer LSTM/binary	0.81	0.82	0.82	0.80
2 layers LSTM	0.83	0.83	0.83	0.83
CNN+ LSTM	0.90	0.89	0.89	0.89

Fig.25. Result from all baseline and NN model for Suicidal

XIV. Word Cloud

Word cloud is a very powerful python tool for visualization. In our project, we have extracted positive and negative sentiment to examine what kind of words are used commonly. The larger or more visible words, which means the frequency of the word used is higher. As it shown below, the most common words for positive text are 'love', 'thank' and 'good', great. However, negative sentiment text contains words such as, 'hell', 'damn', 'hate' and 'bad'.

Fig.26. Most common used words in positive tweets

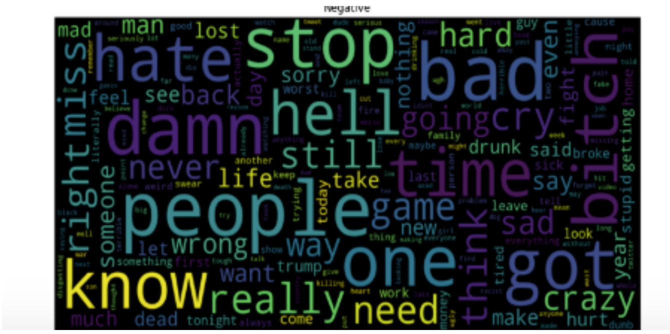

Fig.27. Most common used words in negative tweets

XV. Clustering

Clustering is an unsupervised learning technique. It is the process of grouping a set of objects such that objects in same cluster are more similar to each other than those in another cluster. A good clustering method will produce high superiority clusters with high intra-class similarity and low inter-class similarity. In this project, the data is first cleaned. Then from the place coordinates, the latitude and longitude are extracted. The records which are extracted are labelled with negative and potential suicidal and then clustering is performed for the models using latitude and longitude. In this model, the user with the similar sentiments are clustered, thus predicting the sentiment of the users based on the different locations in US and also helps to identify users with suicidal ideation, so that help can be provided to them. For this, we have used two clustering algorithms; K-means and DBSCAN.
K-means algorithm is used when we have unlabeled data. The goal of the algorithm is to find groups in the data, where K is represented as the number of groups or clusters. In this, the data points are clustered on the basis of feature similarity. Based on the features that are provided, each data point is assigned to one of the K groups. Here, users who have negative and suicidal potential tweets were clustered on the basis of their locations.

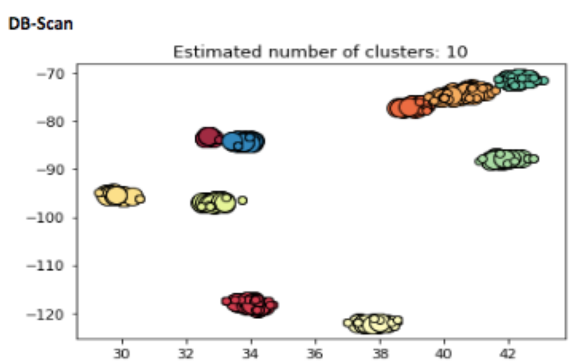

Fig.28. Cluster potential suicidal user based on location

XVI. Conclusion

We have tested using traditional Machine learning algorithm and comparing with result. From Machine learning perspective, XGB with n-estimator of 500 gives the most accuracy which is 68%. The second comes to gradient boosting with parameter of 400, it gives 65% of accuracy. The worst ML algorithm for our dataset is Bernoulli NB, which only have 49% accuracy. However, for suicidal classification, Random Forest is the best model which gives 87% accuracy. Furthermore, we have implemented deep learning LSTM and CNN model. For both classification, CNN + LSTM model with 15 epochs, 900 batches, SoftMax activation, gives us the highest accuracy which is approximately 90%, comparing to traditional ML algorithm.

XVII. References

[1] Vioulès, M., J. Azé, B. and S, B. (2018). *Detection of suicide-related posts in Twitter data streams - IBM Journals & Magazine*. [online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/8269767 [Accessed 4 Dec. 2018].

[2] World Health Organization. (2018). *Suicide across the world (2016)*. [online] Available at: https://www.who.int/mental_health/prevention/suicide/suicide prevent/en/ [Accessed 4 Dec. 2018].

[3] AFSP. (2018). *Suicide Statistics — AFSP*. [online] Available at: https://afsp.org/about-suicide/suicide-statistics/ [Accessed 4 Dec. 2018].

[4] O'Dea, B., Wan, S., Batterham, P., Calear, A., Paris, C. and Christensen, H. (2018). *Detecting suicidality on Twitter*. Detecting suicidality on Twitter.

[5] O'Dea, B., Wan, S., Batterham, P., Calear, A., Paris, C. and Christensen, H. (2018). *Detecting suicidality on Twitter. Emotion Detection in text*.

[6] Michiko, U., Kota, M, Yasuyuki.,S.(2018). *Twitter Celebrity Suicide: User Reaction to prominent suicide*.

[7] Kaggle.com. (2018). *Sentiment140 dataset with 1.6 million tweets*. [online] Available at: https://www.kaggle.com/kazanova/sentiment140\ [Accessed 4 Dec. 2018].

[8] Iosrjournals.org. (2018). *Data Preprocessing, Sentiment Analysis & NER On Twitter Data*. [online] Available at: http://www.iosrjournals.org/iosr-jce/papers/Conf.17014-2017/Volume-2/15.%2073-79.pdf?id=7557 [Accessed 4 Dec. 2018].

[9] Brownlee, J. (2018). *How to Prepare Text Data for Machine Learning with scikit-learn*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/prepare-text-data-machine-learning-scikit-learn/ [Accessed 4 Dec. 2018].

[10] Cornn, K. (2018). *Clickbait article detection using deep learning: these results will shock you*. [online] Cs230.stanford.edu. Available at:

http://cs230.stanford.edu/files_winter_2018/projects/6931206.pdf [Accessed 30 Nov. 2018].

[11] Brownlee, J. (2018). *A Gentle Introduction to XGBoost for Applied Machine Learning*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/ [Accessed 4 Dec. 2018].

[12] Cornn, K. (2018). *Clickbait article detection using deep learning: these results will shock you*. [online] Cs230.stanford.edu. Available at: http://cs230.stanford.edu/files_winter_2018/projects/6931206.pdf [Accessed 30 Nov. 2018].