DEEP LEARNING BASED DETECTION OF DEPRESSION

[1] Ashritha R Murthy, [2] Akshay Suryanarayan Hegde, [3] Aniket Kharad, [4] Ganesh S, [5] Manzoor Ahmed

[1] Assistant Professor, Dept. of Computer Science and Engineering, JSS Science and Technology University, Mysuru, India

[2][3][4][5] UG Students, Dept. of Computer Science and Engineering, JSS Science and Technology University, Mysuru, India

Abstract

With busy life, increasing daily activities and dawn of the pandemic like COVID-19 people tend to forget about their mental health, with these busy lives people find themselves with anxiety, stress, depression related problems. Depression is a mental illness which is the mixed feelings of sadness, feeling low, stressed, mood swings. According to WHO approximately 264 million people across all the ages suffer from depression. Patients of these mental illness generally turn towards social media sites like Facebook, Twitter, Reddit to share their feelings with the community and try to seek emotional support. This is the motivation to capture the user posts and identify the individual's psychological problems as early as possible. There are also chances people don't express themselves in social media sites. With our solution discussed in the paper it is difficult to reach out to them with medical guidance.

Keyword: *Depression, COVID-19, Twitter, Facebook, WHO*

1. INTRODUCTION

A widespread psychiatric illness is depression. It has a direct and indirect effect on economic development. Not only for those affected, but also for their families and their social and work-related environments, depression also has significant consequences. It can be the psychological basis for symptoms of panic and anxiety. Increasingly, panic disorder has been focused on health care and the media, impacting young people aged 20-40. Social media is increasingly used by different levels of age groups. Patients of psychiatric disorders also turn to online social media and web forums for specific conditions and emotional support information. Although social media can be used to transform the life of an individual as a really beneficial tool, it can create certain conflicts that can have a negative effect. The practices turn out to be extremely difficult with the growing number of users and their content. This motivates for detecting depression from the user posts. The main idea is to detect such psychological problems from user's posts as early as possible. This leads us to develop Deep Learning based Detection of Depression.

There are multiple ways to detect depression by using text data, audio data, video data, but text data proves to be of more significance as explosion of users in the social media sites connect various people and they tend to post pictures, comments on these sites. In this paper we are considering text data in order to detect depression in individuals.

The Text classification, also known as text categorization, is a classical problem in natural language processing (NLP), which aims to assign labels or tags to textual units such sentences, queries, paragraphs, and documents. It has a wide range of applications including question answering, spam detection, sentiment analysis, news categorization, user intent classification, content moderation, and so on. Text data can come from Different sources, including web data, emails, chats, social media, tickets, insurance claims, user reviews, and questions and answers from customer services, to name a few. Text is an extremely rich source of information. But extracting insights from text can be challenging and time-consuming, due to its unstructured nature. Text classification can be performed either through manual annotation or by automatic labeling. With The growing scale of text data in industrial applications, automatic text classification is becoming increasingly important.

Approaches to automatic text classification can be grouped into categories:

- Rule-based methods
- Machine Learning(data-driven) based method

In this paper, we aim at detailed research and analysis on detecting depression through text, from social media by applying machine learning and deep learning algorithms, we start with collecting data from various social media sources by mining texts from tools like TWINT. Then we perform data preprocessing by cleaning and converting raw text into required format which can be further fed into the machine learning which uses various machine learning algorithms and deep learning models as well. But the dataset being very huge, we propose deep learning algorithms on the processed data, which seems to be appropriate and serves the purpose of applying the model to a huge corpus. With tuned hyperparameters we architect deep layered models. Finally, we analyze all the algorithms through various evaluation metrics, make a comparative analysis, evaluate results to choose the best algorithm to detect the depression. Literature Review is discussed in Section 2 , Methodology is discussed in Section 3, Modelling and Analysis in Section 4 and Results and Discussion in Section 5.

2. LITERATURE SURVEY

According to Dey, Sharmishta, et. al., [1] more than 264 million people across the globe suffer from mental health conditions and depression, between 76% to 85% of lower- and middle-income groups don’t receive any treatment for depression. The author also mentions that since the dawn of the beginning of the internet people have begun to dump their thoughts on the Internet which may prove to be a very powerful tool for the diagnosis of a person’s mental health problems.

[1],[2],[3] have given a survey on various algorithms applied on depressions data which includes Machine Learning, Deep Learning, Data Mining etc. also mentioned

about various trends in the usage of Random Forest, Hidden Markov Model, Naïve Bayes models for the better analysis of the model. Dey, Sharmishta, et. al., [1] mentions 95% accuracy for Deep Learning models and 92.6% F1-score for Artificial Neural Networks.

Mandar Deshpande, Vignesh Rao [2] have proposed a concrete idea on data extraction and data pre-processing using Natural Language Processing. The model proposed incorporates SVM, Multinomial Naïve Bayes as the primary algorithms. Author has taken into consideration F1-Score, Precision and Recall as the accuracy measures. The results for Multinomial Naïve Bayes are 83% and Support Vector Machine is 79%.

Md. Rafqul Islam et. al., [3] proposed a model which uses various Machine Learning Algorithms like KNN, SVM, Decision Tree and variations of it. The model has made use of Ensemble learning techniques to increase the performance and Accuracy of the model. There was also a special mention about the tool called LIWC which is used to extract relevant data from Social Media Sites. In the feature extraction step, there is a thorough analysis of characteristics of the data.

Hao Guoa et. al., [4] proposed a model on the basis of “Resting state functional brain networks” which has been widely studied in brain disease, resting state functional brain networks were constructed for 38 major depressive disorders. The model has an average accuracy of 79.27% and 78.22% for SVM and Neural Network with RBF kernel respectively, with 28 features.

3. METHODOLOGY

Before building the model, it has to undergo various preprocessing, shuffling, feature extraction the sequential steps in Building the models are.

- Collection of Data
- Data Preprocessing
- Feature Extraction
- Applying Machine Learning Models
- Applying Deep Learning Models
- Hyper Parameter Tuning
- Result Analysis

3.1 Collection of Data

The data was mainly collected from Twitter using TWINT automation tool where we specified the date from which we want the tweets and specify a particular keyword based on which the entire twitter was queried and to look for such tweets and the same was extracted and stored in various directories based on the severity of the words chosen. The tweets were queried using the key word “depression”, “stress”, “anxiety”, “PTSD”, “trauma” etc.

The system Architecture is displayed in the following Figure 1.

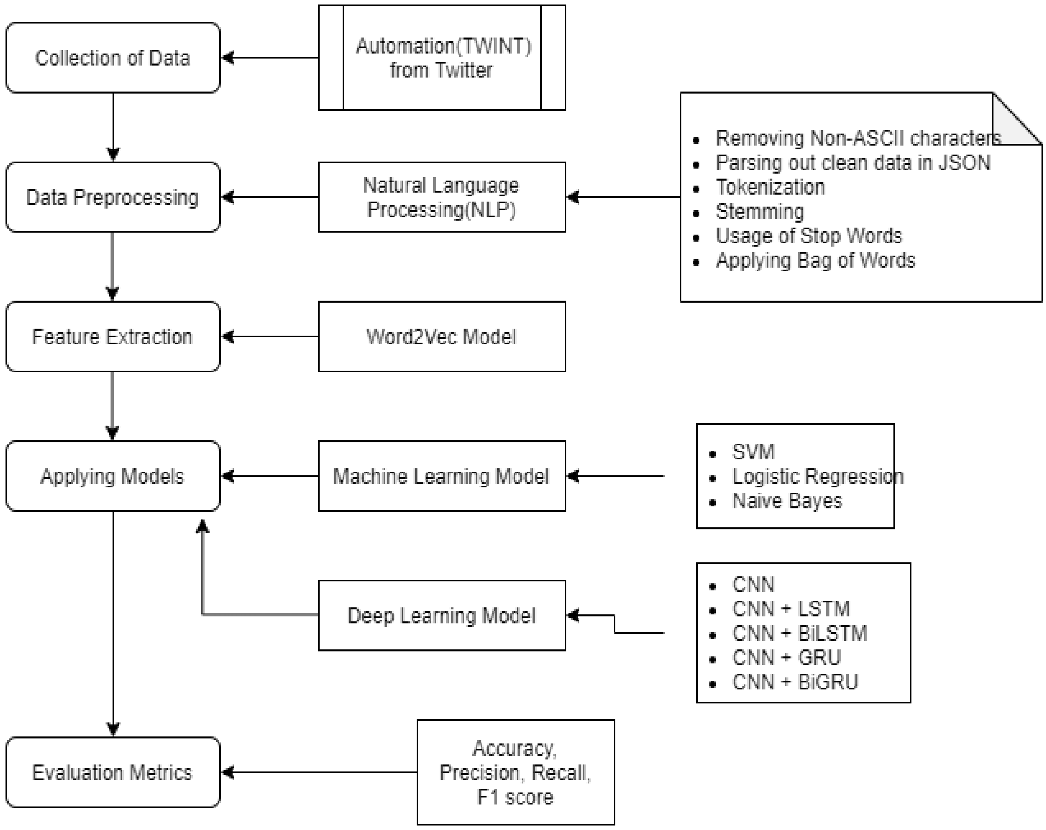

Figure 1. System Architecture

3.2 Data Preprocessing

A common text preprocessing technique is to remove the punctuations from the text data. This is again a text standardization process that will help to treat 'hurray' and 'hurray!!' in the same way and “Alas” and “Alas!”. Some of the punctuations include ‘!’,’$’, ‘@’, ‘*’ etc.

Stop words are commonly occurring words in a language like 'the', 'a' and so on. Basically, these words take up 20% to 30% of the word size and while processing it is difficult to index these. They can be removed from the text most of the time, as they don't provide valuable information for downstream analysis. In cases like Part of Speech tagging, we should not remove them as they provide very valuable information about the POS.

Stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form, for example, if there are two words in the corpus walks and walking, then stemming will stem the suffix to make them walk.

With more and more usage of social media platforms, there is an explosion in the usage of emojis in our day-to-day life as well. Probably we might need to remove these emojis for some of our textual analysis.

Next preprocessing step is to remove any URLs present in the data. For example, if we are doing a twitter analysis, then there is a good chance that the tweet will have some URL in it. We removed them for our further analysis.

3.3 Tokenization

Tokenization refers to giving index for each unique word present in the list of words so that we can feed it to the model. After tokenization we decide the width of matrix as 140 (which means 140 words) we pad the rest of the values in the matrix with 0 to maintain uniformity.

3.4 Feature Extraction

3.4.1 Word2Vec Model

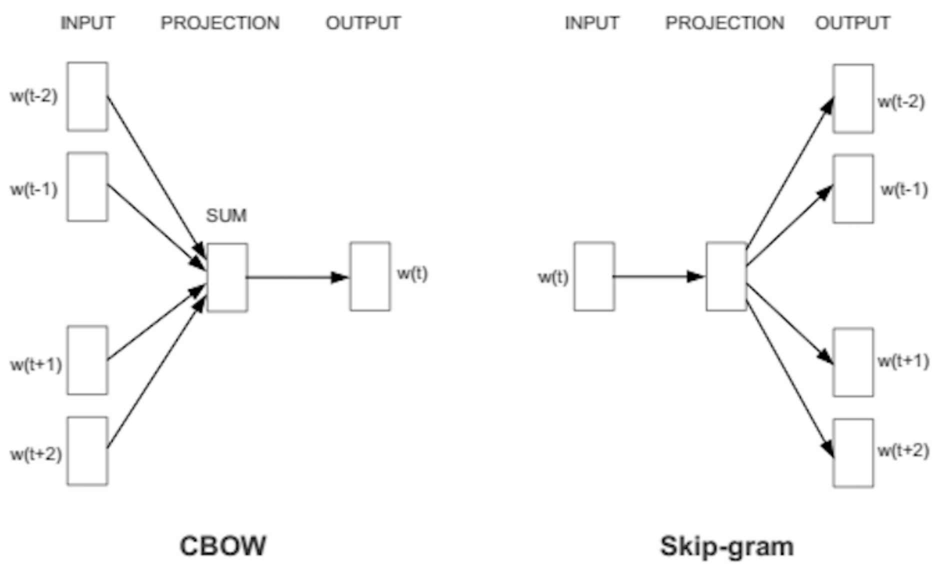

Figure 2 Word2Vec Model

Word2vec is a two-layer neural net that processes text by "vectorizing" words. Its input is a text corpus and its output is a set of vectors: feature vectors that represent words in that corpus. While Word2vec is not a deep neural network, it turns text into a numerical form that deep neural networks can understand.

3.4.2 GloVe (Global Vectorization) Model

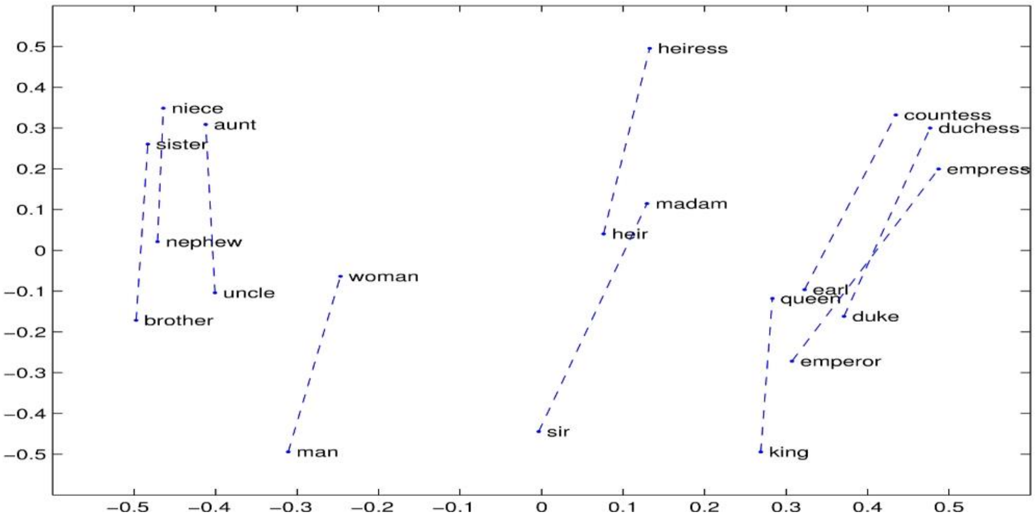

Figure 3 GloVe Model

GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.

The above models are used to extract features and build Embedding Matrix which can be used as weights in the Deep Learning Models. For Word2Vec Model we make use of Google News [7] pre trained models which has 300 features defined for each word, on the other hand for GloVe model we make use of [6] which has 100 features defined for each word. So, for each of the Embedding Matrix built from the model's performance of the model varies.

.

4. MODELLING AND ANALYSIS

4.1 Machine Learning Models

4.1.1 Naïve Bayes

Naïve Bayes is a simple but powerful method for predictive modelling. Naïve Bayes is a supervised Machine Learning Algorithm which is used majorly for classification purposes. Its performance is very good against Text data. It is based on Bayes theorem and with an assumption of independence among predictors.

$P(A/B) = (P(B/A) * P(A))/P(B)$

$P(A)$ = Priori Probability

$P(A/B)$ = Posterior Probability

4.1.2 Random Forest

Random Forests are supervised Machine Learning Algorithm for Classification as well as Regression. It is an ensemble learning technique, it operates by constructing multiple Decision Trees while training. Each Decision Tree comes up with the result of classification and the class with most votes becomes the model's prediction.

The decision trees in the Random Forest generally operate using Entropy and Information Gain (IG) as the factors. The decision tree constructs a tree with using set of attributes and moves towards the node with highest IG and lowest Entropy.

$$Entropy(D) = -\sum_{j=1}^{c} \Pr(cj)\, log2\Pr\,(cj)$$

4.2 Deep Learning Models

The deep learning models are designed on top of the Convolutional Neural Network (CNN) to get a hybrid version of models. Each of these Deep learning models are enhanced using Hyperparameter tuning, by applying dropouts, early stopping techniques, batch sizes, activation functions, optimizers. The feature extraction builds an embedding matrix which are given as weights in the models. The model has got various hidden layers each with convolutional layers, max pooling layer and dropouts.

4.2.1 LSTM

LSTMs are special kind of Recurrent Neural Networks (RNN) designed to avoid long term dependency problem. RNN's generally have chain of repeating modules. The key idea in the LSTM is cell state and four interacting layers. The memory part of LSTM is handled by the gates present in the LSTM Network.

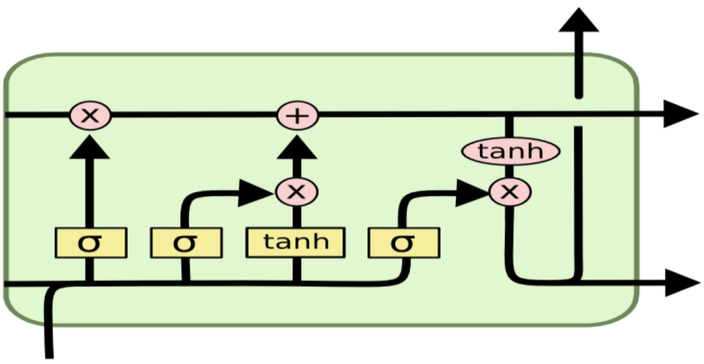

Figure 4 LSTM Network

Forget gate(f) - Determines to what extent to forget the previous data.
Input gate(i) – Determines information to be written into Internal state.
Output gate(o) – Determines what output to generate from current internal state.

LSTM solves the vanishing and exploding gradient problem of the RNN and it remembers all the past Knowledge and tend to forget the irrelevant data.

4.2.2 GRU

Gated Recurrent Networks were also invented to solve the vanishing gradient problem of the RNNs, Unlike LSTM it only has 2 gates.

Update Gate(z) – Determines how much of the knowledge has to be passed into the future. It is similar to Output gate in LSTM network.

Reset Gate(r) – Determines how much of the past knowledge to forget. It is combination of Input gate and Forget gate in LSTM network.

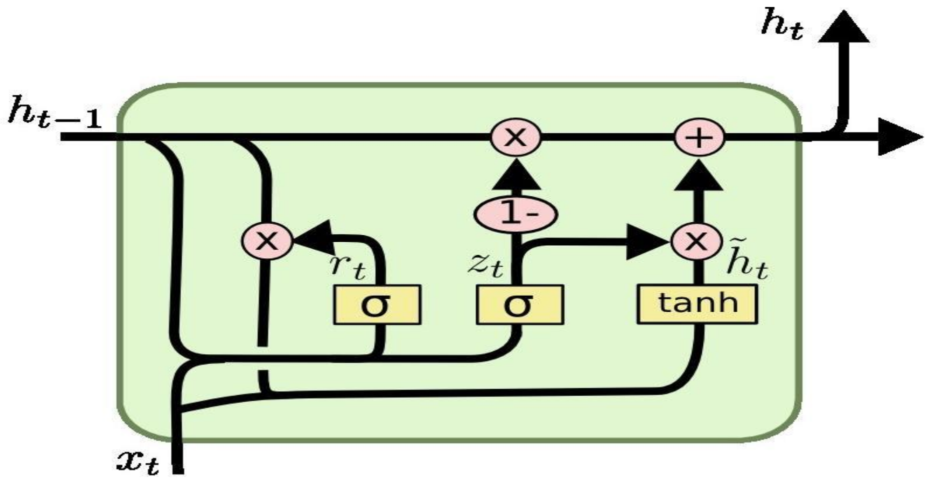

Figure 5 GRU Network

The Bidirectional versions of both LSTM and GRU are used for applying it to the models and also a basic CNN model was also applied on to the model.

5. RESULT ANALYSIS

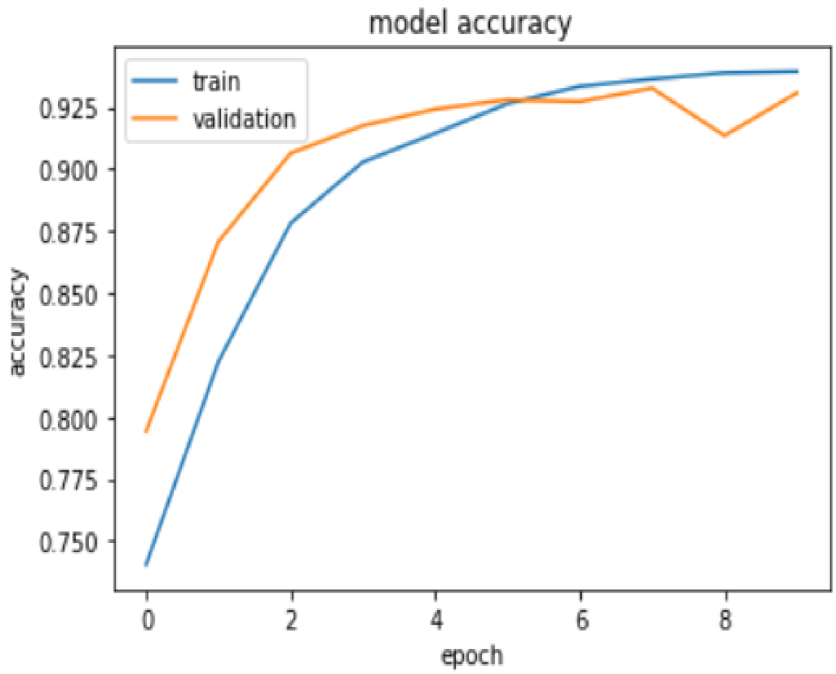

Figure 6. Accuracy vs Epoch

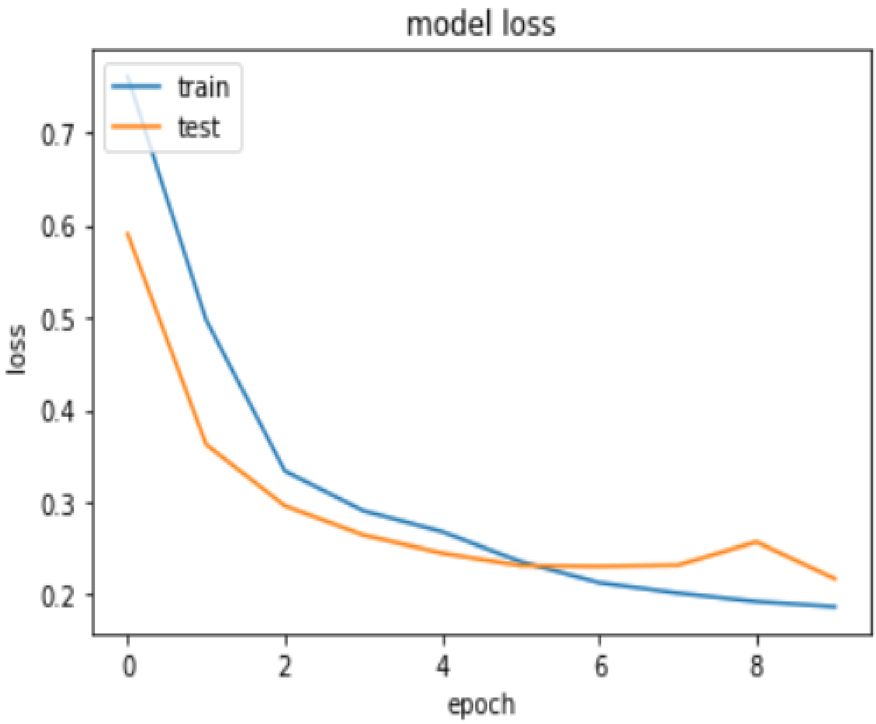

Figure 7. Loss vs Epoch

In the results discussed above we can see that the as the Epochs progress, we have loss decreases and the accuracy increases and they reach a certain saturation point. The graph can be seen as a smooth curve without too many variations because of the Hyperparameter tuning applied to the models.

Analysis and the results of all the models applied using Word2Vec and GloVe Model.

Table 1 Word2Vec Model

Model	Accuracy	Loss
Naïve Bayes	75.99	
Random Forest	85.28	
CNN	94.81	0.1683
CNN + LSTM	94.32	0.1736
CNN + BiLSTM	94.54	0.1763
CNN + GRU	94.81	0.1630
CNN + BiGRU	94.79	0.1729

Table 2 GloVe Model

Model	Accuracy	Loss
Naïve Bayes	75.99	
Random Forest	85.28	
CNN	95.12	0.1405
CNN + LSTM	96.36	0.1295
CNN + BiLSTM	96.63	0.1199
CNN + GRU	96.37	0.1295
CNN + BiGRU	96.48	0.1254

```
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

labels_pred = model.predict(data_test).argmax(axis = 1)
labels_pred = np.round(labels_pred.flatten())
accuracy = accuracy_score(labels_test, labels_pred)
cm = confusion_matrix(labels_test, labels_pred)
print("Accuracy: %.2f%%" % (accuracy*100))
```

```
Accuracy: 95.46%
```

```
[48] print(classification_report(labels_test, labels_pred))
```

```
              precision    recall  f1-score   support

           0       0.96      0.99      0.98      5977
           1       0.90      0.77      0.83       984
           2       0.96      0.93      0.94      1194

    accuracy                           0.95      8155
   macro avg       0.94      0.90      0.92      8155
weighted avg       0.95      0.95      0.95      8155
```

Figure 8 Classification Report

6. CONCLUSION

In this paper, we discussed a quantitative analysis of performance of the models and several machine learning algorithms on the different metrics considered. Finally, we discuss some of the open challenges and future research directions. Due to Covid-19, the number of people using social media has increased and we can see all that may result in depression because of people losing jobs and are in miserable conditions. By using this open-source data, we can get better results than previous models, and with the use of several Machine Learning Techniques such as Naïve Bayes, SVM, KNN we can improve the results. The hybrid Deep Learning model has also proven to be of considerable significance in terms of accuracy. The scope of the research done in this paper is to detect depression using various Text mining techniques and also it can be used for medical diagnosis consulting the experts in the field of medical sciences. The scope of this paper can also be extended in getting to know the state of mind of an individual. The models applied here are Machine Learning and Hybrid Deep learning models but more advanced and CPU intensive algorithms can be deployed, before that we can extract the features from the text using more sophisticated methods available like GloVe, TF-IDF and others and look out for the best methods to extract more meaningful insights from the text. With the explosion of data, it is available everywhere and people seem to express their part of the story in LinkedIn, Reddit, Facebook and many more. Look out for the API's available to get the user comments from these social media websites to work on the same and a quality research can be conducted with a variety of data in hand.

7. REFERENCES

[1] *DEY, SHARMISTHA, et al. "DEPRESSION DETECTION USING INTELLIGENT ALGORITHMS FROM SOCIAL MEDIA CONTEXT-STATE OF THE ART, TRENDS AND FUTURE ROADMAP."*

[2] *Deshpande, Mandar & Rao, Vignesh. (2017). Depression detection using emotion artificial intelligence. 858-862. 10.1109/ISS1.2017.8389299.*

[3] *Islam R, Kabir A, Wang H and Ulhaq A, Depression detection from social network data using machine learning Techniques, Islam et al. Health Inf Sci Syst,vol. 6, No. 8,pp:1-12,2019.*

[4] *Guo, Hao, et al. "Resting-state functional connectivity abnormalities in first-onset unmedicated depression." Neural regeneration research 9.2 (2014): 153.*

[5] *Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, Jianfeng Gao, for Deep Learning Based Text Classification: A Comprehensive Review*

[6] *https://wiki.pathmind.com/word2vec*

[7] *https://nlp.stanford.edu/projects/glove/*

[8] *https://code.google.com/archive/p/word2vec/*

[9] *https://colah.github.io/posts/2015-08-Understanding-LSTMs/*

[10]https://www.programmersought.com/article/15814471891/