TWITTER SENTIMENTAL ANALYSIS OF COVID TOWARDS MENTAL DEPRESSION USING DEEP LEARNING

Naseela Parvez , Aditya Suresh Sankaranarayanan

Department of Information Technology

SRM Institute of Science and Technology

SRM Nagar, Kattankulathur Campus

Chennai, INDIA

e-mail:pessuresh@hotmail.com

Abstract—The coronavirus pandemic hit the worldwide population to a large extent. But one of the subtle effects of the COVID-19 pandemic was the depletion of the mental health of the people. Social media has become an efficient platform to express oneself and Twitter is one of the most used platforms. There has been work where Machine and Deep learning were employed for Tweet Sentimental analysis for different applications including Mental depression. In the current scenario pertaining to COVID, there has been no work that focused on employing machine or Deep learning for predicting the mental depression from Tweets benefitting government for appropriate action. Also most of Tweets sentimental analysis were focused on positive and negative. There has been some research where neutral tweets were taken into consideration. We in this research work have focused on predicting depression of people i.e depressed, non-depressed, Neutral from tweets during lock down period by employing Deep Learning models like Bi-LSTM, BERT and XLNET. Also the BERT model been modified by adding classification layer for tweet classification as BERT is an NLP based model. In addition, the exploratory data analysis was performed for post lockdown tweets. This could prove useful for public as well Government and other agencies to understand the sentiment of people from Tweets for appropriate medication towards wellbeing of people.

Keywords: COVID-19, Twitter, Bi-LSTM, BERT, XLNET

1. Introduction

The coronavirus pandemic affected a considerable portion of the world's population and has been one of a kind in many ways. From the virus being new to mundane humans to spread like a wild forest fire, covid-19 is an unforgettable experience. The currencies of the most secure nations collapsed, lives were lost, and a more significant proportion of the people lost their employment. The secondary effects of covid-19 were depleting economies, unemployment, impacted food systems, etc. At the same time, the primary one has been on the mental health of people and also Individuals utilize online media locales like Facebook, Instagram, Twitter, and Reddit to share their considerations and perspectives. Details on Sentimental Analysis using Tweets been discussed in the following section.

1.1 Tweet Sentimental Analysis for Mental Depression

There has been work where Machine and Deep learning employed for Tweet Sentimental analysis for different applications. Also in terms of work related to mental depression, Bag of Words were used as features and support Vector classifier for classification of tweets. Also there has been work where deep learning models like CNN, LSTM applied for hate speech detection in Tweets. There has been other work pertaining to mental depression tweets where deep learning models like RNN and GRU models used. Details about these are discussed in literature review

It is seen that researchers have used lexical-based approach to do twitter sentiment analysis. There has been some considerable usage of deep learning and machine learning for Tweet Sentimental analysis for different applications including mental depression [1-6]. Also the data from Twitter are unlabeled where clustering strategies employed for labelling them. These strategies, however, are ineffective. In addition, most Twitter sentiment analysis are focused towards binary grouping. There has been limited work reported where neutral tweets considered and deep learning model applied like CNN, LSTM and not beyond that[7-11]. There have been no work where powerful models like BERT and XLNET not used for achieving higher accuracy. The use of hashtags to categorize content is standard practice. It is, however, not a real solution. A single tweet, for example, will contain several hashtags. Lastly there has been no much work found pertaining to COVID Tweet analysis towards mental depression except for one where the work [10] focused on hate speech classification only and employed CNN model towards tweet classification.

So based on the above mentioned drawbacks, we here have proposed employing deep learning models in analyzing the COVID Tweets towards mental depression which is need of hour. In addition, towards labelling the tweets, active learning approach was used. In addition, the deep learning models like Bi-LSTM, BERT and XLNET has been deployed towards multi classification of tweets as Positive, Negative and neutral for mental depression during COVID. Also the BERT model been modified by adding the classification layer for Tweet classification. The models were validated in terms of accuracy, error and loss for proposing the best deep learning model. Using the model that performed best which is BERT, we predicted the class of a small post-vaccination dataset into depressed, non-depressed and neutral. Following this, comparative analysis was carried for the two datasets lockdown and post lockdown to get an insight. This analysis of lockdown and post lockdown would ultimately result in reducing the number of suicide cases that have taken place because people cannot control their mental stress, loneliness, and hardship. Also there is a possibility towards arranging awareness camps to help people deal with depression and anxiety with proper tweet analysis. The contributions of this paper are as follows:

- Labelling the Tweets using Active learning approach and classification as depressed, not depressed and neutral using classifier.
- Deploy and Validate deep learning models Bi-LSTM, BERT and XLNET on mental depression using tweets pertaining to COVID-19 data during lockdown
- Validate the model deployed with best accuracy on post Covid-19 lock down period for classification of tweets and perform comparative analysis on lock down and post lock down data set.

The rest of paper is organized as follows.

2. Literature Review.

Twitter sentiment analysis has been carried out using the Machine Learning (ML) and Deep Learning (DL). ML models have been previously employed in twitter sentiment analysis. However, using deep learning models for determining the sentiments of tweets is new. In this section, the work that has used lexicon-based approach and ML been discussed

2.1 Sentimental Analysis using Machine Learning

In [1], the authors have tried to determine the mood of the public based upon the tweets that are made. The moods are categorized into six parts- tension, depression, anger, fatigue, confusion and vigor. Twitter sentiment analysis in this paper has been carried out based upon 'profile of mood states' which stands for POMS. However, generating a vector of the above six sentiments using POMS scoring limited and hindered the twitter sentiments. In reality, there are tweets that are happy and positive. Although the vector will have lower values of the labels but we do not get a proper insight of the sentiment represented in the tweet.

Depression campaign has prompted many twitter users to come forward and express their mental state. We are focused on Covid-19 pandemic related tweets. One such work that has been situation specific was discussed in [2]. The tweets of this work were focused on 'bell let's talk' campaign- a social awareness campaign for mental illness. For classification purpose, Bag of Words were used as features and the classifier used was SVC (Support Vector Classifier). The work also addressed the problem of imbalanced datasets. Oversampling technique- SMOTE- was applied to deal with class imbalance. If powerful deep learning models were employed, normalization of data is carried out effectively with the help of the loss functions. Machine learning models sometime outperformed deep learning models in case of twitter sentiment analysis. This can be attributed to the data not being cleaned properly.

In [3], the author has used the twitter data to get a customer review for the flight experience. Also, the complaints of the various airline companies have been jotted down. A strong feature extraction technique, TF-IDF has been used in the project. The work has used machine learning model called voting classifier. The classifier used logistic regression and stochastic descent classifier. The classifier has outperformed deep learning model LSTM. A final accuracy of 0.791 has been recorded in this work. This accuracy can be attributed to a smaller dataset.

2.2 Deep Learning for Sentimental Analysis

In the previous section we have discussed the various machine learning and lexicon-based approaches for twitter sentiment analysis. However, recently deep learning has been a new force in the world of technology. Deep learning has been utilized in many sectors for developing technology.

Binary classification problem is the most common problem for text data. In case of twitter sentiment analysis, we try to determine what the person thinks about an issue. The twitter sentiment analysis has focused on many problem statements. E.g. presidential elections, hate speech, fear, mental health etc. But these tweets are mostly categorized into two categories. This means that tweets are classified as positive and negative in most cases. However, in some research [9,11] neutral tweets have been taken into consideration. This means, that the tweets are not related to the problem statement. Although the tweets are considered into neutral categories as well, the models that are used for classification are limited to simple models.

In [4], an attempt has been made to identify racist tweets. It was done using the deep learning models like CNN and GRU. Also, the text embedding technique that has been employed in the paper is GloVe embeddings. This work aimed at predicting hate speech. It addressed the problem as binary classification. However, the twitter data might have tweets that are neutral with respect to hate speeches. The work has ignored considerably large neutral tweets.

In [5], researchers have used the hashtags for the labelling of a tweet. If a tweet has a hashtag ‘happy’, it is classified as a happy tweet and if a tweet has a hashtag ‘sad’ it is labelled as sad tweet. However, in many cases, a tweet has more than one hashtag. A single tweet can have both ‘happy’ and ‘sad’ as hashtags. And hence, the authenticity of the label is questionable. Also, the work has mentioned techniques like SVM, Naïve Bayes etc. for building the classifiers. Although machine learning has been used a lot in many sectors, deep learning is better with textual data because they have been used for many textual problems like sentence prediction, handwritings etc.

One of the papers that focused on using twitter data for depression was [6]. The paper used word based RNN and GRU models. The achieved accuracy for the model was 97% and 98% respectively. Although the model had high accuracy, as mentioned in the work, this was due to a smaller dataset of 13,385 rows. A smaller dataset has a smaller number of examples and thus might perform well on the data but will not work well with the other data. Also, the neutral tweets were ignored in this case

Authors in [7] have done hate speech detection in Tweets which were labelled as racist, sexist and neither. Deep learning models like CNN and LSTM been compared with CNN + Gradient Boosted Decision Tree and LSTM +Gradient boosted Decision. This resulted in LSTM with Random Embedding and Gradient Boosted decision tree having highest precision, recall, F1 score which is 93%.

Authors in this work [8] have combined CNN-LSTM integrated for hate speech detection of Tweets which were labelled as racist, sexist and neither. CNN is effective for feature extraction but CNN-LSTM does not have a bidirectional nature and hence the context of the text data cannot be understood completely. The work presented in the paper uses two type of embedding- one in which the weight is initiated manually on layers and second where pre-trained word embedding was used.

Class imbalance is a problem that we face in classification problems a lot of time. Authors in [9] has addressed the problem of class imbalance. In this paper, the minority class are up-sampled by randomly selecting from the class and adding them back to the data set. Both LSTM as well as Bi-LSTM are used for classification. The analysis has shown that LSTM outperformed Bi-LSTM. Tweets that were taken for hate speech detection are binary which is hate and non-hate.

Authors in [10] deal with COVID-19 tweets made in Arabic language towards hate speech detection which are labelled as hate and non-hate. Neutral tweets were not taken into consideration. CNN model has been used for detection of tweets. The paper has further used Non-Negative Matrix Factorization (NMF) for topic modelling to find the detailed distribution of the tweets. The classification problem addressed is binary. There were some limitations found in this work which are shorter study duration and also chances of misclassification of countries on account of self-reporting of user's locations.

Authors in [11] has implemented deep learning architecture for detection of hate word in Tweets. The models are Gated Recurrent Unit(GRU), Convolution Neural Network (CNN) and Universal Language Model Fine-tuning (ULMFiT) model, which is based on transfer learning technique. The performance analysis resulted in highest accuracy in ULMFiT which uses the concept of Transfer learning on publically available and newly created dataset.

So from the literature review, it has been found that there has been lot of drawbacks in the previously employed method using machine learning and deep learning for sentimental analysis which are:

•Twitter sentiment analysis has been carried out using the lexical based approach. The sentiment scores that are given by this approach are not reliable.

•Labelling of twitter data has been carried out using approaches like clustering previously. However, these approaches are not very effective owing to data distribution. Techniques like active learning have not been used.

•Most of the twitter sentiment analysis were classified as binary which are positive and negative. Very few papers have concentrated on the neutral twitter data.

•Classification on tweets based of hashtags has been a common approach. However, the hashtag does not define the sentiment of the tweet.

•Research on tweet classification have employed deep learning models like CNN, LSTM and some Bi-LSTM. Also basic machine learning models also used.

•The deep learning models have achieved good accuracy but models used are quite old and not state of art like BERT or XLNET for achieving higher accuracy.

•The research has not explored models beyond Bi-LSTM. Transformer based deep learning models that have higher performance with text data have not been explored.

•Lastly, the effect on Covid-19 pandemic on the mental health of people has not been targeted prior to this. Only paper that have focused on COVID-19 twitter have addressed on hate speech and not towards mental depression.

So based on the above-mentioned drawbacks, we here propose to perform tweet sentimental analysis based on COVID tweets using Deep learning models Bi-LSTM, BERT and XLNET. The deep learning models been validated in terms of accuracy, error and loss. Using the model that performed best which is BERT, we predicted the class of a small post-vaccination dataset into depressed, non-depressed and neutral. Following this, comparative analysis was carried for the two datasets- lockdown and post lockdown to get an insight. These would be discussed in detail in the forthcoming sections.

3. COVID Tweet Sentimental Analysis using Deep Learning

We in this work have aimed towards detecting the depleting mental health of people during the pandemic and also give an insight about how the mental health of people has improved post vaccinations and when the lockdown has been eased. To get an understanding of how the mental health of people has been affected during the COVID-19 pandemic, we concentrated on the twitter data.

We here focused on two major datasets - the tweets that were made during the pandemic and the tweets that were made post vaccination and when lockdown was eased. The first part of dataset collected was focused between the months of March to December 2020. The second dataset was a very small dataset collected for 15 March, 2021. The project here focused on identifying the depressed, non-depressed as well as the neutral tweets of the users. Secondly, identify the section of the population that has raised awareness about mental health using twitter platform. Finally develop a perspective about the tweets made in the month of March through December (when the pandemic was on peak) and how the mental health has been after vaccinations were started and the lockdowns were eased. The complete system architecture of the Tweet analysis for mental depression is shown below in Fig.1.

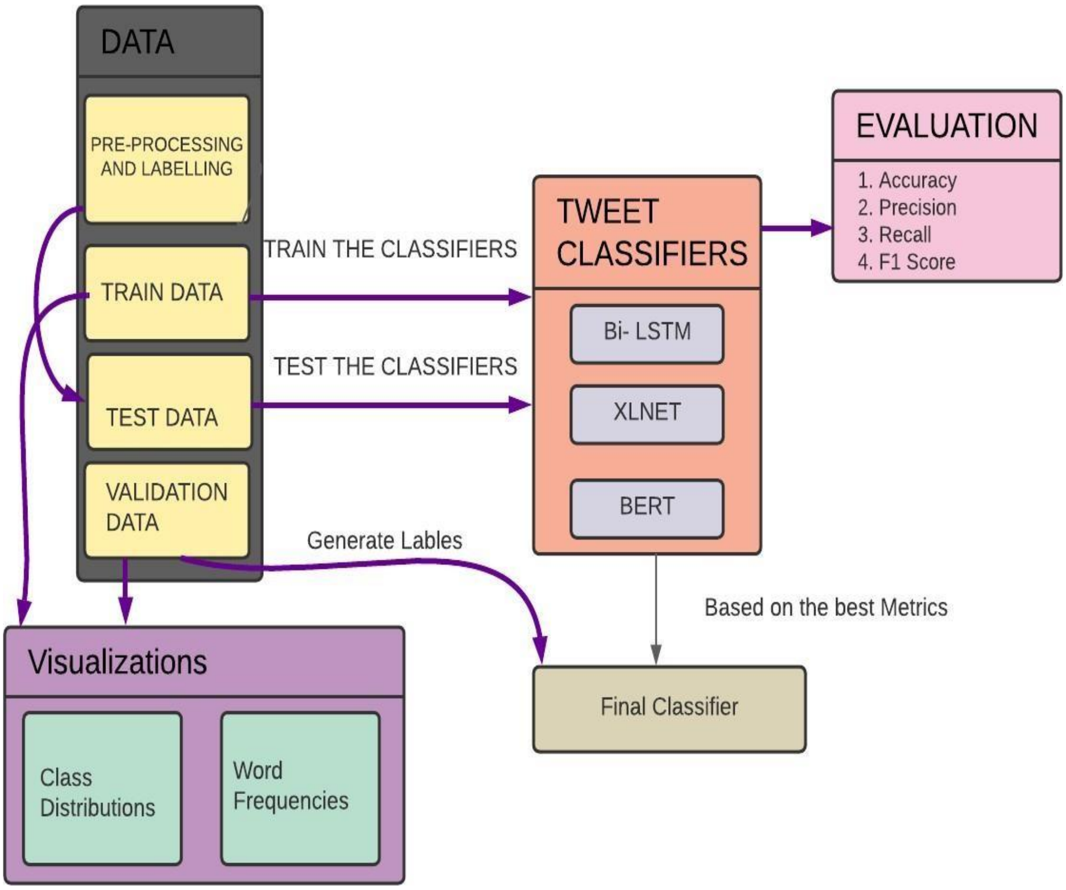

Fig. 1 System Architecture Diagram

The first step in Tweet Analysis using Deep learning model is data preprocessing [12] and Labelling. So for an effective classification of our tweets into the depressed, non-depressed and neutral category, we had to first pre-process our dataset.

3.1 Data Preprocessing

The first step for pre-processing [13] our twitter dataset was to include the tweets that were made in English language. After we had set of only English tweets, we further proceeded into removing anything in the tweet or ‘text’ column that could contribute to the noise for sentiment classification. After we filtered out only the English tweets, we proceeded to remove the stop words that contributed as noise. Further to that, stemming and lemmatization was done to link the word to their root words. Tokenization is another important part of text pre-processing that involves breaking a text into tokens ease the text-classification problem. Finally, the last step of pre-processing is dropping all the columns that did not contribute to the sentiment of our tweet. We only kept the ‘id’ and ‘text’ column for this purpose. Our final dataset consisted of two columns- ‘id’ for the id of tweet and ‘text’ for the content of the tweet.

3.1.1 Labelling the Twitter Data

To label the pre-processed dataset, “active learning” [15] has been employed. Active learning is a subset of the machine learning. The algorithm used in active learning interactively labels the unlabeled data. In active learning, a part of the dataset is selected and labelled manually. This labelled dataset is then required to label the rest of the dataset. However, the ‘text’ column containing tweets cannot be used by any classifier. Any classifier needs a vector or numerical data to work with. Therefore, labelling the dataset was divided into three major tasks –

1) Label the part of dataset
2) Creating the ‘text’ column to a vector and
3) Choose a classifier to classify unlabeled tweets

3.1.2 TF-IDF Vector

TF-IDF [16] stands for term frequency inverse document frequency. TF-IDF is a statistical measure for the relevance of a term in a document. It is a way to convert a text into a vector form. TF- IDF has an edge over other techniques because it considers term frequency as well as document frequency into consideration. The TF–IDF esteem builds relatively to the occasions a word shows up in the report.

Term frequency is the number of times a word will occur in the document. The inverse document frequency is number of times a word has occurred in a set of documents. If a word occurs many times in the document, the IDF approaches zero. A word that is very common and has occurred in many documents has TF-IDF approaching zero.

$$\text{TF-IDF}(t, d, D) = \text{TF}(t,d) * \text{IDF}(t,D) \qquad (1)$$

$$\text{TF}(t,d) = \log(1+ \text{frequency}(t,d)) \qquad (2)$$

$$\text{IDF}(t,D) = \log(N/df+1) \qquad (3)$$

df(t) = occurrence of t in document

The ‘text’ column was created into TF-IDF vectors. A new column ‘Vectorized’ was added to the dataset containing the TF-IDF vectors of the tweets. These were used to label the tweets

3.1.3 Support Vector Classifier

We have used the tf-idf vectors generated earlier for the purpose of classifying the tweets as depressed, non- depressed and neutral. The classifier uses this tf-idf representation to link a tweet to a category.

For classification, Support vector classifier (SVC) [16] has been used in this work. SVC was used as a part of active learning algorithm. Linear kernel was selected to separate the depressed, non-depressed and neutral data points. The classifier was trained using the manually labelled dataset and the unlabeled data points were predicted using the trained classifier. Fig.2 shows the hyperplane in SVC classifier separating different classes- depressed, non-depressed and neutral tweets

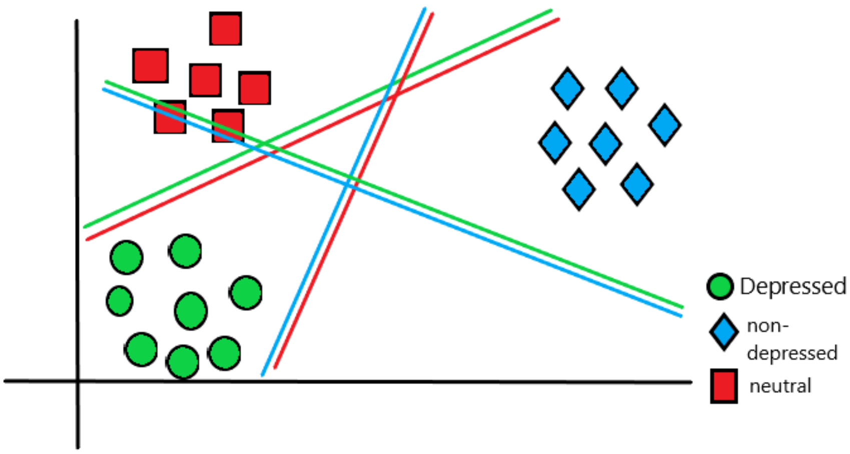

Fig.2 Hyperplane in SVC Classifier

3.2 Twitter Sentimental Analysis Using Deep Learning

Following the pre-processing stages and the labelling of our dataset, the problem of classifying tweets as depressed, non-depressed and neutral was addressed. In our dataset, we have labelled the tweets into three classes- neutral (0), depressed (1), non-depressed (2). We have employed three major deep learning models which are – Bi-LSTM, BERT, XLNET. The working as well as architecture of these algorithms have been described in detail in this section with respect to our Tweet dataset classification.

3.2.1 Bi-LSTM

LSTM stands for Long Short-Term Memory. LSTM can be defined as altered version of the RNN. LSTM is preferred for classification and prediction problems.

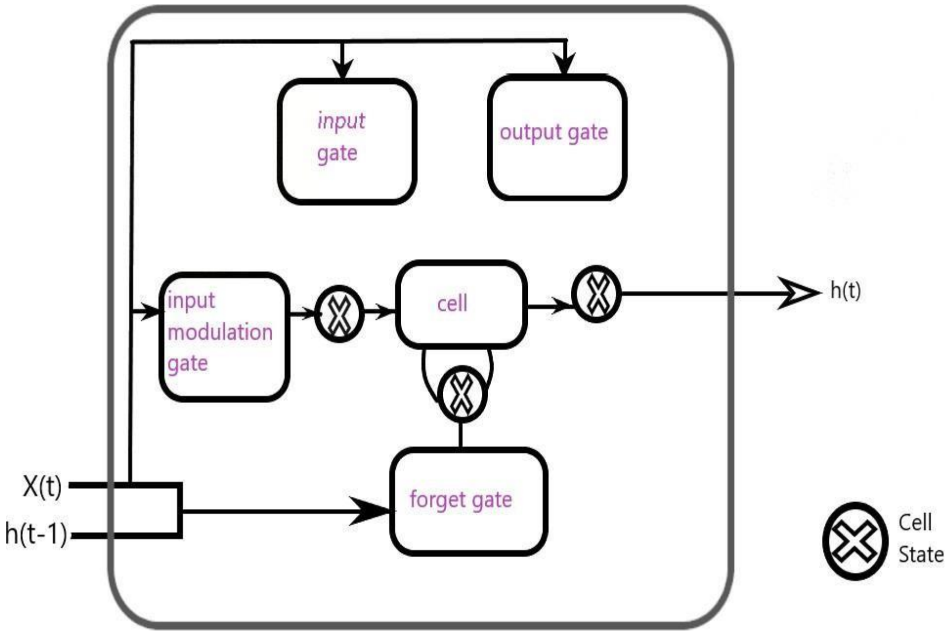

Fig.3 – Architecture of LSTM[17]

The architecture of LSTM as shown in Fig.3 [17] has cell state, input gate, forget gate, output gate and cell state. The cell state is the long short-term memory in the architecture. This means that all the information is stored in the cell state. The cell state is altered by the forget gate. Forget gate in other words is called the remember vector. When the result of the forget gate is 1, the information is to be retained. However, if the result is 0, the information is to be dropped. The input gate determines the information in the cell state. The input gate is nothing but an activation function. An input gate is a sigmoid function. The input gate decides which information can be added to the cell state. It cannot determine which information is to be dropped. The sigmoid function is then joined by a tanh activation function. The output gate gives the results of the cell state in a modified way. Below are the equations for the LSTM gates:

$$input_t = \sigma(W_{input}. [h_{t-1}, X_t] + b_{input}) \quad (4)$$

$$C_t = tanh(W_c . [h_{t-1}, X_t] + b_C) \quad (5)$$

$$forget_t = \sigma(W_{forget}. [h_{t-1}, X_t] + b_{forget}) \quad (6)$$

$$output_t = \sigma(W_{output}. [h_{t-1}, X_t] + b_{output}) \quad (7)$$

$$h_t = output_t * tanh(C_t) \quad (8)$$

where

h_t = cell state

$input_t$ = input gate

$output_t$ = output gate

$forget_t$ = forget gate

C_t = initial cell state

Bi-LSTM is simply combining two LSTM networks Bi-LSTM improves the sequence classification as compared to LSTM. This architecture provides way to have both forward as well as backward information about the sequence. We used Bi-LSTM in this work for Tweet Sentimental analysis

3.2.2BERT

BERT stands for Bidirectional Encoder Representations from Transformer. BERT is a transformer model that utilizes the transformer architecture of attention mechanism. The basic architecture of a transformer model consists of two units- encoder and decoder. An attention mechanism is a method of informing a system that particular words are important with respect to the output. In our case, e.g., sad is an important word with respect to detecting depression. This means that the word, ‘sad’ will be marked as a keyword. The attention mechanism also enables the decoder to look back at the sequences before giving an output. In case of BERT, an encoder passes the keyword to the decoder. This helps solving a sequence- to-sequence problem quicker and with much efficacy The BERT architecture [18] is shown in Fig.4

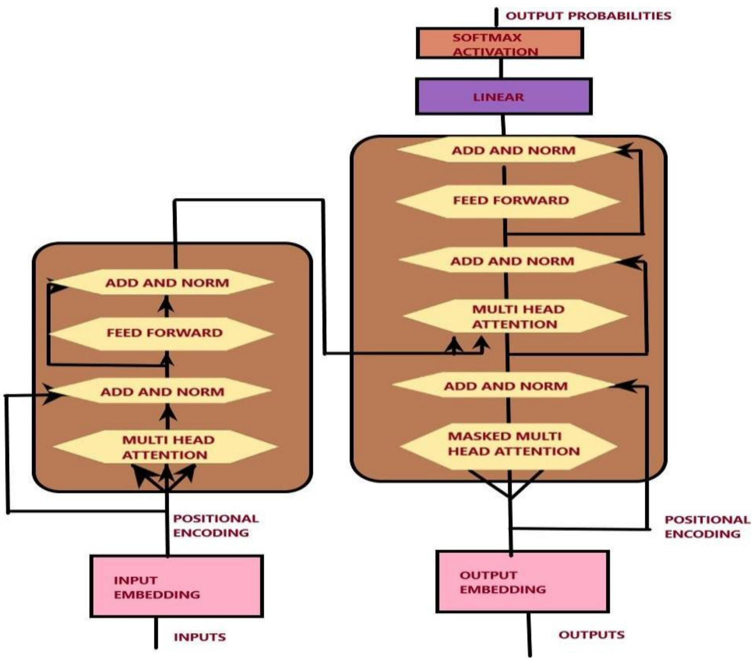

Fig.4 BERT Architecture[18]

The left side of the architecture represents an encoder while the right side represents a decoder. In both the encoder as well as the decoder, the different modules are stacked upon one another. The modules in encoder are – Multi head attention, Add and norm and Feed forward. In decoders there is an additional module masked multi head attention. One of the exclusive features of the architecture is the positional encoding.

In BERT model, there are no recurrent neural networks. Therefore, remembering the information is impossible for the model. The positional encoding is a vector representation that has the same dimensions as that of the embedding. The positional embedding value is appended to our embedding value and then passed as input to the encoder as well as decoder. Hence the positional embedding keeps track of the location of the values in the data.

Bert uses multi-head attention layer in its architecture as shown in Fig.5. This means that many attention mechanisms are running in parallel. In BERT, various attention mechanisms are learnt by the model and these attention mechanisms run in parallel to one another. This is known as multi-head attention layer. In the decoder, an additional attention mechanism called, masked multi-head attention mechanism is added. This means that the data is masked or hidden in this case.

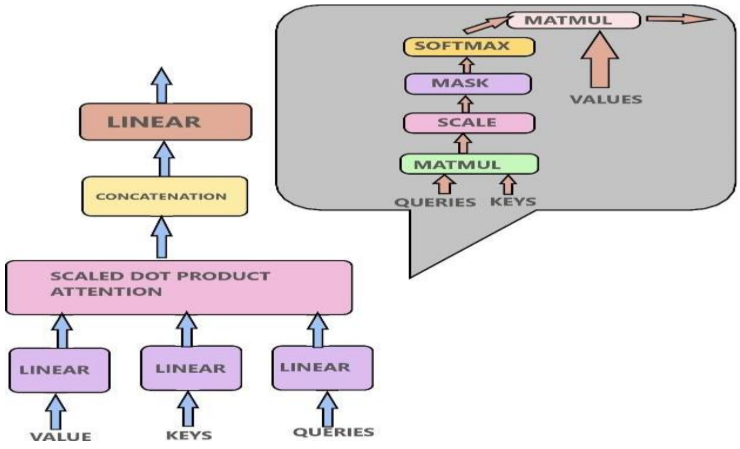

Fig 5 Multi-head attention mechanism[18]

The BERT model is pre-trained on a large set of unlabeled data. The BERT model reads the input at once. Therefore, the BERT model gets a complete context of the words in text. The model can understand the relation of the word with a word that comes before and that comes after.

The attention is mathematically defined as follows:

$$\text{Attention}(q,k,v) = \text{softmax}[(q * k^t) / \text{sqrt}(dk)] * V \quad (9)$$

Where, q = matrix containing set of queries

k == keys

v = values

BERT uses multiheaded attention and the mathematical representation for same is:

$$\text{Multihead Attention}(q,k,v) = \text{concatenate}\ (head_1, head_2 \ldots\ldots\ldots\ldots head_n)\ W^o \quad (10)$$

Where,

$$head_i = \text{Attention}(qW_i^q, kW_i^k, vW_i^v) \quad (11)$$

q = matrix containing set of queries

k = keys

v == values

W = weight

dk = dimensions of keys

BERT [19] is a masked language model. During training the model, some of the words in a text are masked and the model is aimed to guess these words. For the purpose of tweet classification, we use pre-trained BERT model. The BERT models are trained on a large set of unlabeled data. For the purpose of text classification- a) the text data is converted into tokens b) a classification layer is added on top of the model. Every input data in BERT model has a CLS token at the start of a sentence and a SEP token at the end of the sentence

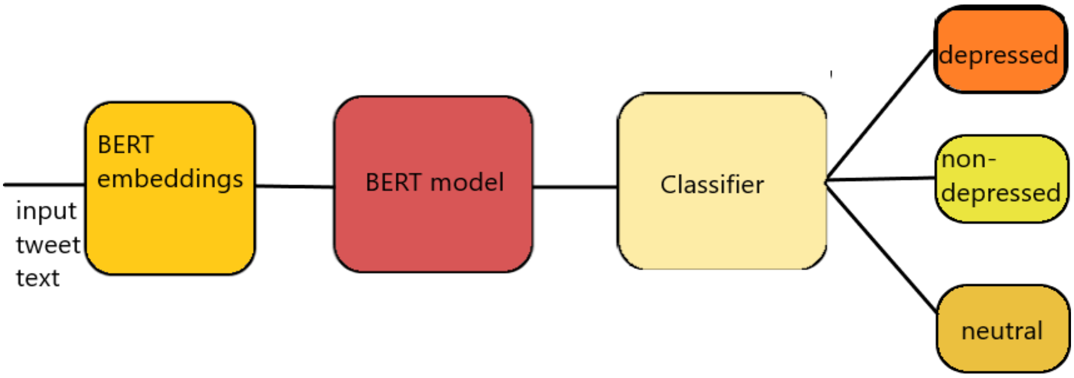

Fig.6 BERT model for classification of tweets

Looking at the architecture of BERT as well as the Fig 6, we know that the textual data has to undergo embedding before passing through encoder as well as decoder. Bert embedding is a type of word embedding in which a word is represented as a vector. There are two main tokens in the BERT embedding – SEP and CLS token. The SEP token is used to separate the two sentences and the CLS token is present at the start of the text. For the classification problem, , the CLS token is crucial. The BERT embedding can separate every word or sentences into token or break a word into parts represented by "##" in the beginning. The last resort is to break the word into characters and also represented by "##" in the beginning.

3.2.3 XLNET

XLNet is an autoregressive model that is non- directional in nature. This means that the predicted word depends on the words around it. The non-directionality of XLNet autoregressive model is achieved by introducing a new concept called "permutations". XLNET draws up two concepts from the BERT model- 1. Positional encoding and 2. Segment recurrence. When a sequence is to be passed through a XLNET architecture, it has to undergo word embedding first. The word embedding is performed by the tokenizers. After, the input is embedded, a positional embedding vector is appended to the word embedding and after this, the data is input to the XLNET architecture. XLNET architecture caches the hidden state of the very first segment in the memory and this is repeated for all the layers of the architecture. This allows the system to update the attention at regular intervals. If a system has a depth D, the segment recurrence takes place D times for every level. Segment recurrence can increase the contextual understanding of the architecture and helps achieve the longest dependency possible for a model. XLNet has two main types with respect to their architecture- XLNet-base have 12 encoder-decoder sequences and XLNet-large have 24 encoder-decoder sequence.

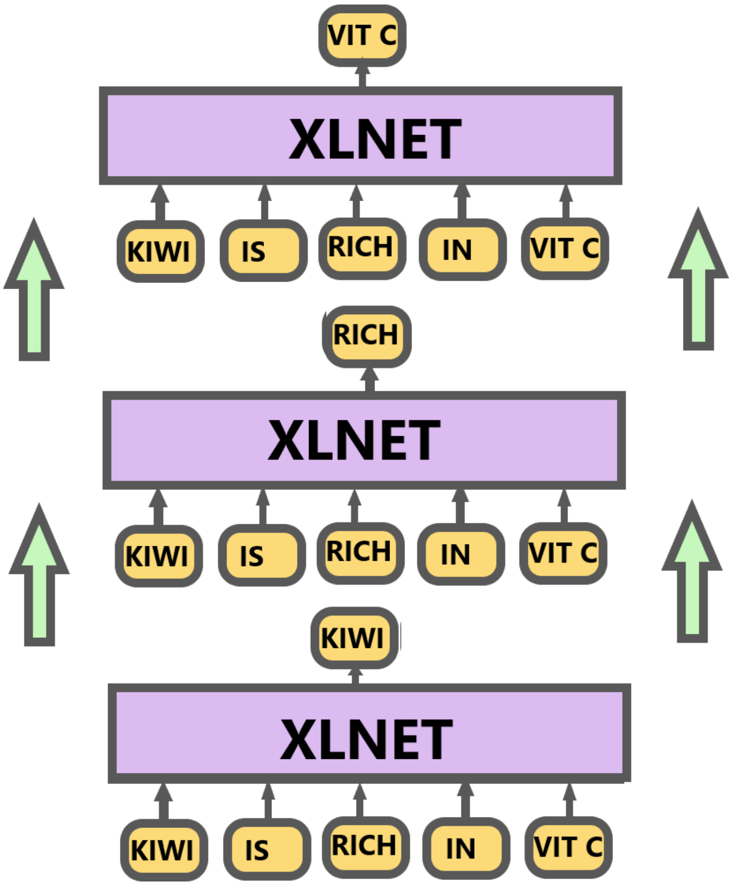

Fig .7 Working of XLNET

The working of XLNET for the sequence is explained in the Fig.7. XLNET[20] is a permutation-based model. In this model, the probability is maximized by taking into consideration all the possible permutations in the factorized order. When we take all the possible permutation into order, we can get a context of the tokens that are prior to a token as well as tokens that follow a particular token.

The following mathematical equation explains the idea behind the XLNET model

$$\Omega = \text{argmax}_{\Omega} [E_{z \sim Z} [\Sigma^{T} \log[\Pr(x_{z[t]} \mid x_{z[<t]})]]] \qquad (12)$$

Where, Ω = model parameter

x = token
T= Sequence length
Z= Permutations
z_t = t^{th} element in permutations Z

The equation 12 explains the principle that is used in XLNET model and that has enabled it to outperform the rest of the models. That is, the permutation language model. XLNET is a generalized autoregressive model. The mathematical equation governing the Auto Regressive (AR) model is given below

$$\max_{\theta} \quad \log p_{\theta}(\mathbf{x}) = \sum_{t=1}^{T} \log p_{\theta}(x_t \mid \mathbf{x}_{<t}) = \sum_{t=1}^{T} \log \frac{\exp\left(h_{\theta}(\mathbf{x}_{1:t-1})^{\top} e(x_t)\right)}{\sum_{x'} \exp\left(h_{\theta}(\mathbf{x}_{1:t-1})^{\top} e(x')\right)}, \qquad (13)$$

Where, $h_{\theta}(\mathbf{x}_{1:t-1})$ = context representation generated by neural networks

e(x) = embedding of x

x = text sequence

The main aim of the equation mentioned above is to maximize the likelihood during the forward autoregressive factorization.

3.3 Evaluation and Visualization

The models deployed are evaluated based on standard metrics like the accuracy, precision, recall value, F1 score. Based on these metrics, the model with the highest accuracy, reduced error and loss be chosen. After best model chosen, Word Cloud visualization shown for post lock down data set and the model analyzed. The implementation results and analyses are explained in the forthcoming section

4. Implementation Results and Analyses

For this research work, dataset from March, 2020 through December, 2020 was collected. A dataset of 105000 rows was collected. The twitter data of the March

2021 was collected using the tweepy library. We just collected tweets few hours for 15 March 2021 for the final comparisons. This dataset was used to implement the final model and predict the labels. Data set was preprocessed by applying TF-IDF vectorization and support vector classifier for labelling the tweets as depressed, not depressed, neutral represented numerically as 0, 1 and So, to understand the dataset better, we create a word cloud visualization of the dataset. The word cloud visualization is a plotting technique in which the words of the text data are represented. The word cloud provides a way for us to understand the textual data better. The words in the text data that have higher frequencies appear larger in size and the words that appear smaller have less frequency in text data.

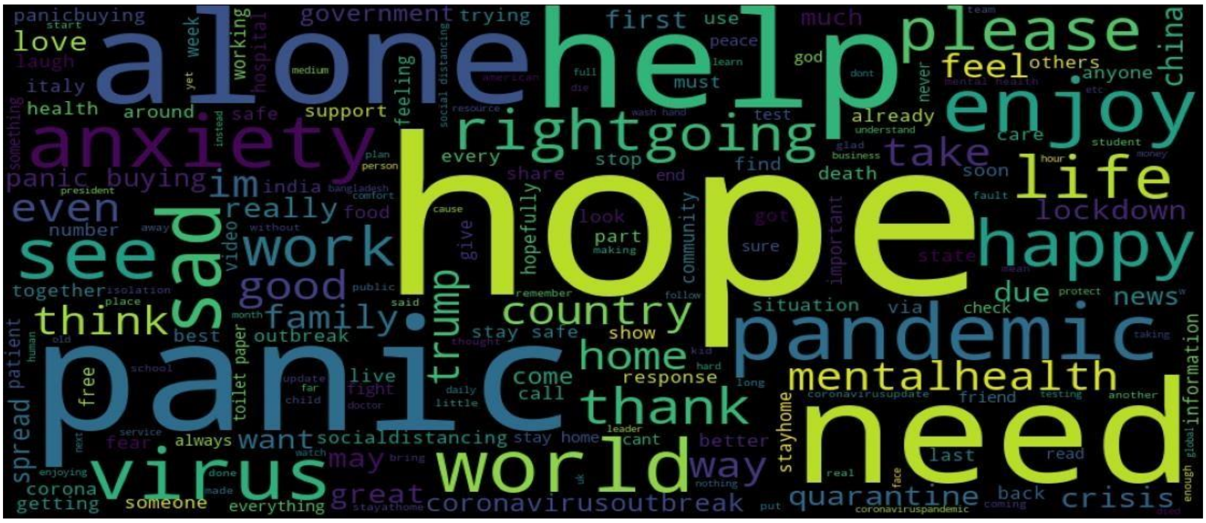

Fig.8 Word cloud visualization of tweets during lockdown

From Fig8, we can see the words panic, alone, help, anxiety appear in larger size. These words can be associated with the vocabulary of the depressed words. Therefore, we can make an assumption that the tweets that were made during the lockdown were made by people who were suffering from depression, anxiety, depleting mental health etc.

4.1 Performance of Deep Learning Models

4.2.1 Bi-LSTM Results

In building the Bi-LSTM model we employed a powerful technique of word embedding known as GloVe embedding. GloVe embeddings are used to create a vector representation of the text. The GloVe embeddings use both the global as well as local statistic of a corpus to create a word embedding.

Our research work is a multiclass classification problem. For this purpose, we have used "Sparse Categorical Cross Entropy" loss function. The sparse categorical cross entropy loss function works similar to the categorical cross entropy loss function. The only difference is that the output labels in categorical cross entropy loss function is encoded while the output labels in sparse categorical cross entropy loss function are not encoded. The mathematical formula for the loss function are as follows:

Loss = Σi=1 n yi * log ÿi Where Loss= calculated loss (14)

Where

n = number of scalar values in output

ÿi = ith scalar value in output of model

yi = corresponding target value

To overcome the problem of overfitting in Bi- LSTM, we have used a dropout of 0.2. Dropout is a regularization technique. In this, some of the outputs of the model are ignored or dropped while training. This helps regularize the dataset and prevent overfitting.

Optimizers are used in deep learning models to adjust the value of characteristics such as learning rate and weight decay to reduce the value of loss and simultaneously increase the accuracy of the model. The twitter sentiment analysis presented in this work using a bi-directional LSTM used GloVe embedding to create a vector representation of the tweet. Also, for building a Bi-LSTM model, Adam optimizer was used. Table-1 shows the performance evaluation of Bi-LSTM model

Performance Metrics	Values
Precision	0.819
Recall	0.823
F1 Score	0.818
Accuracy	82.332

Table-1: Bi-LSTM Evaluation Results

4.2.2 BERT Results

The BertTokenizerFast is the Bert tokenizer that has been employed for tokenizing the input tweet texts. To classify the tweets as depressed, non- depressed and neutral, we have added a classification layer on top of the BERT model. AdamW optimizer has been used for the model. The AdamW optimizer separates the weight decay and the learning rate. This implies that both the learning rate and the weight decay can be optimized separately. Table-2 shows the BERT Evaluation Results

Performance Metrics	Values
Precision	0.966
Recall	0.963
F1 Score	0.965
Accuracy	96.716

Table-2: BERT Evaluation Results

4.2.3 XLNET

XLnet is used for the purpose of text-classification and not for word prediction. For the same, the hugging face transformer library has pretrained model. We have used BertForSequenceClassification to predict the class of our tweets. For generating input ids and attention masks, XLNetFastTokenizer has been used to encode all the tweets. AdamW optimizer with a learning rate of 5e-5 has been used for training the model. The learning rate enables the model to learn the information which helps improve the prediction. The weight decay has been set at 0.01 and also helps in the regularization of the input for training. Table-3 shows the XLNET results

Performance Metrics	Values
Precision	0.956
Recall	0.955
F1 Score	0.956
Accuracy	95.802

Table-3: XLNET Evaluation Results

4.3 Comparative Analysis of Deep Learning Models

The comparative analysis of three Deep learning models in terms of accuracy, precision, Recall, F1 Score and losses are plotted in Fig.11

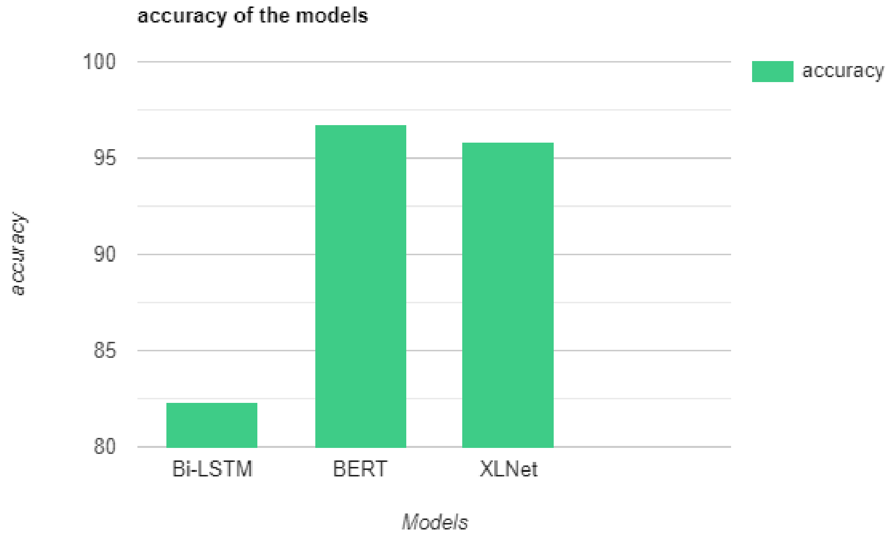

Fig.11 Accuracy of Models

From the analysis of results, it has been shown that BERT outperforms Bi-LSTM and XLNET in terms of accuracy. That is BERT achieved an accuracy of 96% as compared to Bi-LSTM and XLNET. In terms of Precision and Recall, BERT model achieved precision score of 96% as compared to XLNET and Bi-LSTM. This shows that BERT has predicted maximum Tweets accurately and positively with very less false positive and false negatives. In terms of F1 Score, the value is 0.97 which is near perfect classifier as maximum value is 1.0 as compared to XLNET and Bi-LSTM. Also losses for all the three models decreases with increasing number of epochs which shows there is no overfitting.

4.4 Post Lockdown Tweet Analysis

Using the model that performed best which is BERT, we predicted the class of a small post-vaccination dataset into depressed, non-depressed and neutral and draw a comparative analysis of the mental-health during lockdown vs post-vaccination / post- lockdown to get an insight. Fig.12 shows the word cloud visualization how words like happy, glad, hope have been used more in the data post vaccination. This gives us a faint perspective of improved mental health. But to actually draw an analysis, we need to know the distribution of the data into the various classes.

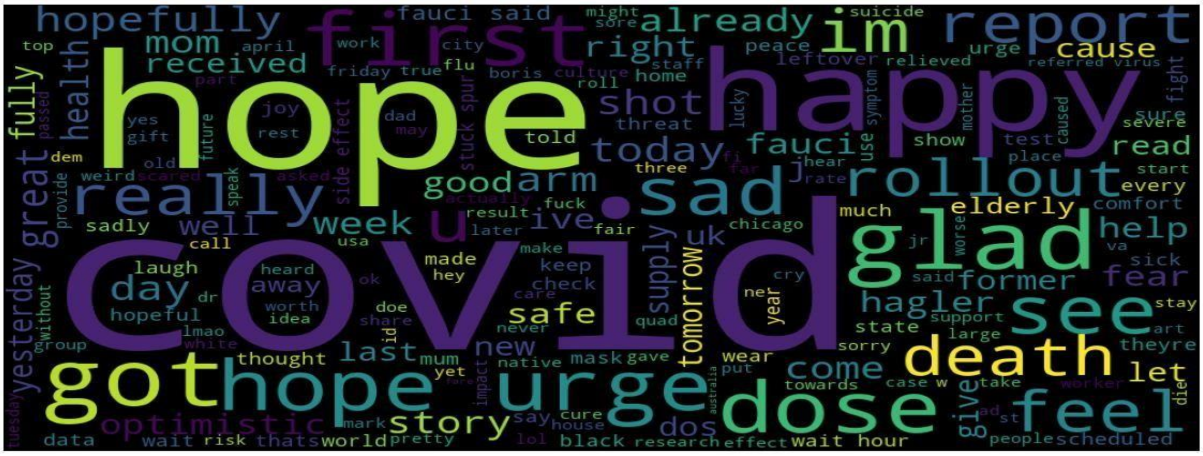

Fig.12 Word cloud visualization of tweets post lockdown and post vaccination

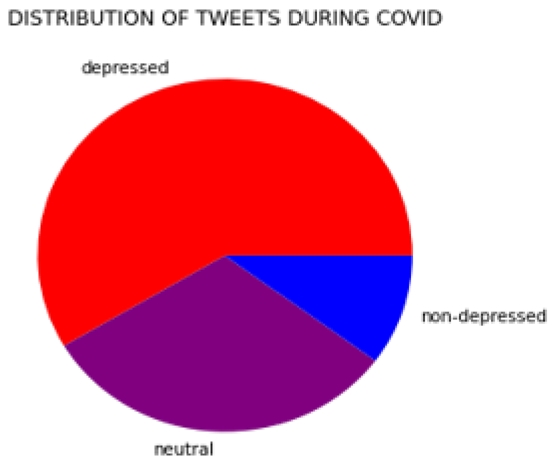

Fig.13 Distribution of Tweets during Lockdown

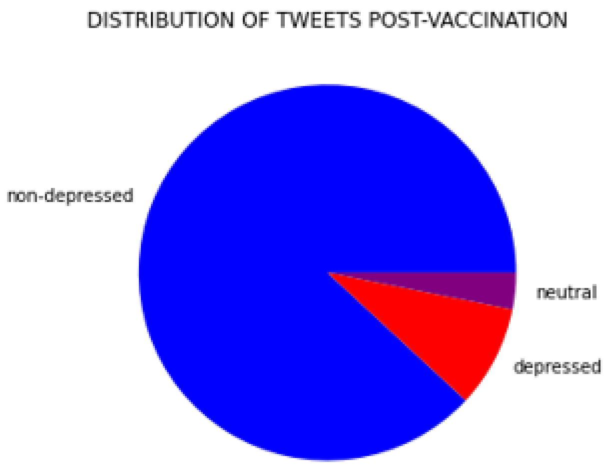

Fig.14 Distribution of Tweets Post Lockdown

From figures 13 and 14, we can clearly see that there has been more than 40% drop in the depressed tweets. In fact, the tweets that have been made post the vaccination are dominated into non-depressed category.

5. Conclusion and Future Work

During the lockdown that was imposed due to covid-19 pandemic, the people were made to sit at home for months. This affected the mental state of a lot of people. There has been some considerable usage of deep learning and machine learning for Tweet Sentimental analysis for different applications including mental depression. There has been no much work found pertaining to COVID Tweet analysis towards mental depression So towards this, we in this work focused on applying deep learning models on COVID Tweet data set for predicting the Tweet data as depressed, non-depressed and neutral The models that have been implemented were Bi-LSTM, BERT and XLNET. BERT outperformed the other two models which are Bi-LSTM and XLNET with an accuracy of 96.7% approximately. This model was then used to get an insight of how the vaccinations and relaxations in lockdown have brought an improvement in the mental health of people. The research work got lot of scope for future enhancement. So, if we used the models to train in languages like Italian, French, Spanish as well as German we can also decide about the mental health in worst hit countries like Italy, France, Spain etc. The post vaccination data is not abundant. The vaccination has started very recently and many countries have re imposed restrictions. In future when the vaccination drive would be carried out in mass and people are vaccinated, we can understand better about how the vaccination has impacted the mental health of the people.

References

[1] Bollen, J., Mao, H., & Pepe, A. (2011). Modeling Public Mood and Emotion: Twitter Sentiment and Socio-Economic Phenomena. Proceedings of the International AAAI Conference on Web and Social Media, 5(1). Retrieved from https://ojs.aaai.org/index.php/ICWSM/article/view/14171

[2] Zunaira, J., Diana, I., & Prasadith, B.(2017). Monitoring Tweets for Depression to detect-at-risk users. Proceedings of Fourth Workshop on Computational Linguistics and Clinical Psychology, Vancouver, Canada. Retrieved from https://www.aclweb.org/anthology/W17-3104.pdf

[3] Rane, A and Kumar, A(2018). Sentiment Classification System of Twitter Data for US Airline Service Analysis. Proceedings of 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 2018, pp. 769-773, doi: 10.1109/COMPSAC.2018.00114

[4] Zhang Z., Robinson D., Tepper J. (2018) Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network. In: Gangemi A. et al. (eds) The Semantic Web. ESWC 2018. Lecture Notes in Computer Science, vol 10843. Springer, Cham. https://doi.org/10.1007/978-3-319-93417-4_48

[5] Maryam, H., Elke, R., & Emmanuel, A.(2014). EMOTEX: Detecting Emotions in Twitter Messages. Proceedings of ASE BIG Data/Social Com/Cyber Security Conference, Standford University, USA. https://web.cs.wpi.edu/~emmanuel/publications/PDFs/C30.pdf

[6] Diveesh, S., & Aileen, W.(n.d). Detecting Depression through Tweets. Available from https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6879557.pdf

[7] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep Learning for Hate Speech Detection in Tweets. In Proceedings of the 26th International Conference on World Wide Web Companion (WWW '17 Companion). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 759–760. DOI:https://doi.org/10.1145/3041021.3054223

[8] Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Hate Speech Detection Using a Convolution-LSTM Based Deep Neural Network. In Proceedings of ACM The Web conference (WWW’2018). ACM, New York, NY, USA, Article 4, 10 pages. https://doi.org/10.475/123_4

[9] Chayan Paul, Pronami Bora-Detecting Hate Speech using Deep Learning Techniques. International Journal of Advanced Computer Science and Applications(IJACSA), vol.12(2), pp.619-623.

[10] Alshalan R, Al-Khalifa H, Alsaeed D, Al-Baity H, Alshalan S. Detection of Hate Speech in COVID-19-Related Tweets in the Arab Region: Deep Learning and Topic Modeling Approach. J Med Internet Res. 2020 Dec 8;22(12):e22609. doi: 10.2196/22609.

[11] Amrutha B R and Bindu, K R(2019). Detecting Hate Speech in Tweets Using Different Deep Neural Network Architectures. Proceedings of *Intelligent Computing and Control Systems (ICCS)*, pp. 923-926. doi: 10.1109/ICCS45141.2019.9065763 .

[12]Prateek, J (2018). Analytics Vidya [Online]. Available: https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysisdataset-python

[13] *Krouska, A ., Troussas, C., and Virvou, M.(2016). The effect of preprocessing techniques on Twitter sentiment analysis. Proceedings of* 7th International Conference on Information, Intelligence, Systems & Applications (IISA), *pp. 1-5. doi: 10.1109/IISA.2016.7785373*

[14]. Ana Solaguren, B (2020). Active Learning in Machine Learning[Online]. Available from https://towardsdatascience.com/active-learning-in-machine-learning-525e61be16e5

[15] Bruno, S (2019). What is TF-IDF [Online]. Available: https://monkeylearn.com/blog/what-is-tf-idf/

[16] Rohith, G (2018). Support Vector Machine- Introduction to Machine Learning Algorithms [Online]. Available: https://towardsdatascience.com/support-vector-machine-introduction-to-machinelearning-algorithms-934a444fca47

[17]Raghav, A (2019). Bi-LSTM[Online]. Available from https://medium.com/@raghavaggarwal0089/bi-lstm-bc3d68da8bd0

[18] Ashish, V et al (2017). Attention is all You Need. Proceedings of Neural Information Processing Systems, Long Beach, CA, USA. Available from https://arxiv.org/pdf/1706.03762.pdf

[19] Rani, H (2018). BERT Explained: State of art Language Model for NLP[Online]. Available from https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

[20] Aishwarya, V S (2019). XLNET Explained in Simple Terms[Online] Available from https://towardsdatascience.com/xlnet-explained-in-simple-terms-255b9fb2c97c