# A Machine Learning and Deep Learning Approach to Classify Mental Illness with the Collaboration of Natural Language Processing

Md Rafin Khan, Shadman Sakib, Adria Binte Habib, and Muhammad Iqbal Hossain

Department of Computer Science and Engineering, Brac University, Dhaka, Bangladesh
shadmansakib018@gmail.com, khanrafin0@gmail.com, binte.adria708@gmail.com, iqbal.hossain@bracu.ac.bd

**Abstract.** The most alarming yet abstained issue of our so-called 'Generation Z' is mental health. In many developing countries, it is unfortunately treated as a mere joke by a majority of the population. The only way to tackle this is to find out the correct mental illness associated with an individual and provide a systematic solution as early as possible. In this paper, the authors emphasized the category of a disease rather than just generalizing it as depression. Four highly anticipated mental health statuses were selected which were Schizophrenia, PTSD, Bipolar Disorder, and Depression. This research proposes to identify which of these mental illnesses a person is most likely to be diagnosed with. It has been done by multiple classification algorithms and the language patterns of such self-reported diagnosed people from a corpus of Reddit posts to discover better outcome.

**Keywords:** Schizophrenia, PTSD, Bipolar Disorder, Depression, Tokenization, Lemmatization, TF-IDF & Count-Vectorizer

## 1 Introduction

In many countries, mental illnesses are thought of as imaginary things, and most families are ashamed if one of their family members gets affected with mental illness, so we were thinking what if there was a more discreet way for people to diagnose themselves without the fear of social stigma. Despite the fact that there are billions of individuals, there are just a few clinical psychiatrists [1]. The article also mentions that it can be difficult to know who to turn to for help when there is a scarcity of facilities, hospitals, or psychiatric services. According to the authors, there are limited laboratory tests for diagnosing most forms of mental illness, and the major source of diagnosis is the patient's self-reported experience or behaviors recorded by family or friends [2]. This is where the strength of advanced algorithms of Machine Learning comes into play which will forecast what type of mental illness the person has so that they can diagnose themselves from home.

As we all know that mental health has been one of the noteworthy issues in healthcare and plays a vital impact on one's quality of life, therefore we must find a way to quickly detect and diagnose it. In most cases, it is very difficult to express one's true emotions in front of family and friends, hence individuals tend to express themselves on social media platforms hoping to engage with other fellow victims for compassion and/or methods to ease the suffering or even just share their experiences. The authors stated that the ability to discuss mental health issues anonymously on the internet motivates people to reveal personal information and seek help [3]. As a result, social media such as Reddit, Twitter and many others become a significant resource for mental health researchers in studying mental health. Although data from platforms on the internet such as Twitter, Facebook or Reddit is readily available, labeled data to study mental illness is confined [4]. Due to scarce information, it has been difficult to understand and address the different challenges in this domain, hence information retrieved from social networks not only provides medical assistance to the users in need but also expands our awareness of the common conditions of mental disorders.

The goal of this study is to determine whether or not a person has PTSD, Bipolar Disorder, Schizophrenia, or Depression with the help of Natural Language Processing and Machine learning algorithms from the corpus of any social media post. Prior work [5] has inspired us to improve existing models and test new ones by working on the SMHD dataset which was collected from Georgetown University, in order to improve prediction accuracy and overall precision, recall, and f1 performance. We have used a large-scale Reddit dataset, collected via an agreement, to conduct our research. Reddit is an open-source forum where members of the community (redditors) can submit content (such as articles, comments, or direct links), vote on submissions, and organize content by topics of interest (subreddits). The following is a breakdown of the paper's structure: First, an explanation is provided on dataset collection and preprocessing techniques. After that, various classification algorithms are implemented. Then the accuracy, precision, recall, and f1 score are measured. Finally, concluded with a discussion of plausible future works in this area.

## 2 Related Work

In the recent era of 5G, social media has a drastic impact on our everyday lives. It is a platform for human beings to keep in touch or update their daily lives through posts, pictures, opinions etc. However, who knew that the opinions and thoughts of the social media users would amount to such valuable research study. Previous works of researchers who used twitter posts as datasets to identify depression and other mental disorders have left us valuable findings that we can use and enhance our results. In 2013, the authors had suggested a strategy for detecting depressed users from Twitter posts where they used crowd sourcing to compile the Twitter users [2]. They measured data such as, user engagement and mood, egocentric social graphs and linguistic style, depressive language use and antidepressant medication mentions on social media. They then contrasted the activities of depressed and non-depressed users, revealing indicators of depression such as decreased social activity, increased negative feeling, high self-attentional focus, increased relationship and medical concerns and heightened expression of religious views. They used multiple sorts of signals to create an MDD classifier that can predict if an individual is sensitive to depression ahead of the beginning of MDD. As time proceeded, the works of identifying the type of mental health conditions were emphasized more rather than carrying out surveys. For instance, the authors have analysed four types of mental illness (Bipolar, Depression, PTSD, SAD) from 1200 twitter users using Natural Language Processing. The diagnosis statement from their tweets such as "I was diagnosed with depression" was conducted through an LIWC tool and the deviations were measured from a control group against those four groups [6]. Then, they used an open vocabulary analysis to collect language use that is related to mental health in addition to what LIWC catches. Subsequently in 2014, the researchers conducted an elaborated work to classify PTSD users from twitter and found out an increasing rate among them, especially targeted among US military soldiers returning from prolonged wars [7]. Similarly, the LIWC tool was used to investigate the PTSD narrations used by these self-reported users for language comparison. Other than that LIWC was also used in finding linguistic patterns between users to classify ten mental health conditions from Twitter posts [4]. Nonetheless the paper [8] analyses tweets to figure out symptoms of depression like negative mood, sleep disturbances, and energy loss. The paper [9] shows further study investigated on the mental health-related social media text categorization generated from Reddit. Although studies to date witnessed CNN to be a better model in terms of performance, the authors implemented hierarchical RNN architecture to address the problem. Because of the fact that computational costs are higher and removal of unnecessary contents makes the model faster, they also employed attention mechanisms to determine which portions of a text contribute the most to text categorization. Even though twitter posts are a significant source of data for language usage, long forums and contents are also pivotal for a valid dataset. In this case, Reddit users are applied for building a corpus which gives newer insights to linguistics. The paper [5] mentions that unlike Twitter, which has a post limitation in terms of word count, Reddit platform has no such constraints. Also this dataset they have used, contains posts of diverse mental health condition patients along with mentally healthy Reddit users also known as control users. Their data was compiled with the use of high-precision diagnosis patterns. The filtering of control users in the dataset was rigid. For example, any Reddit user who never had any post related to mental health as well as having no more than 50 posts were not included. The paper also mentioned that control users tend to post twice as much as any diagnosed user and their posts are comparably shorter. They looked at how different linguistic and psychological signs indicated disparities in language usage between those with mental illnesses (diagnosed users) and others who were not (control users). To identify the diagnosed users, several text categorization algorithms were tested, with FastText proving to be the most effective overall. In 2016, the authors used Reddit posts and comments and paired 150 depressed users with 750 control users to find out the language distinction between users who are depressed and those who are not [10]. They have outlined the methods they used to generate a test collection of textual encounters made by depressed and non-depressed persons. The new collection will help researchers look into not only the differences in language between depressed and non-depressed persons, but also the evolution of depressed users' language use. The authors [3] applied self-reported diagnoses to a broader set of Reddit users, yielding the Reddit Selfreported Depression Diagnosis (RSDD) dataset, which had over 9,000 users with depression and over 100,000 control users (using an improved user control identification technique). Posts in the RSDD dataset were annotated to check that they contained assertions of a diagnosis. Similar kind of task was done by one of the authors in their paper [4], who were also able to authenticate self-reports by deleting jokes, quotes and false statements.

## 3 Data Analysis

### 3.1 Dataset Overview

The SMHD (Self-reported Mental Health Diagnoses) dataset is collected from Georgetown University [5]. SMHD had conditions corresponding to branches in [11] a total of 9 conditions and only 4 from these are listed in Table 1. These are Schizophrenia, Depression and Bipolar are top-level DSM-5 disorders whereas PTSD is one level lower.

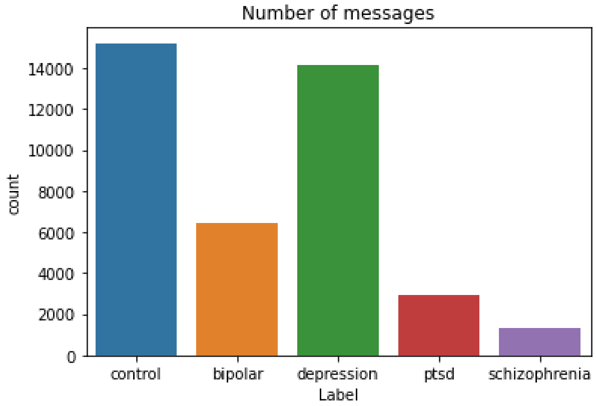

**Fig. 1.** Visual Representation of Number of Messages by the self-reported diagnosed users per condition

### 3.2 Dataset Formation

The data is in the form of json lines (.jl) format, which basically means that each line of the files were in json format. Each json line of the files represented one user, it contained an id, the label of the user's mental health condition and all the posts that the user wrote with the time and date when each comment was posted. Since the dataset is enormous; approximately 50GB, we loaded it using the ijson library of Python, which loads and parses files using iterators to load data lazily. As a result, if an unnecessary key passes by it can be simply ignored and the created object will be removed from memory. This helped avoid exceeding memory usage constraints set by Google Colab runtime. A data frame was created where each row consists of only the labels and the posts all concatenated per user. Another effective measure was taken to optimize memory usage was to pre-process the data, which has a more in-depth discussion in section 4.1, before concatenating the posts iteratively. Lastly, the Pandas Dataframe was converted to a csv file to access it at ease.

## 4 Methodology

### 4.1 Data Pre-processing

In order to apply machine learning algorithms in the text data, the data must be clean. In other words, algorithms perform better with numbers rather than text [12]. Thus the dataset was loaded into a pandas dataframe and then ‘feature engineered’.To bring all the values of different numerical range into a standard region standardization, log normalization and feature scaling was used. To make the dataset structured, the data was filtered by removing dirty data such as missing row values, NaN type or mixed data such as emoticons. At first, punctuation marks and stopwords were removed, i.e connecting words ’i’, ’we’, ’me’, ’myself’, ’you’, ’you’re’ which form a meaningful sentence, from the text since it does not add any value to classification models. Besides, lemmatization was also used which reduces the inflected words to its root word [13]. The reason behind doing lemmatization instead of stemming is because lemmatization always reduces to a dictionary word although it is computationally expensive. Despite the fact that stemming is faster than lemmatization, it simply chops off the end of a word using heuristics while lemmatization uses more informed analysis. Previous works also suggest that truncation of post length improves classification performance [5]. For dealing with text data, each of the words need to give unique numbers since machine learning models can not be trained on text data. The fit and transform method of the TF-IDF class was used from the Sci-Kit learn library. TF-IDF is a simple tool for tokenizing texts and creating a vocabulary of known terms, as well as encoding new texts with that vocabulary [14]. Tokenization basically refers to splitting up the raw text into a list of words which helps in the comprehension of the meaning or the creation of the NLP model and Python does not know which word is more important, hence we need further pre-processing. TF-IDF creates a document term matrix where columns are individual unique words and the cells contain a weight which signifies how important a word is for an individual text which means if the data is unbalanced, it will not be taken into consideration. In other terms, words that appear often in the text, for example ‘what’, ‘is’, ‘if’, ‘the’, are scored low since they will have little meaning in that particular document. However, if a word appears frequently in a document but not in others, then it is likely to be relevant. By this way, the TF-IDF algorithm sorts the data into categories which assist our proposed models to work faster and bring outstanding results.

$$W_{i,j} = TF_{i,j} log(N/DF_i) \tag{1}$$

The dataset was divided into train (70%) and test (30%) groups to make the classification model more precise, then investigated model construction and employed some machine learning algorithms, optimized it, and evaluated the performance of each model. To make our workflow more streamlined, a pipeline was used which allowed us to sequentially apply a list of transformations. This allowed us to implement a few classifiers in a short amount of time.

### 4.2 Proposed pipeline

After the data were processed, it is time to fit it into the appropriate estimator. For various data types, estimators perform differently, therefore picking the proper estimator might be tricky. The followed approach was to classify all mental health conditions against our control group. Essentially a one-to-one approach. The reason it was done due to the text data between the mental health conditions being quite similar; as it seems it should be. A person that is diagnosed with PTSD is highly probable to be suffering from depression as well. Thus we want to classify between a person either being mentally ill or healthy. Following much investigation, we have arrived at the conclusion that the models listed below should be used in our work.

## 5 Experimentation

### 5.1 Support Vector Machine

In order to select the best hyper-parameter combination GridSearchCV was used. It is the process of determining the ideal settings for a model’s hyper-parameters. The value of hyper-parameters has a substantial impact on a model’s performance. It’s worth noting that there’s no way to know ahead of time what the best values for hyper-parameters are, therefore we should try all of them to find the best ones, because manually adjusting hyper-parameters would take a significant amount of time and resources. Though it is computationally expensive, we decided it would be worth it for better results. The parameters that we tuned were C: cost parameter to all points that violate the constraints, gamma: defines how far the influence of a single training example reaches, and the Kernel. Having a low value of C creates a smooth hyperplane surface whereas a high

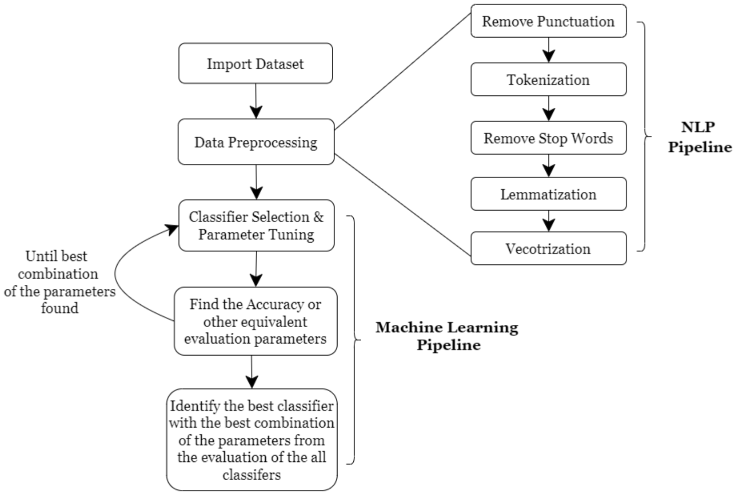
Import Dataset
Data Preprocessing
Remove Punctuation
Tokenization
Remove Stop Words
Lemmatization
Vecotrization
NLP Pipeline
Classifier Selection & Parameter Tuning
Until best combination of the parameters found
Find the Accuracy or other equivalent evaluation parameters
Identify the best classifier with the best combination of the parameters from the evaluation of the all classifers
Machine Learning Pipeline

**Fig. 2.** Proposed pipeline

value tries to fit all training examples correctly at the cost of a complex surface. Having run the GridSearchCV the best parameters were found. It is given in the Table 1.

### 5.2 Logistic Regression

Logistic regression is not only simple and easy to model, it is also very efficient which is extremely useful in our case due to our data being very large. Even though our dataset is fairly big it is very simple consisting of only two columns (Label and Text), therefore logistic regression is perfect since it produces better accuracy for simpler data. Like any other model logistic regression can also overfit but chances of overfitting is low, still to avoid this the authors have implemented the l2 regularization within the model which basically operates as a force that removes a little portion of the weights in each iteration causing the weights to never reach zero. We implemented newton-cg algorithm for optimization and used a max iteration of 2000 which is basically the number of iterations taken for the optimizer to converge.

### 5.3 Gated Recurrent Unit (GRU)

For the model a GRU layer has been utilized consisting of 100 units with activation of ‘relu’ and dropout of 0.3 followed by a dense layer of 1000 units with activation ‘relu’ and dropout 0.7, this is repeated twice followed by an output dense layer of unit 1 with activation ‘sigmoid’. the max token length has been taken to be 600 and padded any text length that is less than the given max token

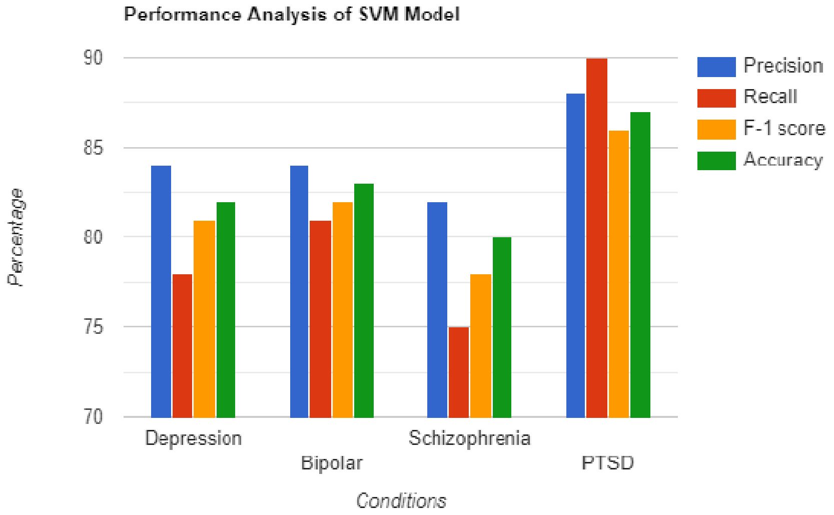

**Fig. 3.** Performance Analysis of SVM

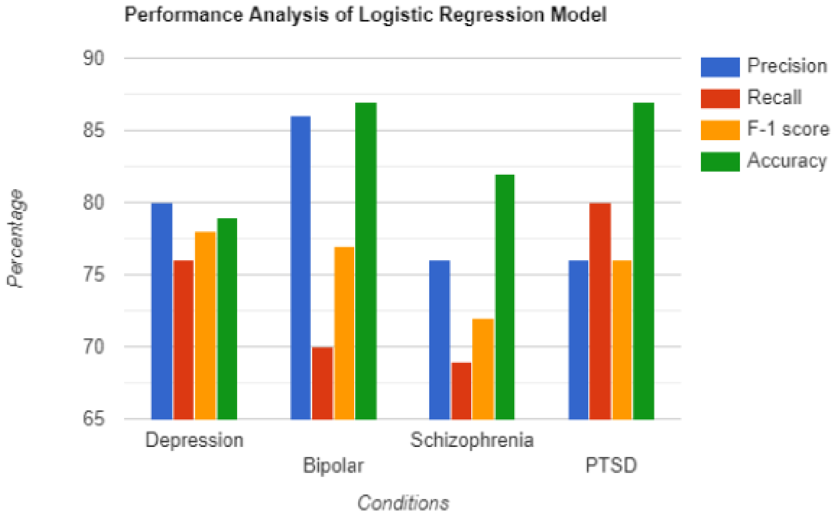

**Fig. 4.** Performance Analysis of Logistic Regression

length. To minimize overfitting, dropouts were introduced, and the loss function was binary cross entropy using the ‘adam’ optimizer to optimize weights and learning rate, which helped reduce loss. Finally, the model was trained for 10 epochs for each sickness. Since ours is binary classification, sigmoid is the best for output layer activation since it gives a result between 0 and 1 which can be inferred as how confident the model is in an example being in a particular class. And binary cross entropy basically compares two probability distributions and calculates the difference between them which is perfect for our binary classification.

### 5.4 Bidirectional Encoder Representations from Transformers (BERT)

To make the model even more lightweight ktrain was used. Ktrain is a keras library that aids in the creation, training, debugging, and deployment of neural networks. Ktrain allowed to easily employ the distilBERT pre-trained model and estimate an optimal learning rate. To utilize Ktrain, the “get learner” function was used to wrap our data and model, in this case ”distilbert-base-uncased,” inside a ktrain learner object.A batch size of 16 was used for faster performance. The Learner object allows training in various ways. One of the most crucial hyperparameters to configure in a neural network is the learning rate. Default learning rates for various optimizers, such as Adam and SGD, may not always be appropriate for a given problem. The training of a neural network requires minimizing a loss function. If the learning rate is too low, training will be postponed or halted. If the learning rate is too high, the loss will not be reduced. Both of these conditions are detrimental to the performance of a model. The author says that when graphing the learning rate vs. the loss, the greatest learning rate associated with a dropping loss is a preferable choice for training [20]. Thus a learning rate of $1 \times 10^4$ was used inferring from the graph shown in Figure 4. The model was trained on a maximum of 5 epochs and a minimum of 3 for the larger datasets.

**Table 1.** Optimized Parameters

| **Algorithms** | **Parameters** |
|---|---|
| SVM | C = 80, Gamma = 0.01, Kernel = rbf |
| Logistic Regression | Algorithm = newton cg, Top k = 3, Median rate ratio = 0.8959 |
| GRU | Hidden Layer = 100, Activation function = ReLu, Dropout = 0.3,Loss function = Binary Crossentropy, Optimizer = Adam, Epoch = 10 |
| BERT | Batch size = 16, Epoch = 5 |

## 6 Result Analysis

The results from the classification models can be found in Table 2. As shown in the table, all of our models produced overall balanced outcomes, with SVM

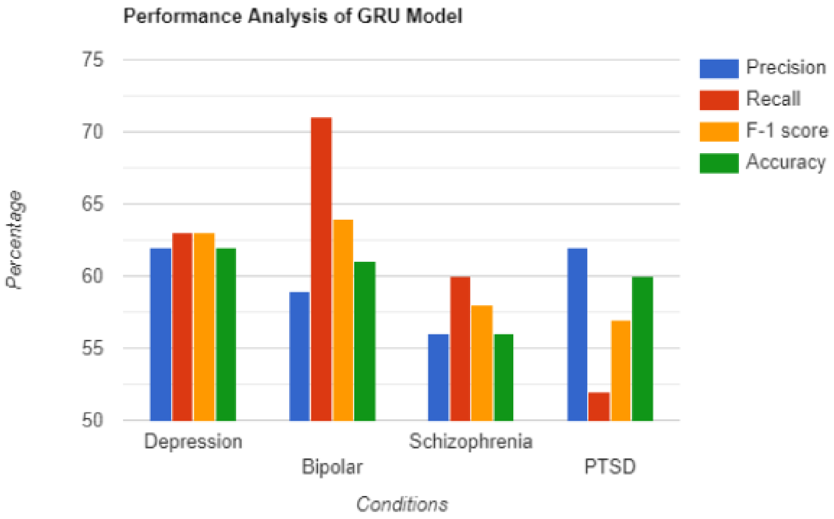

**Fig. 5.** Performance Analysis of GRU

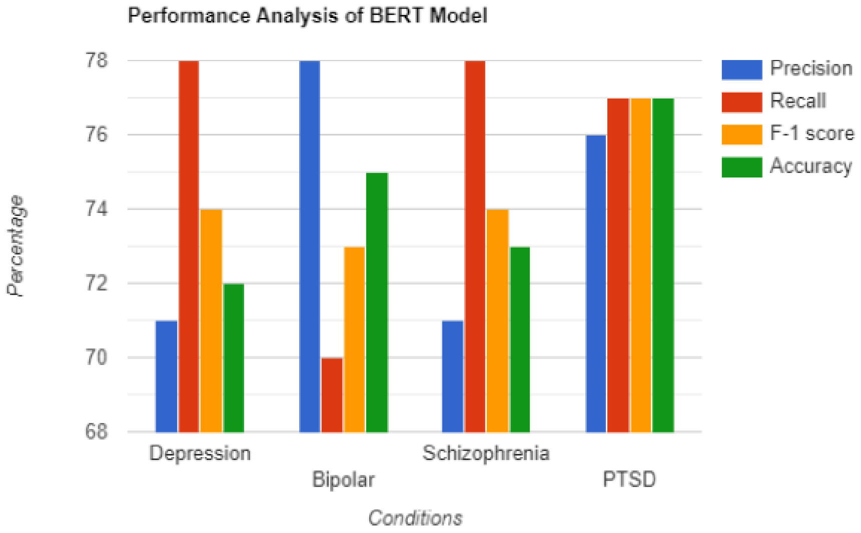

**Fig. 6.** Performance Analysis of BERT

producing the greatest overall values while BERT and GRU models had higher recall than precision in the majority of the illnesses. As discussed earlier SVM performs remarkably well in high dimensional data, whereas GRU and BERT had word embedding limitations due to scarce GPU capability. 512 tokens were feeded for BERT and 600 tokens for GRU and Logistic Regression to reduce computational cost, whereas 4000 max weighted features were fed into SVM. This enabled the model to learn more types of words used by the patients. Even so SVM performs a lot faster than any neural network and is capable of predicting results faster. Also risk of overfitting is less in SVM over Logistic Regression. A very important Key Performance Indicator was noticed to be "Recall" other than just Precision. Precision is basically the number of times the model was correct when the classifier predicted the "True" class whereas recall is the number times the classifier got it correct when the class was actually "True" in short higher recall value means lower type II error, which is why the authors were focusing more on recall than precision since someone who has the illness but is misdiagnosed as negative will be in more danger of the illness progressing than someone who does not have the illness but is misdiagnosed as positive.

**Table 2.** Result analysis of classification models

| Algorithms | Depression | Bipolar | Schizophrenia | PTSD |
|---|---|---|---|---|
| SVM & Bow features | P=84, R=78, F1=81, A=82 | P=84, R=78, F1=81, A=82 | P=84, R=78, F1=81, A=82 | P=84, R=78, F1=81, A=82 |
| Logistic Regression & Bow features | P=80, R=76, F1=78, A=79 | P=86, R=70, F1=77, A=87 | P=76, R=69, F1=72, A=82 | P=76, R=80, F1=76, A=87 |
| GRU | P=62, R=63, F1=63, A=62 | P=59, R=71, F1=64, A=61 | P=56, R=60, F1=58, A=56 | P=62, R=52, F1=57, A=60 |
| BERT | P=71, R=78, F1=74, A=72 | P=78, R=70, F1=73, A=75 | P=71, R=78, F1=74, A=73 | P=76, R=77, F1=77, A=77 |

## 7 Conclusion

The relentless advancement of Machine Learning over the years is truly astonishing. This work is dedicated to detect mental illness with the assistance of machine learning and deep learning models.The performance analysis of the models showed that the models are performed really good. Among all the models, the model built with BERT algorithm performed well for all the diseases. An overall balanced Key Performance Indicator (KPI) is achieved among all the models that have used but looking forward to improving those further. This research is limited due to our lack of access to high-end gear. A larger versions of the proposed models like BERT LARGE and LSTM with embedding techniques need to examine with.

## References

1. Tackling Mental Health Stigma in Bangladesh. ADD International. (n.d.). Retrieved March 23, 2022, from https://add.org.uk/tackling-mental-health-stigma-bangladesh
2. De Choudhury, M., Gamon, M., Counts, S., Horvitz, E, 2013. Predictingdepression via social media. In Proceedings of the International AAAIConference on Web and Social Media (Vol. 7, No. 1)
3. Yates, A., Cohan, A., Goharian, N. 2017. Depression and self-harm riskassessment in online forums. arXiv preprint arXiv:1709.01848.
4. Coppersmith, G., Dredze, M., Harman, C., Hollingshead, K. 2015. FromADHD to SAD: Analyzing the language of mental health in Twitterthrough self-reported diagnoses. In Proceedings of the 2nd workshop oncomputational linguistics and clinical psychology: from linguistic signalto clinical reality
5. Cohan, A., Desmet, B., Yates, A., Soldaini, L., MacAvaney, S.,Goharian, N. 2018. SMHD: a large-scale resource for exploring onlinelanguage usage for multiple mental health conditions. arXiv preprintarXiv:1806.05258.
6. Coppersmith, G., Dredze, M., Harman, C. 2014. Quantifyingmental health signals on Twitter. In Proceedings of the workshop oncomputational linguistics and clinical psychology: From linguistic signalto clinical reality.
7. Coppersmith, G., Harman, C., Dredze, M. 2014. Measuring posttraumatic stress disorder in Twitter. In Proceedings of the InternationalAAAI Conference on Web and Social Media.
8. Mowery, D. L., Park, Y. A., Bryan, C., Conway, M. 2016. Towardsautomatically classifying depressive symptoms from Twitter data forpopulation health. In Proceedings of the Workshop on ComputationalModeling of People's Opinions, Personality, and Emotions in SocialMedia (PEOPLES).
9. Ive, J., Gkotsis, G., Dutta, R., Stewart, R., Velupillai, S. 2018.Hierarchical neural model with attention mechanisms for the classificationof social media text related to mental health. In Proceedings of the FifthWorkshop on Computational Linguistics and Clinical Psychology: FromKeyboard to Clinic (pp. 69-77).
10. Losada, D. E., Crestani, F. 2016. A test collection for researchon depression and language use. In International Conference of theCross-Language Evaluation Forum for European Languages (pp. 28-39).Springer, Cham.
11. Diagnostic and statistical manual of mental disorders (DSM–5-TR). DSM-5. (n.d.). Retrieved April 6, 2022, from https://www.psychiatry.org/psychiatrists/practice/dsm
12. Pykes, K. (2020, May 1). Feature Engineering for Numerical Data. Medium. Retrieved April 6, 2022, from https://towardsdatascience.com/feature-engineering-for-numerical-data-e20167ec18
13. Stemming and lemmatization in python. DataCamp Community. (n.d.). Retrieved April 6, 2022, from https://www.datacamp.com/community/tutorials/stemming-lemmatization-python
14. Brownlee, J. (2020, June 27). How to encode text data for machine learning with scikit-learn. Machine Learning Mastery. Retrieved April 6, 2022, from https://machinelearningmastery.com/prepare-text-data-machine-learning-scikit-learn/