Mathematical and Computational Applications

MDPI

Article

A Deep Learning Approach to Unveil Types of Mental Illness by Analyzing Social Media Posts

Rajashree Dash *, Spandan Udgata, Rupesh K. Mohapatra, Vishanka Dash and Ashrita Das

Department of Computer Science & Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar 751030, Odisha, India; spandanudgata87654@gmail.com (S.U.); rkmohapatrar27@gmail.com (R.K.M.); vishankadash@gmail.com (V.D.); ashds187@gmail.com (A.D.)
* Correspondence: rajashreedash@soa.ac.in

Abstract: Mental illness has emerged as a widespread global health concern, often unnoticed and unspoken. In this era of digitization, social media has provided a prominent space for people to express their feelings and find solutions faster. Thus, this area of study with a sheer amount of information, which refers to users' behavioral attributes combined with the power of machine learning (ML), can be explored to make the entire diagnosis process smooth. In this study, an efficient ML model using Long Short-Term Memory (LSTM) is developed to determine the kind of mental illness a user may have using a random text made by the user on their social media. This study is based on natural language processing, where the prerequisites involve data collection from different social media sites and then pre-processing the collected data as per the requirements through stemming, lemmatization, stop word removal, etc. After examining the linguistic patterns of different social media posts, a reduced feature space is generated using appropriate feature engineering, which is further fed as input to the LSTM model to identify a type of mental illness. The performance of the proposed model is also compared with three other ML models, which includes using the full feature space and the reduced one. The optimal resulting model is selected by training and testing all of the models on the publicly available Reddit Mental Health Dataset. Overall, utilizing deep learning (DL) for mental health analysis can offer a promising avenue toward improved interventions, outcomes, and a better understanding of mental health issues at both the individual and population levels, aiding in decision-making processes.

Keywords: mental illness; social media; classification; LSTM; feature selection

check for updates

Academic Editor: Leonardo Trujillo

Received: 27 December 2024
Revised: 28 April 2025
Accepted: 29 April 2025
Published: 3 May 2025

Citation: Dash, R.; Udgata, S.; Mohapatra, R.K.; Dash, V.; Das, A. A Deep Learning Approach to Unveil Types of Mental Illness by Analyzing Social Media Posts. *Math. Comput. Appl.* **2025**, *30*, 49. https://doi.org/10.3390/mca30030049

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Mental illnesses, a multifaceted health concern, have evolved into a worldwide concern, inflicting significant suffering in people's lives [1,2]. They have traditionally been diagnosed through self-reporting and standardized questionnaires. However, nowadays, social media has become a unique platform for individuals to express their emotions and seek support more quickly [3–6]. These vast repositories of user-generated data on social media platforms offer a novel avenue for leveraging behavioral patterns to predict mental well-being. Such predictive capabilities could facilitate timely interventions from healthcare professionals and support networks, thereby mitigating the potential consequences of untreated mental distress.

The wealth of behavioral data on social media, coupled with advancements in machine learning (ML), can offer a promising opportunity to streamline and improve the diagnostic process for mental health disorders [4,5]. Several compelling reasons drive the exploration

of this research area. Firstly, the increasing prevalence of mental health issues necessitates more efficient and accessible diagnostic methods. Secondly, many individuals may not recognize or acknowledge their mental health challenges, making early intervention crucial. Thirdly, the vast amount of data generated on social media platforms can provide invaluable insights into behavioral patterns associated with mental health conditions. Harnessing these data could lead to personalized support and tailored interventions, enhancing overall mental health outcomes. Moreover, integrating machine learning into mental health analysis not only enhances diagnostic accuracy but also fosters a deeper understanding of mental health trends and risk factors [1–6].

Aldarwish et al. (2017) employed ML models like Support Vector Machine (SVM) and Naive Bayes (NB) to predict the presence of depression levels of a user by analyzing their social media posts [7]. Their study aimed to classify the severity of depression among users based on their online expressions by collecting all information related to a person's mood and negativism. In [8], SVM and NB were employed for the prediction of a user's mental health status by utilizing a dataset from Sina Weibo, a Chinese microblog service provider. Thorstad et al. (2019) investigated whether people's everyday language contains sufficient signals to predict the future occurrence of mental illness using language samples collected from the social media website Reddit [9]. The study highlights the impact of words drawn from nonclinical subreddits on distinguishing different categories of mental illness using an L2-penalized Logistic Regression (LR) classifier. Hussain et al. (2020) showed the effectiveness of an XGBoost model in comparison to NB and SVM in predicting four categories of mental illness by analyzing a person's posts and comments shared on the Reddit platform [10]. Ansari et al. (2021) analyzed the performance of Random Forest (RF), SVM, and LR with a data augmentation approach to handle the problem of insufficient labeled data during mental disorder classification and showed the importance of using data augmentation with classifiers by experimenting on two social media datasets [11]. Vaishnavi et al. (2022) assessed the accuracy of five ML techniques, of which the stacking-based technique obtained an accuracy of prediction of 81.75%, surpassing the Logistic Regression (LR), K-Nearest Neighbor (KNN), Decision Tree (DT), and Random Forest (RF) models [12].

Kim et al. (2020) developed a deep learning model using a Convolution Neural Network (CNN) to identify a user's mental state based on their posting information, behaviors, and interactions on social media, specifically on Reddit [13]. Six different binary classification models were developed, each targeting a specific mental disorder. The experimental results revealed that the CNN model outperforms the XGBoost model across all subreddits. Ang et al. (2023) analyzed the performance of a CNN to classify various mental illnesses across multiple social media platforms [14]. The model was trained and tested on labeled Twitter data and then validated on Reddit data and vice versa. The experimental findings demonstrate significant cross-platform generalization, with the model successfully classifying mental illnesses when tested on datasets unrelated to training. Uddin et al. (2022) [15] and Kour et al. (2022) [16] highlighted the effectiveness of DL models like LSTMs in capturing long-term dependencies within text data. This method allows for a more nuanced understanding of how users express themselves online, leading to better detection of depression symptoms. Murarka et al. (2020) explored the performance of an LSTM and two transformer-based models for the detection and classification of five prominent kinds of mental illnesses on social media, specifically Reddit [17]. The motivation for this study came into action during the COVID-19 pandemic, which reduced social contact between people close to zero, resulting in the isolation of many people who were unaware of their mental illnesses and felt the effects of that lockdown time. Narayanan et al. (2022) achieved high accuracy using a hybrid CNN-LSTM model, demonstrating the potential of combining

different deep learning architectures for mental illness detection [18]. Bokolo et al. (2023) employed ML models like SVM, Naive Bayes (NB), and advanced transformer models (BERT and RoBERTa) to classify users based on depression levels or mental illness categories [19]. These models analyze social media posts for signs of negativity, specific keywords, or sentiment patterns. Alkahtani et al. (2024) investigated the performance of two ML models and an LSTM model in assessing twelve types of mental health disorders by collecting a mental health dataset from Kaggle [20]. This research highlights the potential of LSTM over traditional ML models in improving the diagnosis and management of mental health disorders. An LSTM-based stress prediction model suggested by Bhavani and Naveen (2024) in [21] outperformed five ML models, namely SVM, KNN, LR, RF, and Adaboost, with a 100% accuracy rate on a self-prepared dataset created using a questionnaire. Instead of focusing on the diagnosis of a specific mental condition, Revathy et al. (2024) suggested an optimized RNN for classifying people as those needing treatment and those not needing treatment by extracting features from the OSMI dataset [22]. The RNN model is fed the time domain and statistics features that were obtained by the dual domain feature extraction technique. With an accuracy of 98% in identifying mental illness, the model performs better than CNN and SVM models. In [23], a deep learning model with a simple architecture and transformer demonstrated efficacy in identifying suicidal thoughts and depression from social media content. Zhu et al. (2024) offered another application of LSTM as a binary classifier for the detection of mental health status by extracting features and patterns from text data on social media [24]. Rather than emphasizing the identification of a specific kind of mental disorder, they used LSTM to categorize whether or not a person has mental illness. Bendebane et al. (2024) proposed a CNN-GRU-based multiclass model to analyze tweets to identify anxiety and depression disorders [25]. With an accuracy of up to 93%, the model outperformed binary models that predict anxiety and depression independently. A four-layer CNN hybridized with two pre-trained transformers, namely MentalBERT and MelBERT, was suggested by Karamat et al. (2024) for multiclass mental disease identification using social media data [26]. The performance of the CNN was also compared by cascading it with two other transformers, namely BERT and RoBERTa, and a BiLSTM-based feature extractor. They employed a random down-sampling technique to handle the data imbalance problem. Comparing the model with other transformers and the BiLSTM-CNN revealed the superior performance of the proposed model with an overall accuracy of 92%. In order to facilitate mental health analysis, Yang et al. (2024) introduced a multi-task and multi-source interpretable mental health instruction (IMHI) dataset with 105 k samples in [27]. The dataset was prepared by gathering raw data from several social media platforms, including Reddit, Twitter, and Short Message Service texts, and then including explanations for the collected samples that were generated using ChatGPT-4. The authors fine-tuned a MentaLLaMa model on the IMHI dataset to perform eight mental health analysis tasks, including both binary and multiclass analyses. The model addresses the identification of up to five different types of mental disorders. With a batch size of 2556 and a maximum model input length of 20,248, the model was trained across 10 epochs. An evaluation of the IMHI dataset using the weighted F1-score and BART score demonstrated that the suggested model approaches the MentalBERT and MentalRoBERT models and generates human-level explanations.

Table 1 presents a brief summary of the ML techniques explored in identifying different types of mental illnesses via social media posts.

Table 1. A brief summary of various research papers related to ML and mental health.

Reference	Techniques Used	Dataset Used	Evaluation Metrics Used	Type of Classification	Research Limitations
[7]	SVM, NB	Facebook, Live Journal, and Twitter dataset	Accuracy, Precision, Recall	Binary	• Used only for detecting depression • Did not explore any FS or data balancing approaches
[8]	SVM, NB	Sina Weibo dataset	RAE, RRSE, Pearson Correlation Coefficient, Accuracy	Binary	• Only identified mental status as good or not good • Not focused on identifying particular mental illness • Did not explore any FS or data balancing approaches
[9]	L2-Penalized LR	Reddit dataset	Accuracy, Precision, Recall, F1-score, Clustering Analysis	Multiclass (4 classes)	• Did not explore any FS approaches • Not validated with any other ML models or deep learning models
[10]	XGBoost, NB, SVM	Reddit dataset	Accuracy, Precision, Recall, F1-score	Multiclass (4 classes)	• Did not explore any data balancing approaches
[11]	RF, SVM, LR	Reddit dataset and SDCNL dataset	Precision, Recall, F1-score, *t*-test, *p*-value	Binary	• Model is separately used to identify stress or depression; not used to identify other mental illnesses • Did not explore any FS approaches • Not validated with any other deep learning models
[12]	LR, KNN, Decision Tree, RF,	Reddit dataset	Accuracy, ROC Curve	Binary	• Only identified mental status as good or not good • Not focused on identifying particular mental illness • Did not explore any FS or data balancing approaches

Table 1. *Cont.*

Reference	Techniques Used	Dataset Used	Evaluation Metrics Used	Type of Classification	Research Limitations
[13]	XGBoost, CNN	Reddit dataset	Accuracy, Precision, Recall, F1-score	Binary	• Multiple binary classification models were developed to identify 6 types of mental illness • Did not explore any FS approaches
[14]	CNN	Twitter, Reddit	Accuracy, Precision, Recall, F1-score	Binary	• Six separate binary classification models were developed, each one being used to identify 1 out of 6 types of mental illness • Did not explore any FS and data balancing approaches
[15]	LSTM, RNN	Norwegian information website ung.no	Accuracy, Precision, Recall, F1-score, Support	Binary	• Used only for detecting depression • Did not explore impact of data balancing
[16]	CNN, RNN, LSTM, CNN-biLSTM	Twitter dataset	Accuracy, Precision, Recall, F1-score, Error Rate, AUC curve	Binary	• Used only for detecting depression • Did not explore any FS or data balancing approaches
[17]	LSTM, BERT, RoBERTa	Reddit dataset	Accuracy, Precision, Recall, F1-score	Multiclass (6 classes)	• Did not explore any FS or data balancing approaches • Did not handle multilabel classification
[18]	CNN, LSTM, Hybrid CNN-LSTM	Twitter dataset	Accuracy, Precision, Recall, F1-score, Support	Binary	• Used only for detecting depression • Did not explore any FS or data balancing approaches
[19]	LR, NB, RF, BERT, RoBERT	Twitter dataset	Accuracy, Precision, Recall, F1-score, Confusion Matrix Diagram	Binary	• Used only for detecting depression • Did not explore any FS or data balancing approaches

Table 1. *Cont.*

Reference	Techniques Used	Dataset Used	Evaluation Metrics Used	Type of Classification	Research Limitations
[20]	KNN, RF, LSTM	Mental health disorder dataset Kaggle	Accuracy, Precision, Recall, F1-score	Multiclass (12 classes)	• Did not explore any FS or data balancing approaches
[21]	LSTM, KNN, RF, SVM, LR, ADABoost	Self-prepared dataset through questionnaires	Accuracy, Precision, Recall, F1-score	Binary	• Used only for detecting stress • Did not explore any FS or data balancing approaches
[22]	RNN, CNN, SVM	OSMI dataset	Accuracy, F1-score	Binary	• Instead of focusing on particular disorder identification, used to classify people as those needing treatment and those not needing treatment • Did not explore any data balancing approaches
[23]	RNN, CNN, LSTM, BERT, SVM, NB, RF, LR, DT	Reddit dataset	AUC, Precision, Recall, F1-score	Binary	• Used only for detecting depression • Did not explore any FS or data balancing approaches
[24]	LSTM	Self-prepared dataset	Accuracy, Precision, Recall	Binary	• Instead of focusing on particular disorder identification, used to classify mental health status • Did not explore any FS or data balancing approaches • Not validated with any other ML or DL models
[25]	CNN-BiGRU, CNN-BiLSTM CNN-BiRNN, CNN-GRU, CNN-LSTM, CNN-RNN	Twitter dataset	Accuracy, F1-score	Multiclass (3 classes)	• Used only for detecting depression and anxiety disorders • Did not explore any FS or data balancing approaches

Table 1. *Cont.*

Reference	Techniques Used	Dataset Used	Evaluation Metrics Used	Type of Classification	Research Limitations
[26]	Hybrid transformer (MentalBERT/MelBERT)-based CNN, BiLSTM-CNNBERT/RoBERTa (CNN)	Reddit dataset	Accuracy, Precision, Recall, F1-score	Multiclass (4 classes)	• Used only for detecting depression, anxiety, BPD, PTSD • Used random down-sampling for data balancing and not validated with other data balancing approaches • Not validated with any other ML or DL models
[27]	LLaMA, BERT, RoBERTa, MentalBERT, MentalRoBERTa	IMHI dataset	Weighted F1-score, BART Score	Multiclass (6 classes)	• Did not explore any FS or data balancing approaches • Lacking more domain-specific knowledge in training model • BART score exhibits moderate correlations to human evaluation results, which limits reliability of model

From the survey, it is seen that few studies examined the performance of DNNs with different data balancing and feature selection approaches for mental illness prediction. Most of the approaches used in mental health research often include binary distinctions. Multiclass classification could offer a more nuanced understanding by recognizing multiple co-occurring disorders or varying severity levels within a single diagnosis. To overcome these research gaps, this research endeavor aims to develop a sophisticated deep learning model capable of identifying a wide array of mental disorders, including depression, anxiety, bipolar disorder, Post-Traumatic Stress Disorder (PTSD), schizophrenia, suicidal ideation, and a neutral category. The proposed methodology leverages the power of Long-Short Term Memory (LSTM) architecture, enhanced with data balancing techniques, within a sophisticated multiclass classification framework. Unlike many studies that focus on detecting the presence or absence of a single mental health condition, this project aims to classify users into a range of potential mental illnesses with the nuanced expressions and linguistic patterns found in user-generated text. Recognizing the challenges posed by imbalanced datasets in mental health research, this study also investigates the impact of data balancing techniques on model performance by using different target values. The effectiveness of the LSTM-based model is validated by comparing its performance against three traditional ML models: LR, RF, and Multinomial Naive Bayes' (MNB). Furthermore, to bridge the gap between textual data and machine learning algorithms, two prominent vectorization techniques, CountVectorizer and Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer, are utilized in the experimentation. The performance of all the models is evaluated both with and without feature selection using the Chi-square technique over a comprehensive dataset named the "Reddit Mental Health Dataset", comprising a diverse collection of user posts from mental health communities on the popular platform of Reddit. By analyzing this rich dataset, we seek to determine whether a user exhibits linguistic markers indicative of a specific mental illness.

The main contributions are listed as follows:

- A multiclass classification framework is developed using LSTM to identify a wide array of mental disorders, including depression, anxiety, bipolar disorder, Post-Traumatic Stress Disorder (PTSD), schizophrenia, suicidal ideation, and a neutral category.
- The effectiveness of the LSTM-based model is also validated by comparing its performance against three traditional ML models, LR, RF, and MNB.
- Along with feature engineering, this study also examines how data balancing improves LSTM's performance in handling the multiclass classification problem with an unbalanced dataset.

The next section provides a structured schematic layout of the deployed model along with a comprehensive overview of the methodologies employed. Section 3 provides insights into the system's specifications, parameters, and functions employed during model implementation and the analysis of the outcomes of different models. Finally, Section 4 concludes the work, acknowledging the limitations of the study and outlining the potential avenues for future research directions.

2. Materials and Methods

This section begins with a detailed description of data sources and preprocessing steps. It encompasses the application and methodology of all the models.

2.1. Dataset Description

This study uses a publicly available dataset named the "Reddit Mental Health Dataset" from the social networking site called "Reddit" for analysis, validation, and model creation. Reddit is a social media platform comprising user-generated content organized into themed

communities called "subreddits" [28]. Users can post, share, and discuss content ranging from news and entertainment to niche interests. The original dataset contains posts from 28 subreddits from 2018 to 2020. We use a small piece of data for analysis from the original dataset that comprises 6 different classes of mental illnesses, which are anxiety, bipolar disorder, depression, PTSD, schizophrenia, and suicide, and a set of neutral data classes for the year 2019 only. Table 2 shows the number of posts from each class in the dataset. A pie chart representing the distribution of the number of posts in each class is shown in Figure 1.

Table 2. Number of posts in each class.

Name of Class	No. of Posts
Neutral	9628
Anxiety	13,211
Bipolar Disorder	910
Depression	33,549
PTSD	1397
Schizophrenia	1548
Suicide	14,581
Total	74,824

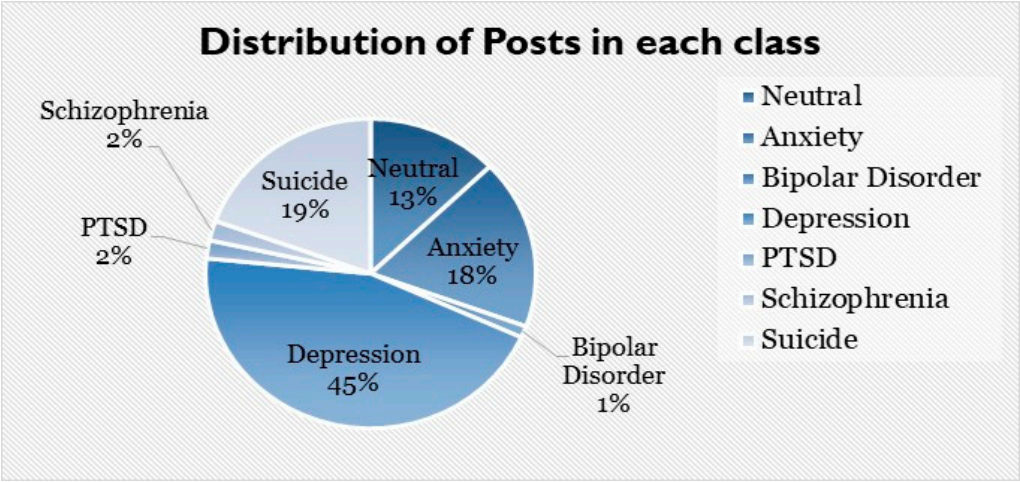

Figure 1. A pie chart representing the distribution of the number of posts in each class.

2.2. Proposed Model for Predicting Type of Mental Illness Through Social Media Post Analysis

Figure 2 shows a schematic layout of the proposed model. The model starts with data preprocessing, a crucial step for text analysis. Here, social media data undergo cleaning to remove noise and inconsistencies. This includes eliminating unnecessary characters, abbreviations, emojis, and punctuation. Additionally, stop words and duplicate entries are removed, and spelling errors are corrected. Lemmatization and tokenization further refine the data by converting words to their base forms and splitting text into individual units.

To transform text into numerical data suitable for machine learning algorithms, two techniques are used: the count vectorizer and TF-IDF vectorizer. The count vectorizer creates a matrix based on word frequencies, while TF-IDF assigns weights to words based on their importance within a document and across the entire dataset. Three traditional ML models, namely LR, RF, and MNB, are initially implemented, followed by an LSTM model for multiclass classification. Following that, the effects of feature selection (using

Chi-square) and class balancing techniques on the performance of both traditional ML models and the LSTM model designed for multiclass classification purposes are evaluated. In pursuit of optimal model performance, the model undergoes evaluation both with and without feature selection using the Chi-square technique. Chi-square FS serves as a valuable tool for identifying and retaining the most relevant features by assessing the statistical dependence between features and the target variable (mental health category). By focusing on the most informative aspects of the text data, this technique can substantially improve the model's ability to discern patterns and make accurate predictions. Moreover, the pervasive issue of class imbalance, frequently encountered in social media data, is explicitly addressed after selecting relevant features. Table 2 makes clear how imbalanced the dataset used in the study is. So, the model is rigorously tested on both unbalanced and balanced datasets to ensure robustness and fairness.

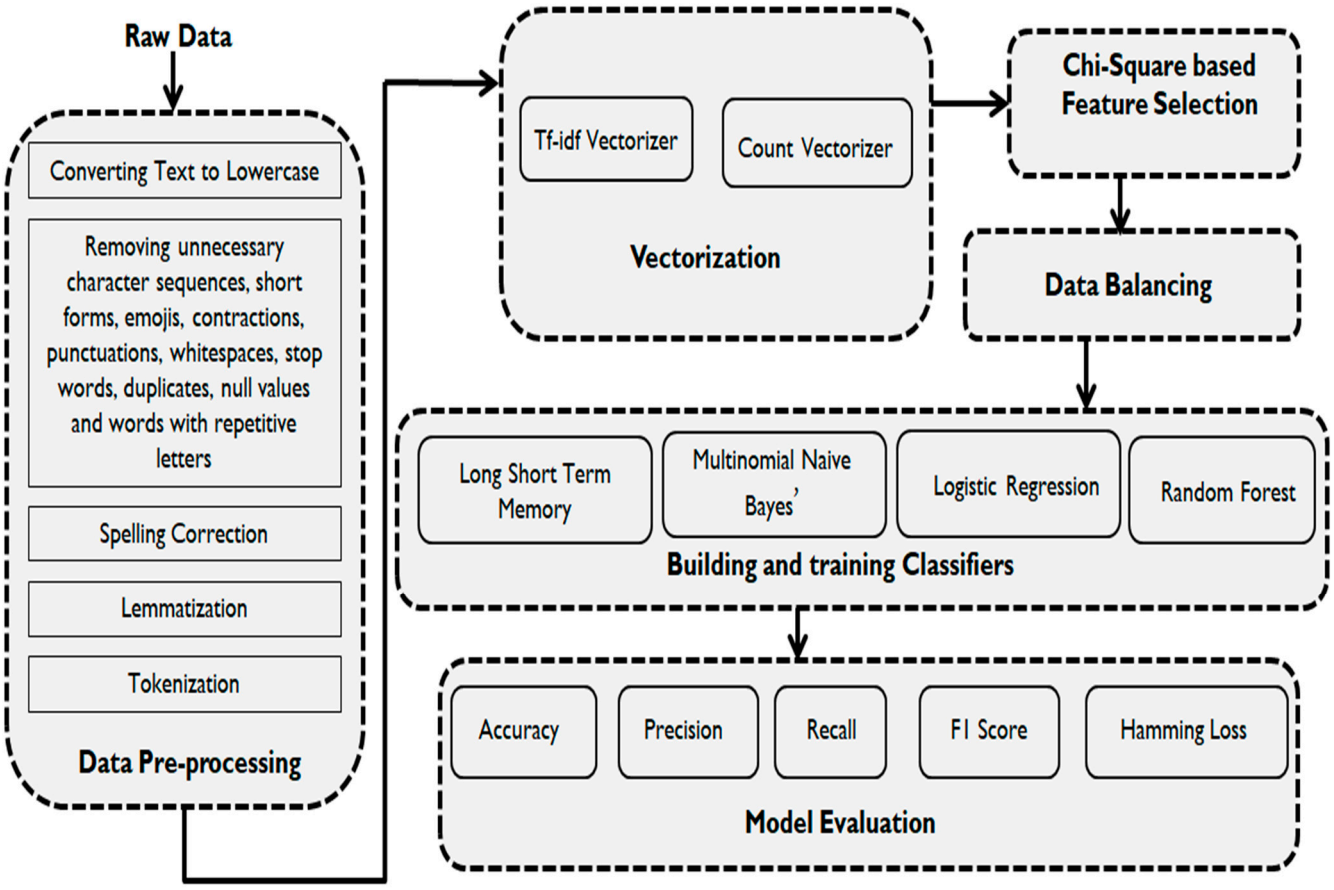

Figure 2. Schematic layout of proposed model.

Resampling is a popular data-level strategy for addressing the class imbalance problem in numerous studies [29,30]. It includes undersampling, oversampling, and hybrid strategies that attempt to balance out the dataset either randomly or deterministically [29]. With the undersampling strategy, a subset of the original dataset is generated by randomly or selectively deleting some of the samples of the majority class while retaining the original minority class [9,26,29]. By employing random oversampling (ROS) to replicate samples of the minority class [29,30] or the synthetic minority oversampling technique (SMOTE) [31] to create new synthetic samples in the minority class, the oversampling strategy produces a superset of the original set. Through the inclusion of more replicated minority samples, ROS alone can raise the probability of overfitting. Likewise, random undersampling (RUS) alone may result in the loss of valuable data due to the removal of a large proportion of samples. Both strategies can be helpful in certain situations, and neither of them is

superior to the other. Additionally, the skewness in the target variable's class proportions determines how effective imbalance learning strategies are in classification problems. This can lead to the emergence of a family of hybrid data balancing techniques that combine both under and oversampling [32–35]. A novel hybrid sampling technique combining oversampling and undersampling is presented in [32] to address the class imbalance problem in which undersampling is applied first to eliminate some samples of the majority class with less classification information, and then oversampling is applied to gradually produce some new positive samples. A hybrid balancing method integrating synthetic majority undersampling with SMOTE oversampling is proposed in [33] with an emphasis on binary classification tasks. Another hybrid data balancing approach combining ROS with the neighborhood cleaning rule (NCL) for discarding instances from the majority class is suggested in [34] to address the class imbalance issue related to a multiclass text classification task. As demonstrated by an evaluation of ROS, RUS with a hybrid data balancing method combining SMOTE with RUS over RF, conducted in [35], the efficiency of the hybrid data balancing method is evident for educational data mining-based classification tasks. Motivated by the application of hybrid data balancing techniques in [32–35], and to allow for a more balanced trade-off between maintaining information and reducing bias, in this study, two hybrid balancing techniques are strategically employed. In the first hybrid approach, random oversampling is used to increase the number of samples in the minority class, and random undersampling is used to reduce the number of majority class instances to balance the dataset. In the second hybrid approach, SMOTE-based oversampling is combined with undersampling to balance the dataset. Hybrid balancing techniques are also compared with the RUS strategy. Through simulation, the number of documents to be retained for each class is determined.

LSTMs may learn long-term dependencies, which makes them appropriate for sequential data like text. Integer encoding with an inbuilt TensorFlow library is used to vectorize the words. After determining the average length of each row/post, the maximum sequence length is set to 250. Each word is converted into random numbers; if a row is shorter than 250, it is padded with zeros; if longer, it is truncated. This ensures uniform column size. Unique words are extracted, and a pre-trained word-embedding matrix called Global Vector (GloVe) is used to convert each word in the post to a 200-dimensional vector. This results in a 40,122 $\times$ 200 embedding matrix. This matrix is passed to the embedding layer, which applies it to the posts, converting the entire dataset into a 3D matrix with a size of 59,859 $\times$ 250 $\times$ 200. Here, 59,859 is the batch size or sample size, 250 is the sequence number or timestamp for the LSTM, and 200 represents the features. In order to prevent overfitting, a dropout layer is implemented before the LSTM layer. During training, it randomly sets a portion of the input units to 0 at each update. Sequence-wise, each word, represented by a 200-dimensional vector, is fed into the model. The LSTM processes and trains on these data, capturing temporal dependencies and relationships in the sequence. After the LSTM layer, the data are passed to a dense layer, which performs the final classification based on the learned features. Figure 3 provides a visual representation of the overall LSTM model architecture used for mental illness prediction, while Figure 4 delves deeper into the internal architecture of an individual LSTM unit.

The LSTM layer operates on sequences one step at a time, following a structured process:

Input and Hidden State Combination: At each step, the current word embedding layer is combined with the hidden state obtained from the previous step. This integration ensures that the model considers both the current input and the contextual information from prior words.

Gate Control and Information Flow: The LSTM unit's internal gates (forget gate, input gate, and output gate) play a pivotal role in controlling the flow of information within the cell.

These gates determine which aspects of the past are relevant to remember and which can be forgotten, allowing for selective memory retention.

New Hidden State Generation: The LSTM unit generates a new hidden state that encapsulates the most pertinent information from the current word and the contextual cues derived from previous words. This new hidden state then serves as the input for the next step in the sequence, perpetuating the learning process.

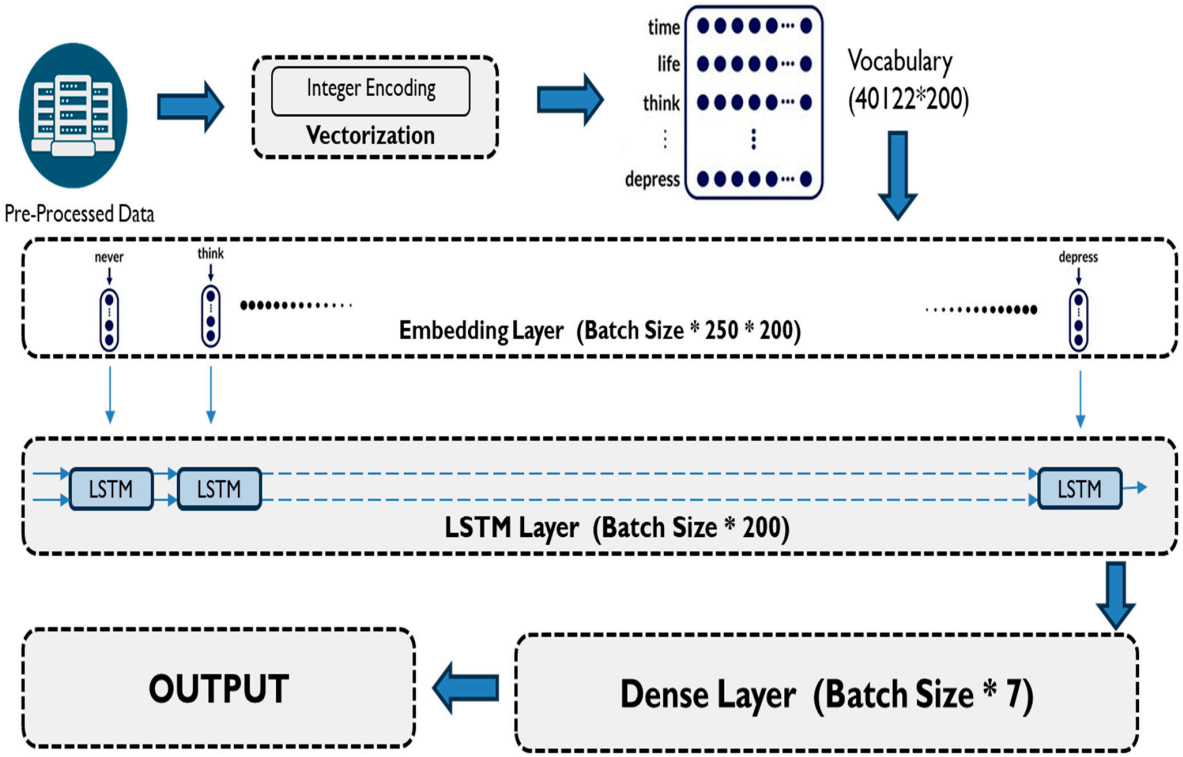

Figure 3. Schematic layout of LSTM model.

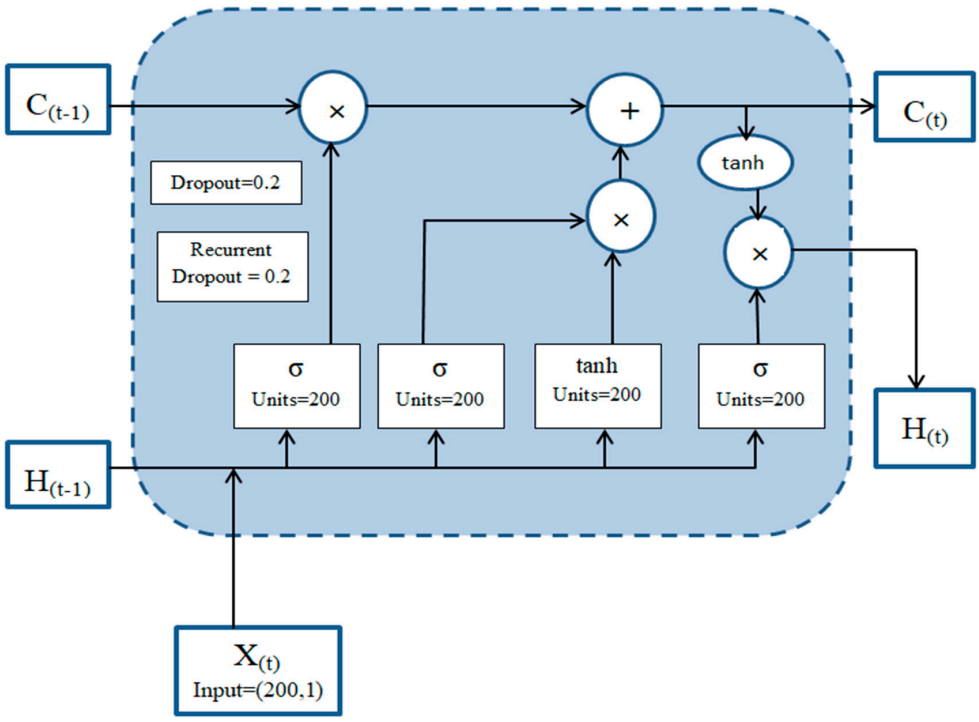

Figure 4. Internal architecture of LSTM unit.

After the entire sequence is processed, the final hidden state with a size of 200 effectively summarizes the high-level features extracted from the sequence. The dense layer receives the output from the previous layer, which is the final hidden state of the LSTM layer. It acts as a fully connected layer. This means that every neuron in this layer is connected to all neurons in the previous layer, the LSTM layer. Each connection has a weight associated with it. These weights, along with a bias term for each neuron in the dense layer, are the parameters the model learned during training. For each neuron in the dense layer, it takes a weighted sum of the elements in the input vector (the final hidden state from the LSTM). This weighted sum essentially combines the information extracted from the sequence by the LSTM layer. The weighted sum is then passed through an activation function, i.e., softmax, which ensures the output of the dense layer lies between 0 and 1, and the sum of all outputs is 1. This makes the output a valid probability distribution for multiclass classification. The output of the dense layer is a vector of size 7, where each element represents the probability of the input belonging to a specific class. Thus, the model predicts the class with the highest predicted probability.

Finally, the performance of the model is accessed using accuracy, Hamming loss, and the micro-averaged values of precision, recall, and the F1-score, as calculated using Equations (1)–(5), respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$\text{Hamming loss} = \frac{1}{\text{NL}} \sum_{i=1}^{\text{N}} \sum_{j=1}^{\text{L}} \text{XOR}\left(\text{target}_{ij}, \text{predicted}_{ij}\right) \tag{2}$$

$$\text{Precision} = \frac{\sum_{i=1}^{\text{C}} \text{tp}_i}{\sum_{i=1}^{\text{C}} \text{tp}_i + \text{fp}_i} \tag{3}$$

$$\text{Recall} = \frac{\sum_{i=1}^{\text{C}} \text{tp}_i}{\sum_{i=1}^{\text{C}} \text{tp}_i + \text{fn}_i} \tag{4}$$

$$\text{F1-score} = \frac{2 * \text{p}_{\text{micro}} * \text{r}_{\text{micro}}}{\text{p}_{\text{micro}} + \text{r}_{\text{micro}}} \tag{5}$$

3. Experimentation and Results

This section outlines the system's specifications, parameters configured for training ML models, and the findings obtained from all the experimentations. All experiments were conducted using Python 3.11 on an Intel Core i5 9th Gen system with 8 GB RAM and a Windows 11 opearting system. Table 3 details the parameters used for different ML classifiers.

For the LSTM model, the input dimension of the embedding layer was set to 40,122, representing the size of the vocabulary, and its output dimension was set to 200. The embeddings were initialized with the pre-trained GloVe vectors, allowing the embeddings to be updated during training. A dropout rate of 0.2 was used in the spatial dropout layer. The number of units of the LSTM layer was set to 200, with its dropout rate and recurrent dropout rate both set to 0.2. Following the LSTM layer, a dense layer comprising seven neurons with a softmax activation function was used for the classification task. The number of training samples to be used in each iteration was specified by a batch size of 128. Training was conducted over 50 epochs, selecting the Adam optimizer, and the categorical_crossentropy loss function with an "Earlystopping" callback to stop the training automatically when the model's performance stopped improving. For the LR model, the regularization strength parameter was set to 1, selecting the 'lbfgs' solver with 100 iterations. The number of estimators of the RF model was set to 100 with a min sample

split value of 2. The performance of the MNB model was assessed by setting the smoothing parameter to 1 and the FitPrior flag to true. To maintain data distribution homogeneity and facilitate fair model comparison, all experiments related to this study were carried out using a uniform dataset.

Table 3. Parameter specifications of ML classifiers.

Classifier	Parameters
LR	• C (Regularization strength): 1.0 • Solver: 'lbfgs' • Max Iterations: 100
RF	• Number of Estimators: 100 • Max Depth: None (nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples) • Min Samples Split: 2
MNB	• Alpha (smoothing parameter): 1.0 • Fit Prior: True
LSTM	• Embedding Layer: ▪ Input Dimension (vocab_size): 40,122 (size of vocabulary) ▪ Output Dimension (embid_dim): 200 (dimension of dense embedding) ▪ Embeddings Initializer: Initialized with pre-trained GloVe vectors ▪ Trainable: True (embeddings will be updated during training) • Spatial Dropout Layer: 0.2 (20% of input units to drop) • LSTM Layer: ▪ Units: 200 (number of LSTM units) ▪ Dropout: 0.2 (20% of input units to drop) ▪ Recurrent Dropout: 0.2 (20% of connections between recurrent units to drop) • Dense Layer: ▪ Units: 7 (number of output classes) ▪ Activation Function: 'softmax' (for multiclass classification) • Compile Method: ▪ Loss Function: 'categorical_crossentropy' (for multiclass classification) ▪ Optimizer: 'adam' (adaptive moment estimation) ▪ Epoch: 50 ▪ Batch Size: 128

After pre-processing, the original dataset was converted to a matrix of size of 74,284 × 4, including columns such as Post, Lemmatized_text, Text_token, and Labels, representing the categories of the posts. The dataset was unbalanced, as the class with the most number of posts had 33,549 posts, while the class with the least number of posts had just 910. To address this issue, different class balancing techniques such as SMOTE and random sampling were tried and tested, and hybrid balancing (random oversampling + random undersampling) resulted in 12,000 posts for each class, with 84,000 proving to be efficient. The dataset was split into a ratio of 80:20, where 80% was used for training and 20% was used for testing. The dataset description after performing data balancing is shown in Table 4. As per the literature survey, the most frequently used evaluation metrics are

accuracy, precision, recall, f1-score, and hamming loss, which are used in this paper for result analysis.

Table 4. Number of posts in each class in training and test sets after using hybrid balancing (random oversampling + random undersampling).

Name of Class	No. of Posts	No. of Posts (Training)	No. of Posts (Testing)
Neutral	12,000	9621	2379
Anxiety	12,000	9652	2348
Bipolar Disorder	12,000	9589	2411
Depression	12,000	9577	2423
PTSD	12,000	9589	2411
Schizophrenia	12,000	9581	2419
Suicide	12,000	9591	2409
Total	84,000	67,200	16,800

Table 5 illustrates the performance metrics of LR, MNB, and RF using two different vectorizers, the count and TF-IDF vectorizers, without using the feature selection technique. As per the results, LR performs well across both vectorizers with greater accuracy and balanced F1-scores. The TF-IDF vectorizer tends to slightly outperform the count vectorizer in most cases, indicating its effectiveness in capturing the importance of terms.

Table 5. Comparison of traditional ML models without feature selection and without data balancing.

Classifier	Vectorizer	Accuracy	Precision	Recall	F1-Score	Hamming Loss
LR	TF-IDF	0.74	0.81	0.62	0.68	0.25
MNB	TF-IDF	0.52	0.58	0.22	0.22	0.47
RF	TF-IDF	0.66	0.62	0.38	0.38	0.33
LR	Count	0.73	0.74	0.66	0.69	0.26
MNB	Count	0.67	0.72	0.49	0.55	0.32
RF	Count	0.66	0.58	0.37	0.37	0.33

Table 6 presents a comparison of all three traditional ML models with Chi-square-based feature selection. The performance of all three models is improved by using the feature selection technique. When analyzing the results in Tables 5 and 6, it is observed that LR consistently exhibited the highest accuracy among all models and also showed better results in other metrics. Table 7 displays the comparative results of the best-found traditional ML model, i.e., LR with a TF-IDF vectorizer and Chi-square based feature selection, with the LSTM model. Among all models, LSTM with the pre-trained embedding matrix GloVe showed fine results. LSTM shows potential in leveraging specialized embeddings to understand and classify text data more accurately, emphasizing the importance of tailored feature engineering and model optimization in complex natural language processing tasks.

Furthermore, the performance of the LSTM model is analyzed with different data balancing techniques, and the outcomes are presented in Table 8. From the results, it is clear that the hybrid data balancing technique using random sampling in oversampling and undersampling showed impressive results. Figure 5 presents the performance comparison of LSTM and LR with a reduced feature set and on the same balanced dataset, also revealing the better performance of LSTM over LR on the balanced dataset. With data balancing, the accuracy of LSTM and LR is improved by 14.28% and 13%, respectively. Similarly, there are

improvements of 50% and 37.5% in the Hamming loss of LSTM and LR, respectively, by using data balancing.

Table 6. Comparison of traditional ML models with Chi-square-based FS.

Classifier	Vectorizer	Accuracy	Precision	Recall	F1-Score	Hamming Loss
LR	TF-IDF	0.75	0.82	0.63	0.69	0.24
MNB	TF-IDF	0.57	0.55	0.27	0.28	0.42
RF	TF-IDF	0.67	0.66	0.40	0.41	0.32
LR	Count	0.73	0.75	0.66	0.70	0.26
MNB	Count	0.68	0.67	0.60	0.63	0.31
RF	Count	0.67	0.59	0.40	0.40	0.32

Table 7. Performance comparison of best found ML model with LSTM model using FS.

Model	Number of Features	Accuracy	Precision	Recall	F1-Score	Hamming Loss
LR + TF-IDF + Chi-square	15,000	0.75	0.82	0.63	0.69	0.24
LR + TF-IDF + Chi-square	250	0.73	0.79	0.59	0.65	27.10
LSTM + TF-IDF + Chi-square	250	0.45	0.06	0.14	0.09	0.20
LSTM with pre-trained embedding matrix	200	0.77	0.77	0.70	0.73	0.08

Table 8. Performance comparison of LSTM model with and without data balancing.

Data Balancing Techniques	Number of Documents per Class	Accuracy	Precision	Recall	F1-Score	Hamming Loss
Without data balancing	(9628, 13,211, 910, 33,549, 1397, 1548, 14,581)	0.77	0.77	0.70	0.73	0.08
Hybrid (random oversampling + random undersampling)	12,000	0.88	0.88	0.88	0.88	0.04
Hybrid (oversampling using SMOTE + random undersampling)	8000	0.63	0.62	0.63	0.62	0.11
Undersampling	910	0.74	0.75	0.74	0.74	0.08

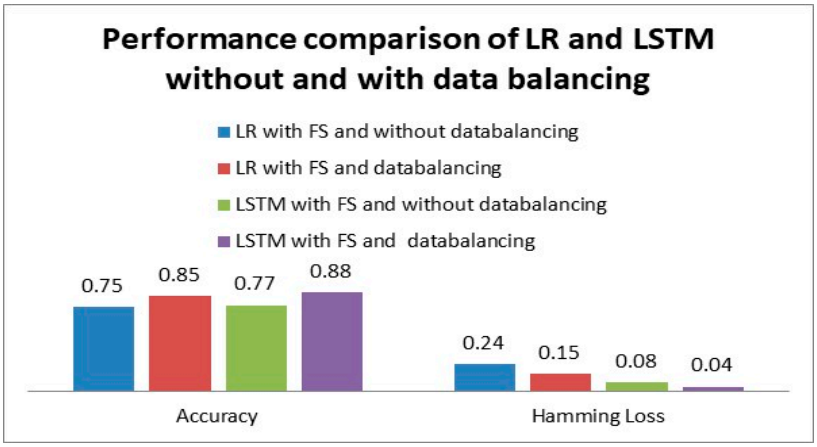

Figure 5. Performance comparison of LSTM and LR with reduced feature set and same balanced dataset.

Figures 6 and 7 present a visual comparison of confusion matrices for the LSTM model with and without data balancing. Without balancing, the model exhibits a strong bias towards predicting "depression", evident in the large diagonal value. However, it struggles with less frequent classes like "suicide" and "schizophrenia", often misclassifying them as "depression". In contrast, with balancing, the model demonstrates a more even distribution of predictions across classes. Notably, the "depression" diagonal is smaller, and predictions for "suicide" and "schizophrenia" improve, indicating reduced bias and enhanced performance in minority classes. This suggests that data balancing mitigates the model's tendency to favor the majority class, leading to better overall classification across all categories.

Figures 8 and 9 illustrate the training performance of the LSTM model with and without data balancing, focusing on accuracy and categorical cross-entropy loss across epochs, respectively. In Figure 8, both models show a rapid increase in accuracy during the initial epochs, with the balanced model slightly outperforming the unbalanced model. However, both models plateau around epoch 5, with the balanced model maintaining a consistent edge in accuracy throughout the remaining epochs. This suggests that balancing aids in faster convergence and a higher final accuracy. Figure 9 reveals a stark contrast in categorical cross-entropy loss. The unbalanced model starts with a higher loss, indicating more multi-label prediction errors. While both models experience a decrease in loss over epochs, the balanced model demonstrates a significantly lower loss throughout. This implies that balancing considerably reduces multi-label prediction errors, leading to more accurate classifications, especially for less frequent classes. Overall, the figures emphasize the positive impact of data balancing on the LSTM model's training. It not only leads to faster convergence and slightly higher accuracy but also substantially reduces.

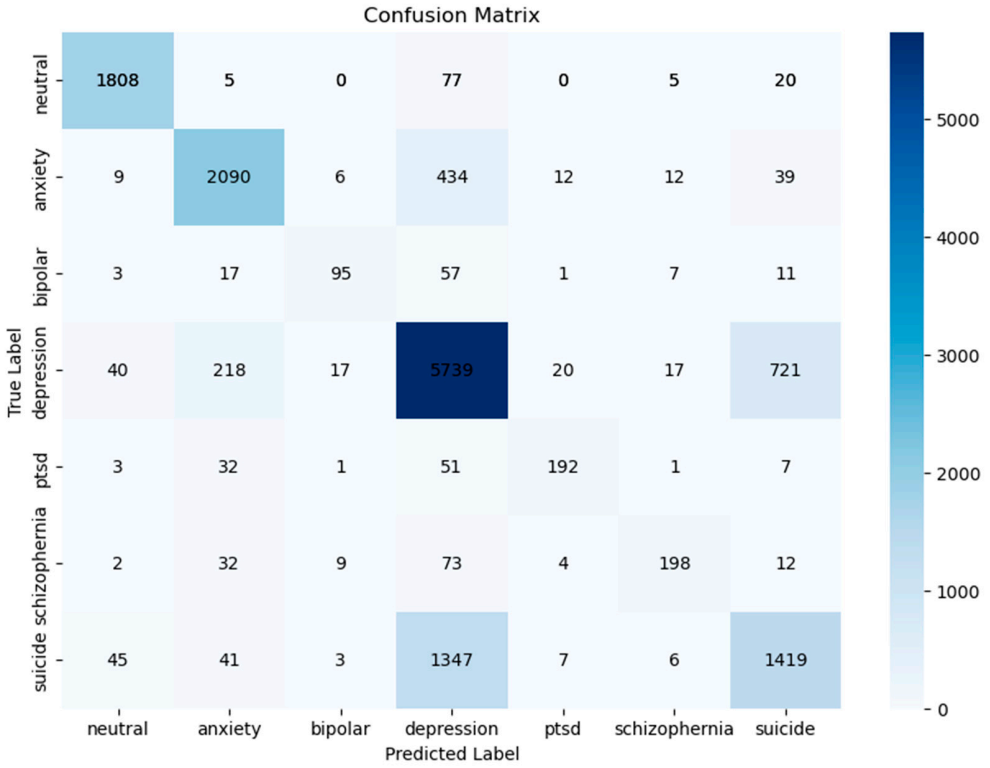

Figure 6. Confusion matrix of LSTM model without data balancing.

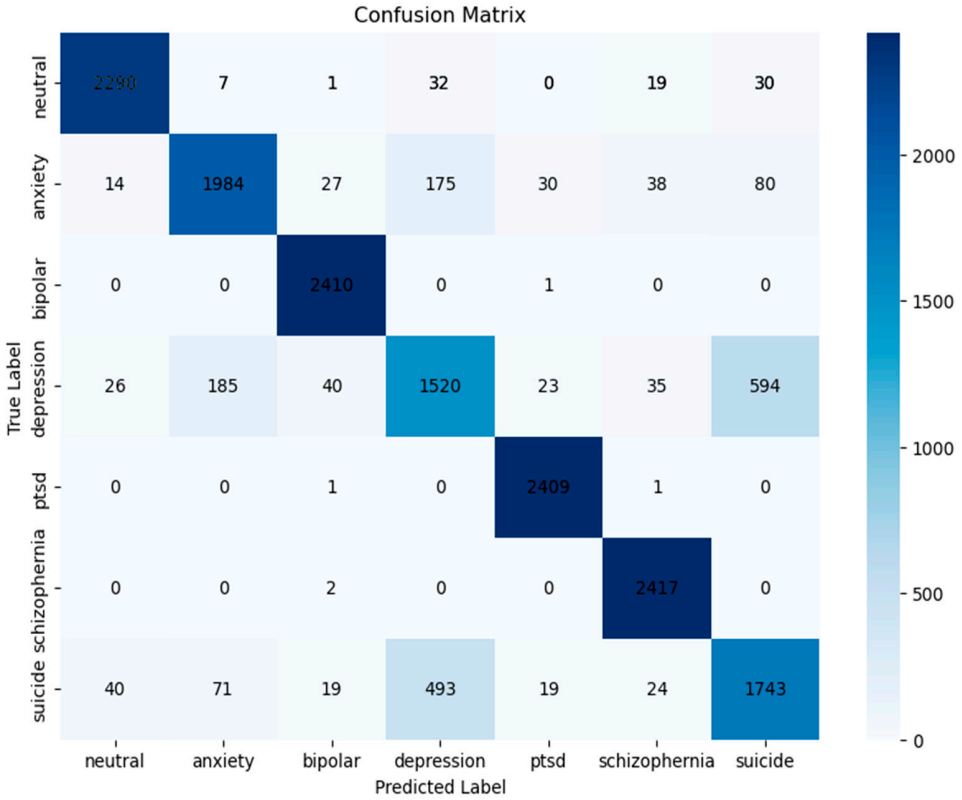

Figure 7. Confusion matrix of LSTM model with hybrid data balancing.

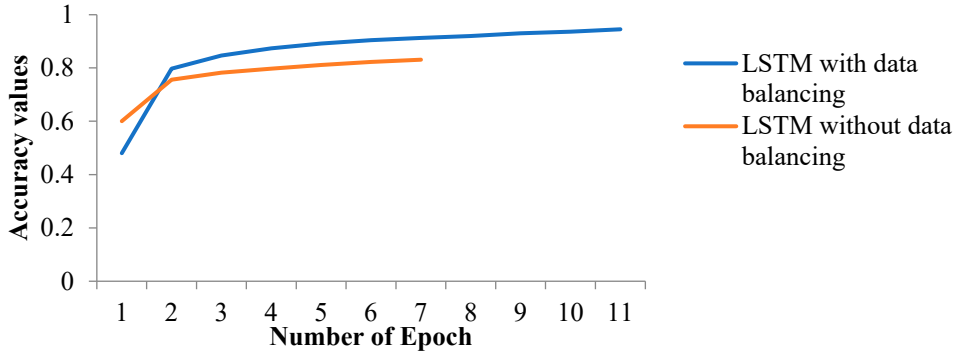

Figure 8. Epoch wise accuracy values during training of LSTM model.

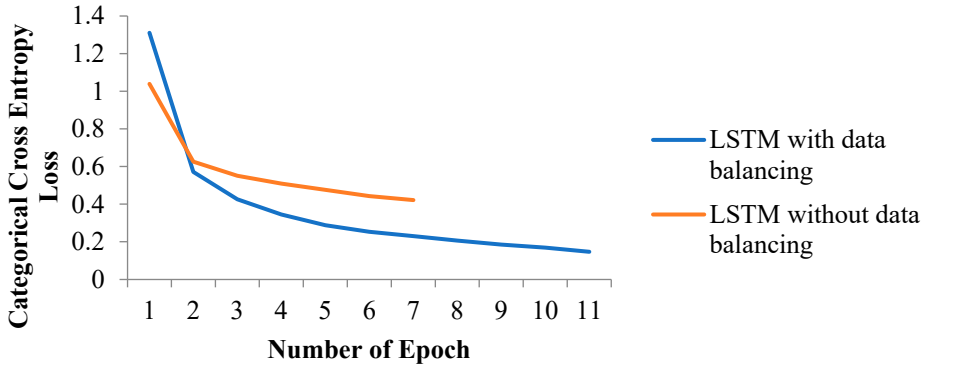

Figure 9. Epoch wise categorical cross entropy loss values during training of LSTM model.

Table 9 illustrates a class-wise comparison of the LSTM model's performance with and without data balancing, focusing on precision, recall, and F1-score across different mental health categories. A class-wise analysis revealed that data balancing generally improved precision (the ability to identify true positives) and the F1-score (balanced metric

of precision and recall) for minority classes. However, the recall for the majority class ("depression") dropped slightly. Each dataset possesses distinct characteristics, such as variable patterns, imbalance ratios, and the number of minority classes. Consequently, the effectiveness of data balancing techniques may vary depending on the uniqueness of each dataset. In our study, the dataset is highly imbalanced, with one class containing only 910 samples compared to another which has 33,549 samples. Therefore, applying the data balancing technique helps to mitigate the issue of class imbalance and improves the model's ability to detect and classify less prevalent mental health conditions accurately. While it may affect some metrics for the majority class, the overall benefits for minority classes are substantial.

Table 9. Class-wise prediction outcomes for LSTM model with and without data balancing.

Class	Without Data Balancing			With Data Balancing		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
Neutral	0.95	0.94	0.95	0.96	0.98	0.97
Anxiety	0.86	0.80	0.83	0.87	0.86	0.86
Bipolar Disorder	0.73	0.50	0.59	0.98	1.00	0.99
Depression	0.74	0.85	0.79	0.70	0.60	0.65
PTSD	0.81	0.67	0.73	0.96	1.00	0.98
Schizophrenia	0.80	0.60	0.69	0.96	1.00	0.98
Suicide	0.64	0.49	0.56	0.71	0.74	0.72

4. Discussion

Given the potential for social media data to offer insightful information about personal well-being, this study has proposed a deep learning-based prediction model for identifying mental illness as a way to address the serious drawbacks of conventional mental health assessment techniques, which frequently mainly rely on questionnaires. The uniqueness of this work lies in its three-pronged approach to analyzing social media data for mental health insights:

- **Multiclass mental illness prediction.** Unlike many studies that focus on detecting the presence or absence of a single mental health condition, this project aims to classify users into a range of potential mental illnesses. This multiclass approach provides a more nuanced and comprehensive understanding of mental health in the digital sphere.
- **LSTM-based classification.** The project leverages the power of LSTM networks, a type of deep learning algorithm particularly adept at understanding sequential data like text. This allows the model to capture complex relationships between words and phrases within social media posts, potentially revealing subtle linguistic patterns indicative of specific mental health conditions. The dataset is created as a $74{,}824 \times 250 \times 200$ matrix and is fed into the LSTM model.
- **Comparative analysis with data balancing.** By recognizing the challenges posed by imbalanced datasets in mental health research, this study investigates the impact of data balancing on model performance by taking different target values, namely 910, 8000, and 12,000. This rigorous evaluation ensures the model's reliability and generalizability across various mental health categories.

Ablation Analysis

Since it is unknown which design parameters will produce the best outcomes, we examined the model independently without using FS and data balancing components.

The performance of traditional ML models was first evaluated for tackling mental illness detection as a multiclass classification problem without FS and without data balancing. The effect of FS on the top-performing ML model and LSTM was then evaluated through a simulation using varying feature sizes. Lastly, the impact of data balancing on the LSTM model with the smaller feature set was examined, and its effectiveness was contrasted with the top-performing ML model utilizing FS and the same balanced dataset.

- Table 5 shows that with an accuracy of 74%, LR utilizing the TF-IDF vectorizer outperforms MNB and RF. Analyzing the outcomes of the three ML models with and without FS clearly reveals that LR performs better than RF and MNB with the TF-IDF vectorizer with respect to all criteria.
- When analyzing the effect of FS on the LR and LSTM models, as indicated in Table 7, it is observed that the LSTM model achieves an accuracy of 77% with 200 features selected using the embedded matrix, whereas the LR model achieves an accuracy of 75% with 15,000 features selected by Chi-square. By employing Chi-square, the accuracy of LR is increased by 13.5%. However, using Chi-square with LSTM does not yield satisfactory results. Rather, LSTM outperforms LR when utilizing the pre-trained embedded matrix with just 200 features.
- Furthermore, when analyzing the performance of the LSTM model with different data balancing techniques, as shown in Table 8, it is clear that the hybrid data balancing technique using random sampling in oversampling and undersampling showed impressive results, and the accuracy of LSTM increases to 88% from 77%, while the Hamming loss is reduced to 0.04.
- Using the reduced feature set and the same balanced dataset used for LSTM, the impact of data balancing on LR is further examined. As shown in Figure 5, it is found that data balancing improves LR's accuracy from 0.75 to 0.85 and lowers the Hamming loss from 0.24 to 0.15.
- When the performance of LSTM and LR is compared with the same balanced dataset using a reduced feature set, it is evident from Figure 5 that LSTM performs better than LR, improving accuracy by 3.5% and Hamming loss by 73%. Overall, data balancing improves both models, with LSTM showing a greater relative gain.
- From the visual comparison of the confusion matrix depicted in Figures 6 and 7, it is observed that the model exhibits a more uniform distribution of prediction across classes when balancing is used. This implies that data balancing improves overall classification across all categories by reducing the model's propensity to favor the majority class.

5. Conclusions

In light of the fact that social media data might offer valuable insights into individual well-being, this study investigated the use of various machine learning algorithms to forecast mental illness. Using a multiclass text dataset, the efficacy of the ML models was assessed through experimentation in various circumstances, taking into account the impact of vectorization, feature selection, and data balancing in relation to model performance. Among the three traditional machine learning models, LR was found to perform better than the others with Chi-square-based feature selection and TF-IDF vectorizer. When the LR model was compared with the deep-learning LSTM model, the LSTM model developed with the pre-trained word embedding layer produced superior results with fewer features. Moreover, the hybrid data balancing strategy further improved the accuracy of the LSTM model from 77% to 88% and decreased the Hamming loss to 0.04. Overall, utilizing deep learning with social media data for mental health analysis has the potential to lead to better outcomes and a better understanding of mental health concerns.

However, this study is subject to a few limitations. All evaluations were conducted using just a single mental health dataset covering six distinct types of mental illness. Future investigations might focus on increasing the model's robustness and generalizability by employing a broader range of social media datasets and additional mental health categories. Again, finding the optimal set of hyperparameter values is a challenging task for LSTM. In subsequent research, a suitable optimization algorithm will be integrated with the LSTM model to select optimal hyperparameter values. Alternative LSTM variants and ensemble DNN models will also be developed to resolve the multiclass imbalance issue in detecting mental illness. Further advancements are necessary to ensure improved and stable performance, particularly with complex real-world data.

Author Contributions: Conceptualization, S.U., R.K.M., V.D. and A.D.; methodology, S.U. and R.K.M.; software, S.U. and R.K.M.; validation, S.U. and R.K.M.; formal analysis, S.U. and R.K.M.; investigation, S.U. and R.K.M.; resources, S.U. and R.K.M.; data curation, V.D. and A.D.; writing—original draft preparation, R.D., V.D. and A.D.; writing—review and editing, R.D., V.D. and A.D.; visualization, S.U., R.K.M. and V.D.; supervision, R.D.; project administration, R.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset used in this study is publicly available at https://zenodo.org/records/3941387#.Y5L6O_fMKUl (accessed on 27 December 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Joshi, D.; Patwardhan, M. An analysis of mental health of social media users using unsupervised approach. *Comput. Hum. Behav. Rep.* **2020**, *2*, 100036. [CrossRef]
2. Chancellor, S.; De Choudhury, M. Methods in predictive techniques for mental health status on social media: A critical review. *NPJ Digit. Med.* **2020**, *3*, 43. [CrossRef] [PubMed]
3. Zhang, T.; Schoene, A.M.; Ji, S.; Ananiadou, S. Natural language processing applied to mental illness detection: A narrative review. *NPJ Digit. Med.* **2022**, *5*, 46. [CrossRef]
4. Chung, J.; Teo, J. Mental health prediction using machine learning: Taxonomy, applications, and challenges. *Appl. Comput. Intell. Soft Comput.* **2022**, *2022*, 9970363. [CrossRef]
5. Garg, M. Mental health analysis in social media posts: A survey. *Arch. Comput. Methods Eng.* **2023**, *30*, 1819–1842. [CrossRef]
6. Safa, R.; Edalatpanah, S.A.; Sorourkhah, A. Predicting mental health using social media: A roadmap for future development. In *Deep Learning in Personalized Healthcare and Decision Support*; Academic Press: Cambridge, MA, USA, 2023; pp. 285–303.
7. Aldarwish, M.M.; Ahmad, H.F. Predicting depression levels using social media posts. In Proceedings of the 2017 IEEE 13th International Symposium on Autonomous decentralized system (ISADS), Bangkok, Thailand, 22–24 March 2017; pp. 277–280.
8. Hao, B.; Li, L.; Li, A.; Zhu, T. Predicting mental health status on social media: A preliminary study on microblog. In *Cross-Cultural Design. Cultural Differences in Everyday Life: 5th International Conference, CCD 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, 21–26 July 2013*; Proceedings; Part II 5; Springer: Berlin/Heidelberg, Germany, 2013; pp. 101–110.
9. Thorstad, R.; Wolff, P. Predicting future mental illness from social media: A big-data approach. *Behav. Res. Methods* **2019**, *51*, 1586–1600. [CrossRef]
10. Hussain, S.; Nasir, A.; Aslam, K.; Tariq, S.; Ullah, M.F. Predicting mental illness using social media posts and comments. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 607–613.
11. Ansari, G.; Garg, M.; Saxena, C. Data augmentation for mental health classification on social media. *arXiv* **2021**, arXiv:2112.10064.
12. Vaishnavi, K.; Kamath, U.N.; Rao, B.A.; Reddy, N.S. Predicting mental health illness using machine learning algorithms. *J. Phys. Conf. Ser.* **2022**, *2161*, 012021.
13. Kim, J.; Lee, J.; Park, E.; Han, J. A deep learning model for detecting mental illness from user content on social media. *Sci. Rep.* **2020**, *10*, 11846. [CrossRef]
14. Ang, C.S.; Venkatachala, R. Generalizability of Machine Learning to Categorize Various Mental Illness Using Social Media Activity Patterns. *Societies* **2023**, *13*, 117. [CrossRef]
15. Uddin, M.Z.; Dysthe, K.K.; Følstad, A.; Brandtzaeg, P.B. Deep learning for prediction of depressive symptoms in a large textual dataset. *Neural Comput. Appl.* **2022**, *34*, 721–744. [CrossRef]

16. Kour, H.; Gupta, M.K. An hybrid deep learning approach for depression prediction from user tweets using feature-rich CNN and bi-directional LSTM. *Multimed. Tools Appl.* **2022**, *81*, 23649–23685. [CrossRef]
17. Murarka, A.; Radhakrishnan, B.; Ravichandran, S. Detection and classification of mental illnesses on social media using roberta. *arXiv* **2020**, arXiv:2011.11226.
18. Narayanan, S.R.; Babu, S.; Thandayantavida, A. Detection of depression from social media using deep learning approach. *J. Posit. Sch. Psychol.* **2022**, *6*, 4909–4915.
19. Bokolo, B.G.; Liu, Q. Deep learning-based depression detection from social media: Comparative evaluation of ml and transformer techniques. *Electronics* **2023**, *12*, 4396. [CrossRef]
20. Alkahtani, H.; Aldhyani, T.H.; Alqarni, A.A. Artificial Intelligence Models to Predict Disability for Mental Health Disorders. *J. Disabil. Res.* **2024**, *3*, 20240022. [CrossRef]
21. Bhavani, B.H.; Naveen, N.C. An Approach to Determine and Categorize Mental Health Condition using Machine Learning and Deep Learning Models. *Eng. Technol. Appl. Sci. Res.* **2024**, *14*, 13780–13786. [CrossRef]
22. Revathy, J.S.; Maheswari, N.U.; Sasikala, S.; Venkatesh, R. Automatic diagnosis of mental illness using optimized dynamically stabilized recurrent neural network. *Biomed. Signal Process. Control* **2024**, *95*, 106321.
23. Ezerceli, Ö.; Dehkharghani, R. Mental disorder and suicidal ideation detection from social media using deep neural networks. *J. Comput. Soc. Sci.* **2024**, *7*, 2277–2307. [CrossRef]
24. Zhu, L.; Hou, S.; Cheng, T. Mental Health Status Detection Model Based on LSTM Neural Network. *Procedia Comput. Sci.* **2024**, *243*, 842–849. [CrossRef]
25. Bendebane, L.; Laboudi, Z.; Saighi, A.; Al-Tarawneh, H.; Ouannas, A.; Grassi, G. A Multi-class deep learning approach for early detection of depressive and anxiety disorders using Twitter data. *Algorithms* **2023**, *16*, 543. [CrossRef]
26. Karamat, A.; Imran, M.; Yaseen, M.U.; Bukhsh, R.; Aslam, S.; Ashraf, N. A Hybrid Transformer Architecture for Multiclass Mental Illness Prediction using Social Media Text. *IEEE Access* **2024**, *13*, 12148–12167. [CrossRef]
27. Yang, K.; Zhang, T.; Kuang, Z.; Xie, Q.; Huang, J.; Ananiadou, S. MentaLLaMA: Interpretable mental health analysis on social media with large language models. In Proceedings of the ACM Web Conference 2024, Singapore, 13–17 May 2024; pp. 4489–4500.
28. Low, D.M.; Rumker, L.; Torous, J.; Cecchi, G.; Ghosh, S.S.; Talkar, T. Natural Language Processing Reveals Vulnerable Mental Health Support Groups and Heightened Health Anxiety on Reddit During COVID-19: Observational Study. *J. Med. Internet Res.* **2020**, *22*, e22635. Available online: https://zenodo.org/records/3941387#.Y5L6O_fMKUl (accessed on 27 December 2024.). [CrossRef]
29. He, H.; Ma, Y. (Eds.) *Imbalanced Learning: Foundations, Algorithms, and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2013.
30. Padurariu, C.; Breaban, M.E. Dealing with data imbalance in text classification. *Procedia Comput. Sci.* **2019**, *159*, 736–745. [CrossRef]
31. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
32. Wang, Q. A hybrid sampling SVM approach to imbalanced data classification. In *Abstract and Applied Analysis*; Hindawi Publishing Corporation: New York, NY, USA, 2014; Volume 2014, p. 972786.
33. Koziarski, M. CSMOUTE: Combined synthetic oversampling and undersampling technique for imbalanced data classification. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021; pp. 1–8.
34. Ependi, U.; Rochim, A.F.; Wibowo, A. A hybrid sampling approach for improving the classification of imbalanced data using ROS and NCL methods. *Int. J. Intell. Eng. Syst.* **2023**, *16*, 345–361.
35. Wongvorachan, T.; He, S.; Bulut, O. A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining. *Information* **2023**, *14*, 54. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.