International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(3s) 2026 pp. 18-39
https://doi.org/10.70917/ijcisim-2026-2277
Article

# Mental Health Symptoms Detection System from Social Media Posts using FastText Embedding and an Ensemble Deep Learning Model

**Reeta Bourasi1*, Sakshi Rai2**

[1,2]Computer Science & Technology, LNCT University, Bhopal, MP, India
[1]just.reeta@gmail.com,[2]sakshirai.sr29@gmail.com
**Corresponding Author :** just.reeta@gmail.com

**Abstract:** Social media is becoming a significant tool for early emotional and psychological identification because of mental health disorders, like depression and anxiety, are increasingly prominent in online social media. This paper presents an ensemble deep learning approach to detect potential indicators of psychotic behavior—such as depression, anxiety, and related mental health conditions—based on user-generated textual data. The suggested method combines FastText word embeddings with a hybrid CNN+LSTM ensemble model to capture local semantic patterns and long-term contextual dependencies. To enhance dataset diversity and linguistic representativeness, data were gathered from Twitter and Reddit which encompass a wide range of user expressions and linguistic variances. A number of experiments were performed with optimized hyperparameters and compared them to conventional deep learning models and a fine-tuned BERT transformer. The suggested model got an overall accuracy of 91.89%, which is quite close to BERT's 93.10% but it maintained significantly lower computational cost and training time. The model was trained and evaluated using MATLAB tool. In order to check the performance of proposed model, evaluation matrices like accuracy, recall, F1-score were used that shows that suggested system is both accurate and morally sound. The findings highlight the potential of lightweight yet robust architectures for large-scale, real-time mental health monitoring. it offers a promising tool for early identification of mental health concerns and contributes to the development of intelligent, non-invasive monitoring systems for online psychological well-being assessment.
**Keywords:** BERT, CNN, FastText, LSTM, RNN, Tweeter

## 1. INTRODUCTION

Social media is a widely used platform for individuals to freely express their emotions [1,2]. Users frequently express their mental health issues or illnesses anonymously using various social media platforms or online communities related to social health. Online health communities serve as a platform for individuals to express sympathy and connect with others who share similar symptoms [3]. Furthermore, users frequently seek health information pertaining to their symptoms on social media platforms in an effort to self-diagnose [4,5]. Identifying depressive symptoms at an early stage and promptly providing evaluation and therapy can significantly enhance the likelihood of managing symptoms and the root cause of the disease. This approach can also help reduce its negative impact on overall well-being, health, and other aspects of personal, economic, and social life [6-8]. Nevertheless, identifying depression symptoms can be difficult and requires significant resources. Present methodologies mostly rely on clinical interviews and questionnaire surveys conducted by hospitals or agencies [9], wherein psychological assessment tools are employed to anticipate mental disorders. This methodology mostly relies on individual questionnaires and can provide a rough diagnosis of depression.

An alternate method for predicting depression is to analyse informal texts submitted by users, instead of conducting interviews or questionnaires. This method for detecting signs of depression in informal texts shows promise since it makes use of cutting-edge developments such as AI and natural language processing. Artificial

Intelligence (AI) used for natural language processing utilizes linguistics and computational approaches to assist machines in deciphering textual content, including sentiments and emotions. Therefore, the primary objective is to evaluate opinions, ideas, and concepts by assigning either negative or positive polarities [10].

Prior research has demonstrated that text analysis algorithms may effectively identify depression symptoms. This technique has been successfully utilized in several applications, such as extracting sentiments from suicide notes and identifying offensive or despondent language in chats or blog posts [11-14]. Many scholars have examined user-generated content on social media to observe individuals' emotional state or mental health, such as depression, anxiety, or schizophrenia [3,15]. A recent study conducted a data collection of Twitter posts from users who claimed to have been diagnosed with depression [16,17]. The study then utilized a deep learning model to analyze the linguistic and emotional traits of the collected tweets, and subsequently monitored the alterations in their social engagement on Twitter. We gathered data from various distinct sources that focus on mental health concerns, encompassing conditions such as bipolar disorder, psychiatric diseases (such as schizophrenia and bipolar disorder), and autism. The objective was to ascertain whether an individual is experiencing symptoms of depression or anxiety [15].

### *1.1 Contributions*

This paper addresses key limitation in existing mental health detection systems based on social media data post. Most of the approaches mostly rely on either only deep learning models or transformer-based architectures. Single-model methods sometimes fail to captures both localized linguistic patterns and large contextual dependencies, which makes detection less accurate. On the other hand, transformer-based models such as BERT work effectively to detects mental health, but it requires a lot of computing power, which makes them hard to scale and dificults to use in real time in places where resources are restricted.

This study shows a hybrid ensemble model that uses FastText embeddings with CNN and LSTM architectures to overcome these limitations. FastText embeddings provide a better representation of social media text that is informal and has many forms. CNN layers show local semantic properties, and LSTM layers show sequential contextual linkages. This combination improves the representation of features and the effectiveness of classification while keeping the computational cost lower than that of transformer-based methods.

Here are the main contributions of this work:

1.We have adopted a hybrid ensemble architecture that combine FastText, CNN, and LSTM in such a way that it takes advantage of the characteristics that each of these structures have in terms of embedding, spatial feature extraction, and sequential learning.

2. The model is tested on a diverse dataset from a number of social media sites, such as Twitter and Reddit that makes it robust to different communication styles.

3. The proposed method is a cheaper way to use computer resources than transformer models.

4. The proposed framework provides a cost-effective and scalable way to set up real-time mental health monitoring systems which makes it useful for both public health and clinical settings.

The contributions differentiate the proposed technique from other methodologies that are already in use and demonstrate its efficacy, efficiency, and practical usefulness in the process of recognizing mental health symptoms using data collected from social media platforms.

Here is an examination of the structure of the paper's organization: The second section of the paper explores the literature review pertaining to the present topic. The execution of our suggested methodology is elaborated in Section 3, where we delineate the materials and techniques employed. Section 4 provides a detailed discussion of the experiments and performance measurement. Section 5 is specifically focused on analysing and presenting the final findings and outcomes. Section 6 and 7 deals with conclusion and future works, respectively.

## 2. RELATED WORKS

Current research has prioritized the examination of language found in social media posts [6–9]. Due to the emergence of social media, individuals with mental disorders have discovered online forums focused on mental health, where they can openly discuss their mental health issues while maintaining anonymity. Safa et al. [18] introduced a novel approach that enables the automated gathering of a substantial quantity of tweets from users, followed by their analysis to determine the presence of depressed characteristics. The individuals in the gathered tweets self-identified as being depressed. They suggested a multi-modal framework that utilized the n-gram language model and Linguistic Inquiry and Word Count (LIWC) to predict depression.

Rissola et al. [19] introduced a novel approach for gathering a dataset of social media posts that either include or do not include references to depression. The author highlighted the challenge of constructing a model with high accuracy for detecting depression due to the absence of a comprehensive dataset. Therefore, the dataset provided by the author has the ability to reliably forecast depression. The researchers utilized the BERT model to train their dataset, yielding highly favourable outcomes characterized by excellent accuracy, precision, recall, and F1 scores.

Jina Kim et al. [20] proposed a deep learning model capable of identifying mental health disorders including bipolar disorder, schizophrenia, autism, bipolar personality disorder, and autism. The author utilized Reddit as a means of gathering data for their study endeavor. The dataset consists of six classes. The researchers encountered a problem of class imbalance and addressed it by employing the SMOTE method. The XG Boost classifier and CNN were employed to categorize Reddit content.

slam et al. [21] employed machine learning methodologies to identify depression on the Facebook platform. Facebook users often convey their emotions through emojis or comments, making it necessary to extract relevant features. Authors utilized LIWC to extract features from Facebook, including both Facebook posts and comments made by individuals. The author employed machine learning methodologies to categorize characteristics derived from legal analysis. The methods comprised decision trees, support vector machines (SVM), k-nearest neighbours (KNN), and ensemble classifiers.

In their study, Manoj et al. [22] examined the identification of depression through the analysis of tweets using natural language processing (NLP). Initially, tweets were extracted and subjected to essential preprocessing procedures. Subsequently, a hybrid text embedding methodology was employed to transform textual data into numerical representations, incorporating both fast text and TF-IDF (Term Frequency Inverse Document Frequency). Subsequently, machine learning classifiers were utilized to analyze the dataset and ascertain the presence of depression. The SVM and Random Forest classifiers were employed, with the Random Forest classifier attaining the greatest accuracy rate of 75%.

Aswathy and colleagues [23] introduced a mobile application designed to aid in the identification of depression. The primary concept is that the user will input various data into the system, such as expressing feelings of depression, and so on. If the statement exhibits characteristics of depression, the application will inform you that you are experiencing depression, hence refuting the possibility that you do not have depression. The core essence lies within the system, encompassing its functionality and construction. The author utilized an imbalanced dataset of tweets. A total of 11,911 tweets exhibited normal sentiments, whereas 2,308 tweets displayed signs of depression. A combined model consisting of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) was employed to train a dataset of tweets submitted by the user on Twitter. The performance of the ensemble model much surpassed that of the simple SVM model. The accuracy of the ensemble model was 0.97, far surpassing the accuracy of the SVM which was 0.83. In their study, Ghosh et al. [24] introduced a bidirectional LSTM CNN model that incorporates attention mechanisms for the purpose of detecting depression in the Bangla language on a social media platform. Prior research employed lexicon-based tagging for the purpose of feature extraction. This research utilizes the attention mechanism to extract features by directing its focus towards pertinent and crucial aspects of the text. The inclusion of attention mechanisms led to a significant enhancement in the model's performance. The precision reached a level of 96 percent.

Recent advancements have explored multimodal and transformer-based approaches for end-to-end emotion and behavioural analysis. In 2024 Zhang et al. [25] proposed RMER-DT, a diffusion-transformer model that achieves impressive robustness in multimodal emotion detection through simultaneous modelling of text, audio, and video features. Zhou et al. [26], in 2023 proposed RAFT (Robust Adversarial Fusion Transformer) for multimodal sentiment analysis Their experiment proved that adversarial fusion enhances resistance to noisy inputs and modality imbalance. In 2024 Wu et al. [27] designed a contextual interaction-based multimodal emotion model that embedded improved semantic information by using transformer attention. Li et al. [28] investigated contrastive removal of negative information for enhancing cross-domain emotion understanding.

Aside from the emotion analysis, some studies [29-35] have explored robust AI mechanisms in other diverse intelligent systems such as reliable image super-resolution [29], knowledge distillation for road segmentation [30], Generalizable deepfake detection [31-32], and federated domain generalization for ophthalmology [33]. These works give emphasize more on increasing attention to model robustness, fairness, and explainability which are crucial factors pertinent to AI deployment in mental health. Further, non-contact emotion recognition systems (e.g., client-server behavioural evaluation systems, Singh et al., [34]) and graph-based learning methods (e.g., anomaly detection in spectral graphs, Huang et al., [35]) also show the field's shift toward cross-modal, real-time inference.

In contrast to current multimodal or visual frameworks, our proposed FastText + CNN + LSTM ensemble exclusively utilizes text-based social media data. Its objective is to be comprehensible and efficient in computation. Our approach employs semantic and morphological indicators from text to identify psychological signs in user posts, in contrast to transformer-based fusion models that require extensive multimodal inputs. The proposed paradigm offers an alternative perspective by equilibrating performance and cost-effectiveness for extensive, language-oriented mental health evaluations.

## 3. PROPOSED METHODOLOGY

This paper investigates the identification of psychotic and depression-related behavioral patterns in social media posts through an ensemble deep learning framework which incorporates FastText embeddings. The proposed system has three main parts: the FastText embedding layer, the Convolutional Neural Network (CNN), and the Long Short-Term Memory (LSTM) Recurrent Neural Network (RNN). The system is developed to automatically identify social media posts that contain self-perceived symptoms of depression or any psychotic behavior. The proposed ensemble method improves the classification of mental health-related content from text data through the complementary capabilities of CNN and LSTM.

The approach is applied to a large dataset collected from the publicly available online information dissemination channel, that is, Twitter and the reddit website. The dataset includes text messages posted by young people.

Before the training of the model, the messages are preprocessed and represented using the FastText embedding approach. The advantages of the proposed approach lie in its superiority over the traditional approaches, which are mainly based on word frequency. Unlike the traditional word frequency-based approaches, the FastText approach considers subword details and morphological differences, which are more effective in dealing with informal language, abbreviations, slang, and spelling differences, as observed in the social media messages.

The CNN module is used for the extraction of local semantic features and n-gram patterns from the embedded text. The convolutional filters of different sizes are used in the CNN to effectively identify the key phrases related to psychological symptoms. The feature dimensionality is reduced in the process, and the local dependencies are maintained. The LSTM network is used for modeling the long-range sequential dependencies in the text. The context is maintained in the LSTM network, which helps the system to identify the emotional relationships that can be related to multiple words or phrases. The combination of CNN and LSTM helps in leveraging the strengths of both.

The choice of FastText, CNN, and LSTM is driven by the complementary capabilities and efficiency of the models. Although transformer models like BERT are capable and have a good understanding of context, they consume a lot of computational power and time for training. The proposed model, on the other hand, is able to provide classification performance comparable to the state-of-the-art models but with much lower computational complexity.

The proposed framework is designed to differentiate between the posts that show symptoms of depression or any psychotic behavior and the posts that are not related to depression. The trained model will be used to classify the social media posts automatically. The block diagram of the proposed methodology is shown in Figure 1.

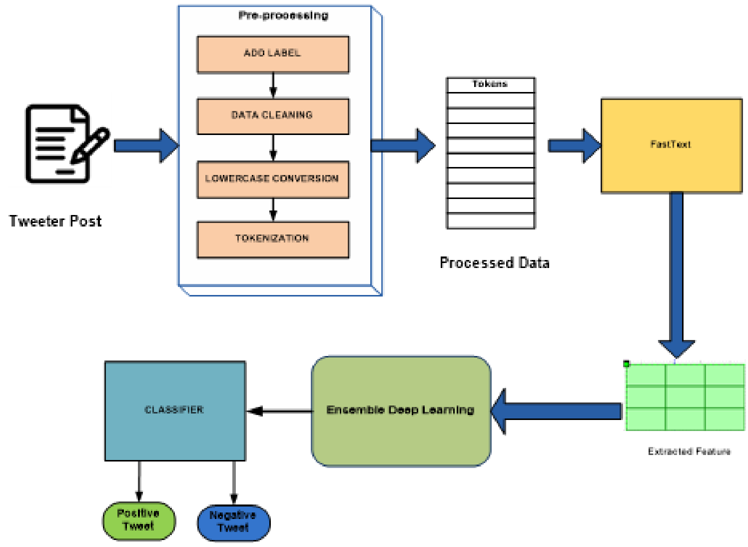



**Fig. 1 Block Diagram of Proposed Work**

This study utilizes a combination of two CNN and one recurrent neural network (LSTM) to carry out the classification task. During the initial phase, the dataset undergoes preprocessing before being fed into several deep learning models, including recurrent neural networks and convolutional neural networks. Figure 2 depicts the flow chart of Ensemble Deep Learning Models.

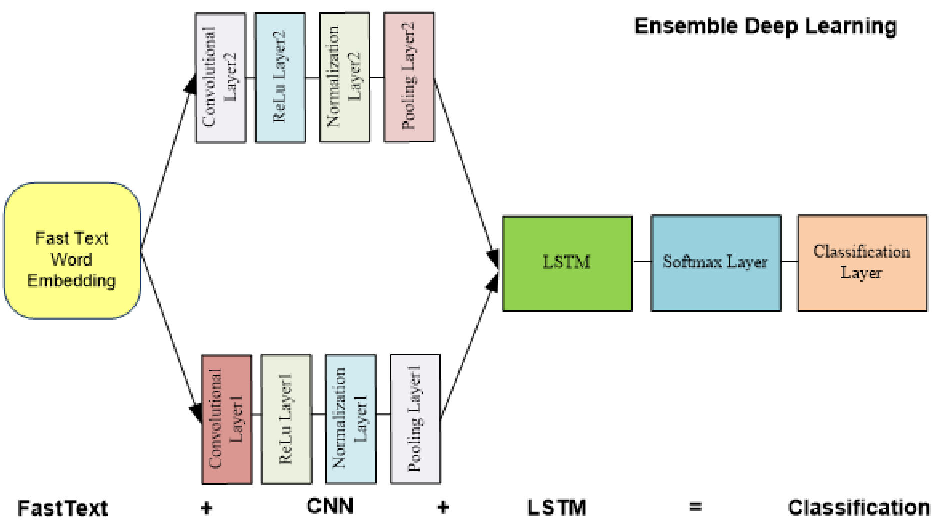



**Fig. 2 Flow Chart of the Ensemble Deep Learning Model**

### *3.1 Dataset*

We have created a comprehensive multi-platform dataset by combining data from Twitter and Reddit. Initially, data were gathered from a portion of the Sentiment140 dataset [36] with other tweets [37-39] related to depression and normal emotions. To enhance linguistic diversity and contextual coverage, we have further included Reddit Mental Health dataset extracted from Kaggle [40] in which posts are related to # Stress, #Depression, #Personality disorder, #Anxiety etc. These Reddit posts were filtered for English language, tokenized, and cleaned following the same preprocessing pipeline used for Twitter data.

The final dataset consists of a total of approximately 47,240 posts, with an almost balanced distribution between depressive and non-depressive samples. Data from Reddit (approx. 6000 samples), integrated in our dataset, enabled the model to learn varied language structures and contextual nuances that are not always present in short Twitter texts.

This enhancement significantly improves the representativeness and generalizability of the proposed model. The dataset can be located in our repository. Table 1 shows the summary of our dataset.

Table I: Dataset Summary

| Platform | Depression | Non-Depression | Total |
|---|---|---|---|
| Twitter | 16238 | 25044 | 41282 |
| Reddit | 05958 | 00000 | 05958 |
| Total | 22196 | 25044 | 47240 |

The data set is comprised of English-language social media postings gathered from Twitter and Reddit. The postings were manually labeled as either depression-related or non-depression-related. The addition of the Reddit data set adds linguistic diversity to the data set.

### 3.1.1 **Collecting data**

The following hashtags are apparently associated with tweets about psychotic behaviour. We gathered this data from four channels in order to analyse mental health from these posts.

- #depression, #Anxiety, #bipolar, #loneliness, #Suside, #hopelessness, #mentalhealth, #mentalillness etc.

In addition to positive data, We added some negative data with the following hashtags:

- #mentalhealth, #health, #happiness, #happy, #joy, #wellbeing etc.

### 3.1.2 **Data pre-processing procedure:**

Figure 3 shows the data pre-processing approach for the post-data collection process. once the data has been collected, we remove duplicate items based on tweet id and redundant columns from post data. Finally, each post is associated with its relevant label. By default, we set a target column to 1, and for entries that do not have depression, we manually set it to 0. The data and labels from the random sample are displayed in figure 4. After acquiring the relevant dataset, our research proceeds to clean it up by:

- Eliminating extra spaces, punctuation marks, and stop words from every post.
- Eliminating any post that has a large number of hashtags, @mentions, and URLs
- Eliminating entries containing fewer than twenty-five characters, or five words.

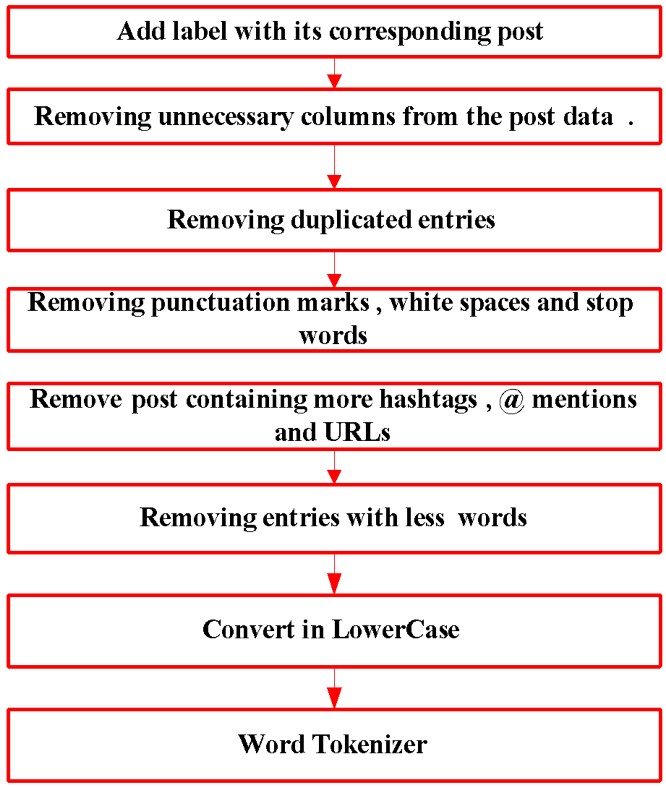



**Fig. 3 A Data Pre-Processing Procedure**

| tweets | Label |
|---|---|
| "RT @richardbranson: Read this wonderful blog about elephant conservation in Zimbabwe: http://t.co/MByWtcAElX #WorldElephantDay http://t.co/…" | 1 |
| "College minor in substance abuse? #addiction https://t.co/5sAMvwU7Ct" | 1 |
| "What would your life be like with less #anxiety?" | 1 |
| "#Therapy is an effective treatment of #depression and #anxiety" | 1 |
| "I fell in love with a #narcissist: http://t.co/sRLtRufedT" | 1 |
| "I love you lady!:) @Chloe_Roknich" | 1 |
| "I'm doing INSANITY MAX:30™ - Tabata Strength in Team Beachbody's SuperGym. Join Me! Go to http://t.co/gPaLPBDw3v" | 1 |
| "I'm doing INSANITY MAX:30™ - Cardio Challenge in Team Beachbody's SuperGym. Join Me! Go to http://t.co/gPaLPBDw3v" | 1 |

**Fig. 4 Sample Dataset File**

Once the dataset is cleaned, we changed the entire post to lowercase letters. Next, we filtered stop words—words that users frequently used—by tokenizing their postings using the matlab built-in function *tokenizedDocument*(). Tokenized documents are documents that are represented as a collection of words (also referred to as tokens) and are employed for use in text analysis. Next, a matlab function called *wordencoding*() is employed to convert words in a vocabulary into numerical indices once the tokens have been obtained. Sample tokenized preprocessed documents are depicted in Fig. 5.

Afterwards, the analysis made use of data from five sources totaling 47,240 posts. The outcomes are deposited into csv files and assigned for evaluation.

```
7 tokens: read wonderful blog elephant conservation zimbabwe http://t.co/
5 tokens: college minor substance abuse #addiction
4 tokens: life like less #anxiety
5 tokens: #therapy effective treatment #depression #anxiety
3 tokens: fell love #narcissist
3 tokens: love lady @chloe_roknich
8 tokens: insanity max tabata strength team beachbodys supergym join
8 tokens: insanity max cardio challenge team beachbodys supergym join
```

**Fig. 5 View the First Few Pre-Processed Documents**

#### 3.1.3 **Finalizing dataset**

We personally evaluated csv files created by the previous script, which comprised filtered tweets with depression and other hashtags. We eliminated non-English tweets from these csv files. The resultant csv files represent a about equal distribution of depressive and non-depressive tweets, with a 50-50 split. Since, all of the tweets originally had depressing titles, so this is a good set of data to train our model on such dataset. By separating the tweets according to their content instead of their tags, we are training the model to pay more attention to the content and make more accurate predictions.

Despite the significant time and effort invested in producing the dataset, we strongly feel it was crucial to ensure its accuracy. This dataset serves as the foundation for developing a precise depression detector, distinguishing it from a mere sentiment detector.

Fig.6 the distribution of the classes in the data using a histogram. Word clouds in Fig.7 show the most common words in each topic for the depressive or non-depressive post.

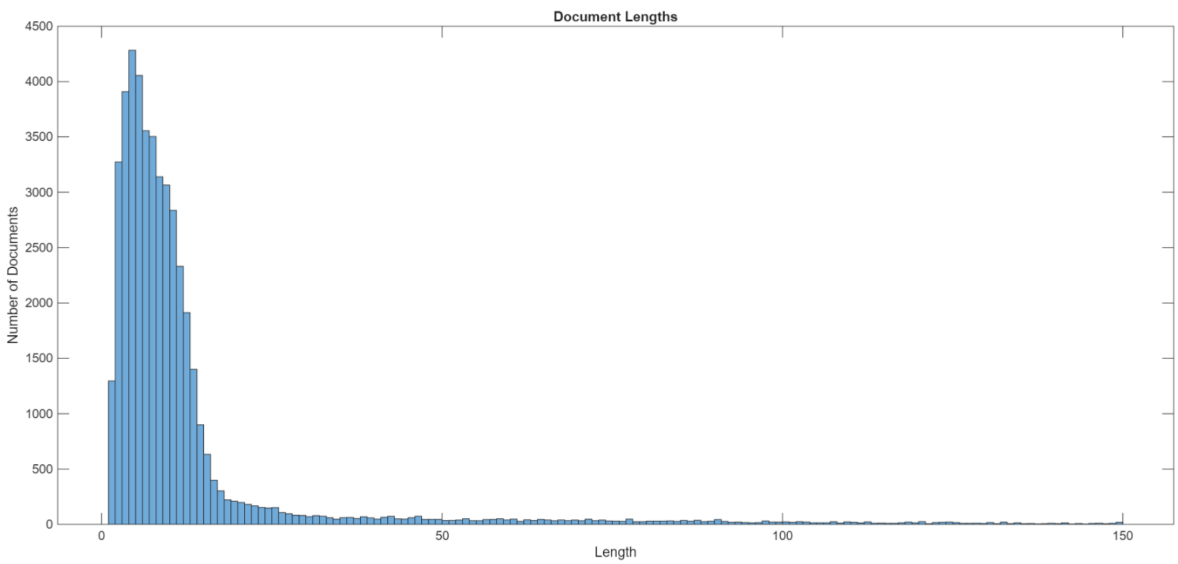



**Fig. 6 shows the Histogram of Distribution of the Classes in the Data**

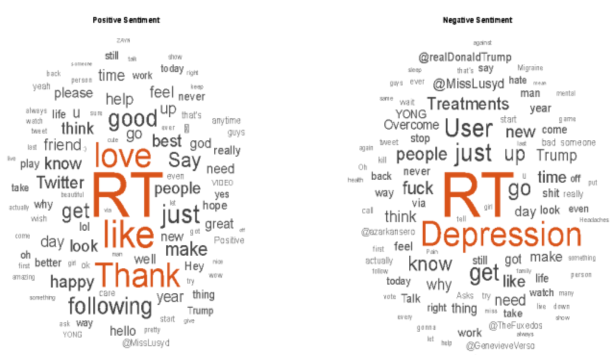



**Fig. 7 Word Cloud Visualization Based on the Distribution of Words for Positive or Negative Sentiments**

### *3.2 Feature Generation Using FastText Method*

In order to provide a quantitative representation for each post, we transformed the words in the training set into numerical representations. Word2vec pretrained models are widely recognized as a significant advancement in natural language processing, as they transform words into vectors with n dimensions. Word2vec trains the initial text corpus using *continuous bag-of-words* (CBOW) and *skip-gram* methods. Word embedding approaches such as *word2vec* and *GloVe* offer unique vector representations for the words in the dictionary but it causes a lack of awareness of the language's internal structure. The lack of consideration for the words' syntactic relationship is a drawback of morphologically rich languages. FastText [41] enhances vector representation for languages with complex morphology by offering embeddings for character n-grams. It represents words by calculating the average of these embeddings. It is an extended version of the *word2vec* model.

FastText represents each word by calculating the average of its character n-grams' vector representations, including the word itself.

Consider the word "large" with n = 3, it can be represented by character n-grams:

<la, lar, arg, rge, ge> and <large>.

The word embedding for the word 'large' can be obtained by adding together the vector representations of all its character n-grams and the word itself.

FastText basically uses both CBOW and Skip-gram models.

### a) Continuous Bag of Words (CBOW):

In the Continuous Bag of Words (CBOW) model [42], the input consists of the context of the target word, and the objective is to predict the word that appears in that context.

In the sentence "I need to apply FastText", the words "I," "need," "to," and "FastText" are provided as input, and the model predicts "apply" as the output.

The input and output data are of the same dimension and are encoded using one-hot encoding. The training process employs a neural network. The neural network consists of three layers: an input layer, a hidden layer, and an output layer. Figure 8 illustrates the functioning of CBOW model.

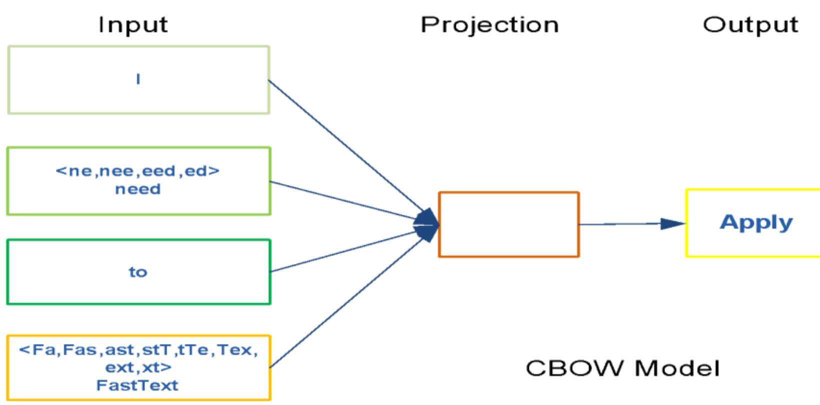



**Fig. 8 Working of CBOW Model**

### b) Skip-gram

Skip-gram operates similarly to CBOW, with the difference that the target word serves as the input, and the model predicts the context surrounding the given word. Additionally, it employs neural networks for the purpose of training. Figure 9 illustrates the functioning of Skip-gram.

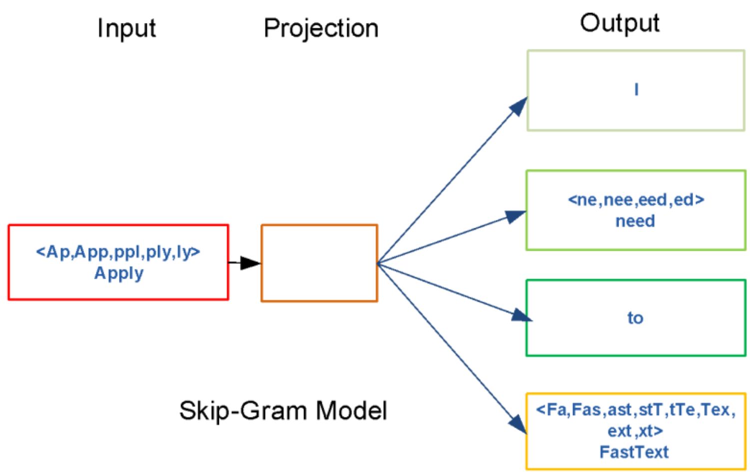



**Fig. 9 Working of Skip-gram Model**

In this work, FastText embedding is used because it captures subword information, making it more effective for handling informal and misspelled social media text. Let the input text sequence be represented as:

$T = \{w_1, w_2, w_3, \dots, w_n\}$ (1)where:

- $T$ represents the input text sequence
- $w_i$ represents the i-th word in the sequence
- $n$ represents the total number of words

Each word is mapped to a vector representation using FastText:

$x_i = FastText(w_i)$ (2)

where:

- $x_i \in R^d$ represents the embedding vector
- $d$ represents the embedding dimension
- FastText(.) represents the embedding function

The output embedding matrix is:

$X = [x_1, x_2, \dots, x_n]$ (3)

where:

- $X \in R^{n \times d}$represents the embedding matrix

This embedding matrix is used as input to the CNN layer.

### *3.3 Feature Classification Using Convolutional Neural Network*

In this work, we created two CNN architecture blocks that perform the analysis for the n-gram lengths of 3 and 4. The training samples are passed into each block of CNN layers, which consist of a convolutional layer, a batch normalization layer, a ReLU layer, and a global average pooling layer. These layers are responsible for extracting features from the input. Every convolution layer is supplied with a collection of filters that assist in extracting features.

ReLU activation unit is employed to process the convolutional output. This unit converts the data into a non-linear format. When the output of the convolution is negative, the ReLU output is set to zero. The vanishing gradient problem limits the utilization of sigmoid units as activation units. ReLU is recommended to prevent such a scenario.

The working of how the Convolutional layer perform the operation is explain as follow:

Let $n$ denote the sentence length and $d$ denote the word embedding dimension. Each sentence can be represented as a matrix:

$$X \in R^{n \times d} \quad (4)$$

where $n$ represents the number of words in the sentence and $d = 300$ represents the embedding dimension.

In this work, m convolution kernels are used, where each kernel produces one feature map, resulting in a final pooled feature vector of dimension m. A convolutional kernel $i$ is defined as:

$$C_i \in R^{k \times d}, i = 1, 2, \dots, m \quad (5)$$

where $k$ represents the filter size corresponding to the n-gram length (3 or 4), and $d$ is the embedding dimension. The convolution operation is applied to extract local features from consecutive word sequences.

The convolutional feature map $F_i$ generated by the $i^{th}$ filter is defined as:

$$F_i = \left(f_{i1}, f_{i2}, \dots, f_{i(n-k+1)}\right) \quad (6)$$

where each feature value is computed as:

$$f_{ij} = \sigma\left(\sum (X[j : j + k, 1 : w] \circ C_i) + b\right) \quad (7)$$

where:

- $F_i$ represents the feature map generated by the i-th convolutional filter,
- $f_{ij}$ represents $j^{th}$ feature value of $i^{th}$ filter,
- $\sigma$ represents the nonlinear activation function (ReLU in this study),
- $b$ represents the bias term,
- $\circ$ denotes the Hadamard (element-wise) product,
- $X[j : j + k, 1 : w]$ represents the input sub-matrix corresponding to $k$ consecutive words.

Each convolutional layer is followed by batch normalization, which stabilizes training and accelerates convergence by normalizing the feature distributions.

After feature extraction, a global average pooling operation is applied to each feature map to reduce dimensionality and retain the most representative features. Unlike max pooling, which selects only the maximum value, global average pooling computes the mean of all feature values in the feature map, preserving the overall distribution of features.

The pooled feature vector z is computed as:

$$\mathbf{z} = [z_1, z_2, \dots, z_m] \quad (8)$$

where, the average pooling function is defined as:

$$z_i = avgpool(F_i) = \frac{1}{(n-k+1)} \sum_{j=1}^{(n-k+1)} f_{ij} \quad (9)$$

Where:

- $(n - k + 1)$ represents the number of features in each feature map.
- m represents number of convolution kernels.
- $\mathbf{z} \in R^m$ is final feature vector

### *3.4 Deep RNN For Psychotic Behavior Analysis*

Psychotic conduct can be represented as time-sequential words in text data when the individual is having a conversation with others. Therefore, a machine learning model that can effectively encode time-sequential data which is very ideal for this type of task. Therefore, this approach utilizes Recurrent Neural Networks (RNNs). The RNN is widely regarded as one of the most prominent deep learning methods for modeling time-sequential data [43]. RNNs primarily comprise recurrent connections that relate the historical information to the current state and hidden states.

The importance of memory in neural networks is quite significant. The conventional RNN algorithms frequently meet a vanishing gradient problem, a constraint in handling long-term data referred to as Long-Term Dependencies. In order to address the issue, researchers created a technique called Long Short-Term Memory (LSTM) [44]. Figure 10 illustrates a single cell unit of a Recurrent Neural Network (RNN). This figure depicts the data flow at time step *t.* This graphic illustrates the mechanisms by which the gates manipulate and transmit the cell and hidden states.

Figure 10 illustrates the LSTM process. It utilizes the feature extracted from the last hidden layer of CNN step and produce new features. The LSTM model can generate two distinct textual representations due to its ability to access both the prior and following contextual information. An LSTM model produces a sequence representation by utilizing the characteristics derived from the CNN. Ultimately, the most strongly associated features are utilized for the final classification.

An LSTM memory block consists of a cell state and three gates: the input gate, the forget gate, and the output gate.

The gates are computed as:

$$i_t = \sigma(W_i[h_{t-1}, z_t] + b_i), f_t = \sigma(W_f[h_{t-1}, z_t] + b_f), o_t = \sigma(W_o[h_{t-1}, z_t] + b_o) \tag{5}$$

The Cell state C is updated as:

$$C_t = f_t \circ C_{t-1} + i_t \circ tanh(W_g[h_{t-1}, z_t] + b_g) \tag{6}$$

The hidden state *H* is expressed as

$$H_t = O_t \circ \; tanh(C_t) \tag{7}$$

where, In LSTM formulation:

- $z_t$ denotes the input feature vector obtained from the CNN layer at time step *t*,
- $C_t$is the current Cell state
- $h_{t-1}$is the previous hidden state,
- $W_i, W_f, W_o, W_g$ are weight matrices for input, forget, output gate and candidate cell state, respectively,
- $b_i, b_f, b_o, b_g$ are bias terms for input, forget, output gate and cell state, respectively.
- where $\sigma$ denotes the sigmoid activation function, and
- $\circ$ indicates element-wise multiplication.

These formulations describe how the LSTM selectively remembers relevant temporal information while discarding less significant patterns from the CNN feature sequence

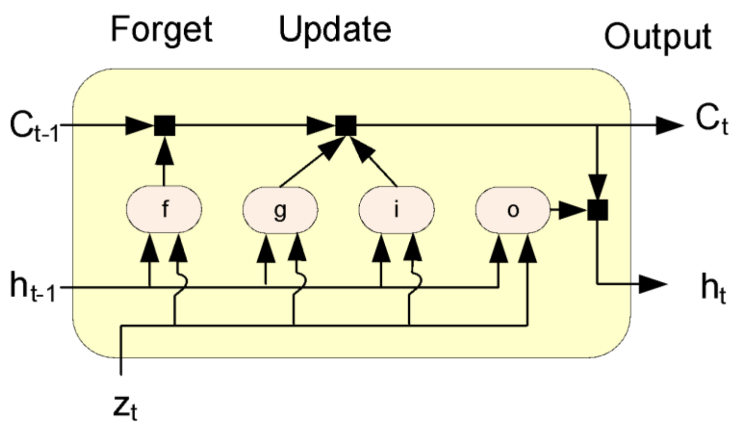



**Fig. 10 A basic Cell Unit of LSTM-based RNN**

The research in this paper is mainly based on text analysis; yet, social media communication widely integrates multimodal features like emoticons, images, and metadata that convey the sentiment and levels of engagement of users. They can complement literary features to ease the interpretation of psychological states. In future work, we will integrate multimodal features, like emoji frequencies, sentiment-weighted emoji embeddings, and visual emotional representations extracted from shared images, into ensemble learning. We will analyze metadata, such as the timing of postings and the rates of user activity, to obtain a complete picture of psychotic or depressive inclinations.

## 4. EXPERIMENTAL SETUP

### *4.1 Implementation Environment*

The experiment utilizes MATLAB image processing and statistical tools on a Windows 10 64-bit operating system running version 2023a. All experiments utilize a 2.4 GHz Intel Core 32-based Processor with 16 GB of RAM.

### *4.2 Performance Parameters*

The confusion-matrix is one of the main quality metrics assessed by the suggested hybrid method [45]. A table II is showing the correlation between the actual and anticipated classes created by the proposed technique.

Table II: Confusion Matrix

| Actual Class | Predicted class | |
|---|---|---|
| | Positive | Negative |
| Positive | TP (true positive) | FN (false negative) |
| Negative | FP (false positive) | TN (true negative) |

Accuracy, sensitivity, specificity, precision, and the F1-score are the metrics used to evaluate proposed approach for classifying psychotic activity. You may find all of the metrics in Table III. According to Table II, the values are obtained by calculating the following: true positive (TP), false positive (FP), false negative (FN), and true negative (TN).

Table III: Performance Measures

| Name | Representation | Range |
|---|---|---|
| Accuracy (Acc) | $\frac{TP + TN}{TP + FP + FN + TN}$ | [0–100] |
| Sensitivity (Sen) | $\frac{TP}{TP + FN}$ | [0–100] |
| Specificity (Spe) | $\frac{TN}{FP + TN}$ | [0–100] |
| Precision (Pre) | $\frac{TP}{TP + FP}$ | [0–100] |
| F1-score | $2 \times \frac{Precision * Sensitivity}{Precision + Sensitivity}$ | [0–100] |

Where, TP-true positive, FP-false positive, TN-true negative, FN-false negative.

### *4.3 Data Splitting and Training Protocol*

The dataset was divided into training, validation, and testing sets using a stratified sampling strategy to preserve class distribution.

- 70% Training Set
- 15% Validation Set
- 15% Test Set

The training set was used to optimize model parameters, the validation set was used for hyperparameter tuning and early stopping, and the test set was strictly reserved for final performance evaluation.No data leakage occurred between sets. All preprocessing steps were applied consistently across splits.

### *4.4 Hyper Parameters*

The hyperparameters that were utilized to retrain the predefined models are listed in Table IV. For the CNN+LSTM networks, the sgdm solver type is used. The initial learning rate is set to 0.001 for all type of networks. The training and testing sets are divided into multiple batches using batch training. Every batch size varies from 16 to 64 items in it. The maximum epoch is set to 25. L2 regularization is also adopted to minimize the overfitting the model and it is set to 0.01. cross-entropy loss function was trained for 25 epochs.

Table IV Hyper parameters used for training the proposed model

| Parameters | CNN | CNN+LSTM |
|---|---|---|
| Solver type | SGDM | SGDM |
| Learning Rate | 0.001- 0.0001 | 0.001- 0.0001 |
| Gradient Threshold Method | l2norm | l2norm |
| Loss function | categorical cross-entropy | categorical cross-entropy |
| L2 Regularization | 0.01 | 0.01 |
| Batch Size | 16 - 128 | 16 - 128 |
| Max Epochs | 25 | 25 |

### *4.5 Hyperparameter Optimization*

In order to increase the performance and robustness of the presented model, we conducted an extended hyperparameter optimization process through the grid search approach. The critical parameters including learning rate, batch size, dropout rate, number of filters in CNN, and number of LSTM units were varied in an organized manner. The best combination was determined based on validation accuracy and F1-score.

The best configuration of the grid search yielded a learning rate of 0.0001, batch size of 64, and 48 LSTM units (see TABLE V). it achieved the most reliable convergence with the lowest loss by using this configuration. When compared to the baseline, the best configuration increased the overall accuracy by +1.2% and provided greater precision and F1-scores.

Although this research employed grid search because of computational practicability, more advanced techniques like Bayesian optimization or genetic algorithms will be investigated in subsequent work to further automate hyperparameter tuning and decrease search time.

Table V : Hyper parameters optimization using grid search approach

| Parameter | Range Tested | Optimal Value | Effect on Accuracy |
|---|---|---|---|
| Learning Rate | 0.001, 0.0005, 0.0001 | 0.0001 | +0.6% |
| Batch Size | 16, 32, 64, 128 | 64 | +0.3% |
| Dropout Rate | 0.2, 0.3, 0.5 | 0.3 | +0.2% |
| LSTM Units | 16, 32, 48, 64 | 48 | +0.3% |
| | | | |

### *4.6 Reproducibility Statement*

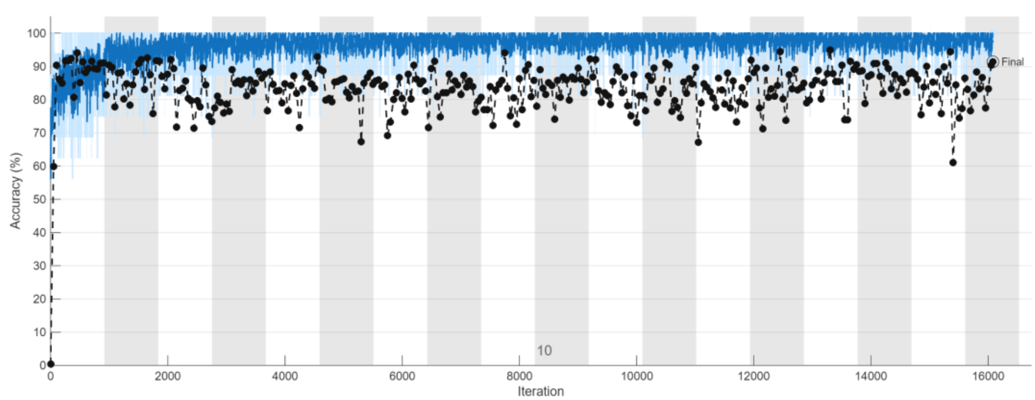
To ensure reproducibility, experiments were conducted using a fixed random seed. Multiple experimental runs were performed, and average performance metrics were reported. All preprocessing steps, hyperparameters, and training configurations are documented to enable replication.

## 5. RESULTS & DISCUSSION

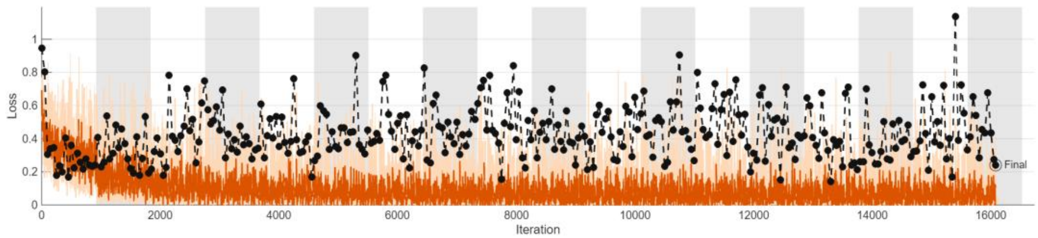
### *5.1 Experimental Results*

Figure 11 illustrates the training and validation loss curves of the proposed FastText + CNN + LSTM model. The model exhibits a smooth and consistent convergence trend, with the loss decreasing steadily over successive iterations. Convergence was achieved through the use of a small learning rate (0.0001) and Stochastic Gradient Descent with Momentum (SGDM), which prevented oscillations and ensured stable optimization. The loss reached a plateau after approximately 16000 iterations, indicating that the model successfully learned discriminative features without overfitting.

To ensure robustness, the model was trained and validated five times using random weight initialization. The average accuracy across runs showed a variance of less than ± 0.8%, demonstrating high consistency in performance. This stability suggests that the proposed architecture effectively generalizes to unseen data.




(a)




(b)

**Fig. 11 Training Progress Graph of Proposed Model (a) Accuracy vs Epochs (b) Loss vs Epochs**

The MATLAB programming language is used to implement the suggested study. The evaluation of the models' performance is based on the criteria of accuracy, /r precision, /r recall, /r and F1 score/.

Table VI shows the confusion matrix of proposed model. Table VII presents a comparison/r between the existing model and the suggested ensembled model.

TABLE VI. Confusion Matrix

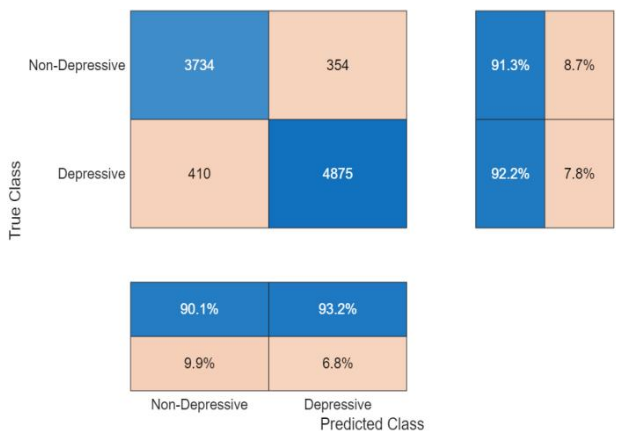



Table VII. Comparative performance of proposed methodology

| **Model** | **Precision (Depressive)** | **Recall / Sensitivity (Depressive)** | **F1-Score (Depressive)** | **Precision (non-depressive)** | **Recall (non-depressive)** | **F1-Score (non-depressive)** | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|
| CNN | 0.8861 | 0.8547 | 0.8700 | 0.8472 | 0.8784 | 0.8625 | 0.8680 |
| CNN + LSTM | 0.9014 | 0.8886 | 0.8949 | 0.8808 | 0.8931 | 0.8869 | 0.8905 |
| FastText + CNN + LSTM (Proposed) | 0.9324 | 0.9225 | 0.9273 | 0.9012 | 0.9135 | 0.9071 | 0.9184 |
| BERT | 0.9364 | 0.9229 | 0.9296 | 0.9212 | 0.9345 | 0.9278 | 0.9310 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| (Base, Fine-tuned) | | | | | | | |

After incorporating Reddit data and performing an extended grid search for hyperparameter tuning, all baseline models exhibited moderate performance improvement. As shown in Table VII, the proposed FastText + CNN + LSTM model achieved an overall accuracy of 91.84 % and an F1-score of 0.917, which is higher than the CNN-only and CNN + LSTM architectures by approximately 2–3 %.

To benchmark the ensemble model against a transformer-based architecture, a fine-tuned BERT-base-uncased model was trained on MATLAB with the same dataset. The fine-tuned BERT achieved an overall accuracy of 93.10 % and an F1-score of 0.928. These results demonstrate that BERT model got some marginal improvement (≈1.2 %) over the proposed model. This improvement is mostly due to BERT's capacity to find subtle linguistic patterns and word associations which depends on context. However, the BERT model required more training time and computing power to run.

From the Table VII, we can draw the conclusion that the FastText + CNN + LSTM model provides an acceptable balance between performance and computational cost. As a result, it is an excellent option for mental health analytical applications that require fast processing or that operate on a large scale.

Figure 12 illustrates a comparison of the performance of the proposed FastText + CNN + LSTM model against the baseline CNN and CNN + LSTM and BERT models. The proposed ensemble model achieves the higher accuracy and an F1-score due to its primary factors (FastText, CNN, and LSTM which handles informal text effectively, captures key local patterns, and preserves contextual information across longer sequences, respectively) which contributes to this enhanced performance are three design decisions.

The performance gain can be attributed to the complementary strengths of the hybrid architecture. FastText enhances subword representation, CNN extracts local semantic patterns, and LSTM preserves long-range contextual dependencies. This layered representation improves discrimination between depression-related and non-depression posts.

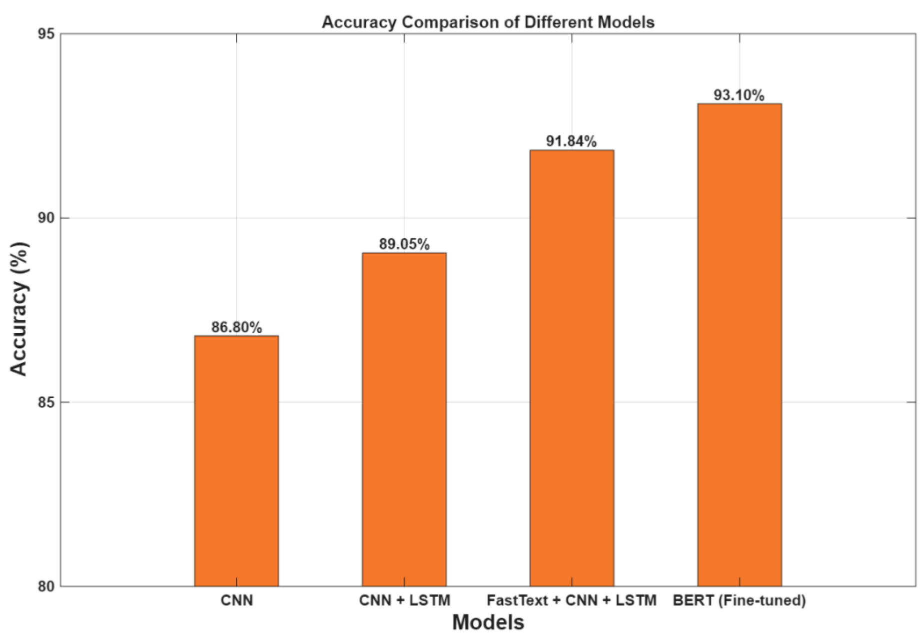



**Figure 12. Comparison Graph of Proposed Method Against other Method**

## *5.2 Ablation Study*

To evaluate the impact of each component in the proposed model pipeline, we conducted an ablation study using five different model configurations. The configurations were as follows: Model M1 was CNN using Word2Vec embeddings; Model M2 was CNN using FastText embeddings; Model M3 was CNN using an LSTM layer; Model M4 was LSTM using FastText embeddings but without the CNN; and Model M5, which is the proposed model, was a CNN, LSTM, and FastText embeddings. As shown in Table VIII, the relative performance of these configurations was evaluated using accuracy, precision, recall, and F1-score. From the table it is clear that when comparing Model M1 and Model M2, it is found that using FastText instead of Word2Vec embeddings improves accuracy by about 3%. This improvement arises from the fundamental difference in how the two methods handle word representation. Word2Vec views each word as a standalone token and is unable to deduce meaning for infrequent or out-of-vocabulary words, confining its capacity to deal with noisy, user-created social media text. Conversely, FastText views each word as a combination of its character n-grams and so is able to encompass morphological and subword information (e.g., prefixes, suffixes, and misspellings). This feature enables FastText to generalize more across variations of emotionally expressive terms, which abound on informal mental health tweets.

Moreover, the accuracy of the model improves by 0.6% when one LSTM layer is swapped with CNN (Model M2 compared to Model M3). Long-range dependencies are not within the capability of CNN alone. However, the LSTM module is very good at capturing temporal and context dependencies between words and phrases across the sequence. In addition, the proposed architecture's CNN-LSTM ensemble yields the optimal overall performance which exhibits the complementary strengths of the convolutional and recurrent neural network components., By stacking CNN and LSTM layers together, the model benefits both spatial abstraction and sequential memory, leading to increased representational power. This synergy allows the ensemble to capture subtle expressions of psychotic or depressive conduct which may otherwise be overlooked in single-component architectures.

TABLE VIII. contribution of each component to overall performance of our proposed pipeline

| **Model** | **Accuracy (%)** | **Precision** | **Recall** | **F1-score** |
|---|---|---|---|---|
| M1 (Word2Vec + CNN) | 86.3 | 0.85 | 0.86 | 0.85 |
| M2 (FastText + CNN) | 89.1 | 0.87 | 0.88 | 0.87 |
| M3 (CNN + LSTM) | 88.4 | 0.89 | 0.90 | 0.89 |
| M4 (FastText + LSTM) | 89.7 | 0.88 | 0.89 | 0.89 |
| M5 (FastText + CNN + LSTM) | 91.9 | 0.92 | 0.91 | 0.92 |

### *5.3 Statistical Significance Analysis*

To validate the performance improvement of the proposed FastText + CNN + LSTM model, a statistical significance test was performed. The performance of the neural network models may change based on the random initialization of the weights, the shuffling of the data, and the training conditions. To validate the performance stability, each model was trained and tested five times independently with different random seeds. The average accuracy and standard deviation were calculated to measure the performance stability.

The statistical significance test was performed to compare the performance of the proposed model and the baseline models using a paired t-test. The t-test is a widely used statistical test for comparing machine learning models over multiple runs. The t-statistic is given by:

$$t = \frac{\bar{d}}{s_d/\sqrt{k}} \quad (8)$$

where:

- $\bar{d}$= mean difference between paired observations
- $s_d$= standard deviation of the differences
- $k$= number of runs
- $t$= t-statistic

The corresponding p-value is then calculated. A p-value less than 0.05 reveals that the accuracy improvement is statistically significant at a 95% confidence level.

Table IX shows the comparison of mean accuracy, standard deviation, and p-value for the proposed model and the baseline models.

Table IX. Statistical significance analysis of different models

| Model | Mean Accuracy (%) | Standard Deviation | p-value (vs Proposed) | Significance |
|---|---|---|---|---|
| **Word2Vec + CNN** | 85.6 | 0.62 | < 0.001 | Significant |
| **FastText + CNN** | 88.4 | 0.55 | < 0.001 | Significant |
| **CNN + LSTM** | 87.9 | 0.58 | < 0.001 | Significant |
| **FastText + LSTM** | 89.0 | 0.49 | 0.002 | Significant |
| **FastText + CNN + LSTM (Proposed)** | 91.9 | 0.41 | — | — |

# 6. CONCLUSION

This paper presents the development of an ensemble deep learning model for the identification of a psychotic behavior state using social media posts. The suggested model combines FastText embeddings with a hybrid CNN–LSTM architecture to accurately determine user's post related to a specific mental disorder, such as depression, anxiety, etc. Each post undergoes pre-processing before being transformed into features. The FastText technique is utilized to offer features matrix that are taken from each post. Subsequently, the essential characteristics are retrieved using a pair of CNN block architecture. Next, a deep learning technique called Long Short-Term Memory (LSTM) is utilized to train the time-sequential characteristics that classify depression from posts that do not contain any such descriptions (non-depression posts). Our suggested model is implemented on MATLAB 2023a and assessed using a heterogeneous dataset sourced from Twitter and Reddit, which improved the model's capacity to generalize across multiple communication methods and linguistic expressions. The experimental results showed that the suggested FastText + CNN + LSTM model got 91.9% correct, which is quite near to the fine-tuned BERT model (93.1%) but at a far lower cost. This shows that the suggested method is a useful and effective way to keep track of mental health in real time on a large scale.

# 7. FUTURE DIRECTION

## *7.1 Theoretical and Practical Implications*

The present study makes a valuable contribution to the field of AI-assisted mental health detection by providing an ensemble deep learning model that effectively integrates FastText embeddings with CNN and LSTM layers. This model achieves a balance between computational efficiency and classification accuracy which provides a real-world solution for social media-based early diagnosis of psychological symptoms. Theoretically, it makes a contribution to the existing literature by providing the use of hybrid deep learning and contextual embeddings for analyzing mental health text. Practically, it offers a versatile, cost-effective method that can be deployed in digital wellness monitoring systems, online counseling platforms etc.

## *7.2 Limitations and Future Work*

Despite strong overall performance, certain limitations were observed. Misclassifications mainly occurred in posts containing sarcasm, implicit emotional expressions, or ambiguous context. Short posts with limited contextual information also contributed to prediction errors.

Additionally, overlapping linguistic patterns between anxiety and non-depression posts occasionally reduced classification clarity. These findings indicate that purely text-based approaches may struggle with subtle emotional nuances.

Future work incorporating multimodal signals (e.g., emojis, images, metadata) may help mitigate these limitations.

These future plans will enhance both the social and economic impact of the proposed framework and it will contribute to a more fair and proactive mental health support ecosystem.

### *7.3 Mitigating Future Risks:*

In despite the fact that the model performs effectively, we will emphasis on future works that will address potential risk such as bias in the dataset, incorrect categorization, and strong dependence on linguistic signals. These risks can be mitigated by:

- Making use of a wide range of datasets from a number of web platforms with balanced demographics;
- Making use of Explainable AI (XAI) technologies, such as SHAP and LIME, to make judgments more transparent; and
- Seeking input from mental health experts to validate automated predictions before their usage in clinical interpretation.

### *7.4 Monetary and Implementation Implications*

Deploying the proposed system in real-world mental health monitoring environments would incur moderate costs, primarily related to cloud computation, storage, and data annotation. Based on current computation and data infrastructure estimates, the end-to-end implementation and maintenance of the model would cost depending on the scale of data and frequency of retraining. These costs are considerably lower than traditional mental health screening programs, making the system a cost-effective supplement for public health monitoring and online wellness applications.

### *7.5 Beneficiaries*

The outcomes of this study will benefit data scientists, mental health researchers, digital healthcare practitioners, and policy planners seeking scalable tools for psychological assessment.

**DECLARATION**

## - Competing Interests

- The authors have no competing interests and no relevant financial or non-financial interests.

**- Funding Information**

- No funds, grants, or other support was received.

**- Author contribution**

- Both authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Reeta Bourasi. The first draft of the manuscript was written by Reeta Bourasi and other author commented on the manuscript. Both authors read and approved the final manuscript.

**- Data Availability Statement**

- We utilized data from an existing GitHub repository called "Detecting-Depression-in-Tweets" as well as from Kaggle called “Reddit Mental Health Data”.

**- Research Involving Human and /or Animals**

- Not Applicable

**- Informed Consent**

- Informed consent was obtained from all individual participants included in the study.

**References**

1. Valenzuela, S., Park, N., & Kee, K. F. Is there social capital in a social network site?: Facebook use and college students’ life satisfaction, trust, and participation. J. Comput.-Med. Commun. 14, 875–901 (2009).

2. Al-Saggaf, Y., & Nielsen, S. Self-disclosure on facebook among female users and its relationship to feelings of loneliness. Comput. Hum. Behav. 36, 460–468 (2014).
3. Shen, J. H., & Rudzicz, F. Detecting anxiety through reddit. In Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology-From Linguistic Signal to Clinical Reality (2017).
4. Wang, Y.-C., Kraut, R., & Levine, J. M. To stay or leave? The relationship of emotional and informational support to commitment in online health support groups. In Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work (ACM, 2012).
5. iles, D. C., & Newbold, J. Self-and other-diagnosis in user-led mental health online communities. Qual. Health Res. 21, 419–428 (2011).
6. Weinberger AH, Gbedemah M, Martinez AM, Nash D, Galea S, Goodwin RD (2018) Trends in depression prevalence in the USA from 2005 to 2015: widening disparities in vulnerable groups.Psychol Med 48(8):1308–1315
7. Cambria E, White B (2014) Jumping NLP curves: a review of natural language processing research. IEEE Comput Intell Mag 9(2):48–57
8. Picard RW (2003) Affective computing: challenges. Int J Human Comput Stud 59(1–2):55–64.
9. Shangfei W, Guozhu P, Zhuangqiang Z, Zhiwei Xu (2019) Capturing emotion distribution for multimedia emotion tagging. IEEE Trans Affect Comput. https://doi.org/10.1109/taffc.2019.2900240.
10. Havigerova´ JM, Haviger J, Kucˇera D, Hoffmannova´ P (2019) Textbased detection of the risk of depression. Front Psychol 10:513
11. Tao J, Tan T (2005) Affective computing: a review. In: Proceedings of the international conference on affective computing and intelligent interaction, pp 981–995
12. ang H, Willis A, De Roeck A, Nuseibeh B (2012) A hybrid model for automatic emotion recognition in suicide notes. Biomed Inform Insights 5(1):17–30
13. Desmet B, Hoste VR (2013) Emotion detection in suicide notes. Expert Syst Appl 40(16):6351–6358
14. Allouch M, Azaria A, Azoulay R, Ben-Izchak E, Zwilling M, Zachor DA (2018) Automatic detection of insulting sentences in conversation. Paper presented at: Proceedings of the 2018 IEEE international conference on the science of electrical engineering in Israel (ICSEE); pp 1–4; IEEE.
15. Yoo, M., Lee, S., & Ha, T. Semantic network analysis for understanding user experiences of bipolar and depressive disorders on reddit. Inf. Process. Manag. 56, 1565–1575 (2019).
16. De Choudhury, M., Gamon, M., Counts, S., & Horvitz, E. Predicting depression via social media. In Seventh international AAAI Conference on Weblogs and Social Media (2013).
17. Gkotsis, G. et al. Characterisation of mental health conditions in social media using informed deep learning. Sci. Rep. 7, 45141 (2017).
18. Ramin Safa, Peyman Bayat, and Leila Moghtader. 2022. Automatic detection of depression symptoms in twitter using multimodal analysis. The Journal of Supercomputing 78, 4 (2022), 4709–4744.
19. Esteban A Ríssola, Seyed Ali Bahrainian, and Fabio Crestani. 2020. A dataset for research on depression in social media. In Proceedings of the 28th ACM conference on user modeling, adaptation and personalization. 338–342.
20. Jina Kim, Jieon Lee, Eunil Park, and Jinyoung Han. 2020. A deep learning model for detecting mental illness from user content on social media. Scientific reports 10, 1 (2020), 1–6.
21. Md Rafiqul Islam, Muhammad Ashad Kabir, Ashir Ahmed, Abu Raihan M Kamal, Hua Wang, and Anwaar Ulhaq. 2018. Depression detection from social network data using machine learning techniques. Health information science and systems 6 (2018), 1–12.
22. Harnain Kour and Manoj Kumar Gupta. 2022. Depression and Suicide Prediction Using Natural Language Processing and Machine Learning. In Cyber Security, Privacy and Networking: Proceedings of ICSPN 2021. Springer, 117–128.
23. KS Aswathy, PC Rafeeque, and Reena Murali. 2019. Deep learning approach for the detection of depression in twitter. In Proceedings of the International Conference on Systems, Energy Environment (ICSEE).
24. Tapotosh Ghosh, Md Hasan Al Banna, Md Jaber Al Nahian, Mohammed Nasir Uddin, M Shamim Kaiser, and Mufti Mahmud. 2023. An attention-based hybrid architecture with explainability for depressive social media text detection in Bangla. Expert Systems with Applications 213 (2023), 119007.
25. Zhang H., Liu Y., & Zhao X. (2024). RMER-DT: Robust multimodal emotion recognition in conversational contexts based on diffusion and transformers. IEEE Transactions on Affective Computing, (early access).
26. Zhou Y., Chen R., & Wang S. (2023). RAFT: Robust Adversarial Fusion Transformer for multimodal sentiment analysis. Array, 100445.
27. Li W., Huang M., & Guo J. (2023). CIME: Contextual interaction-based multimodal emotion analysis with enhanced semantic information. Information Fusion, 96, 102131.
28. Li F., Xu H., & Zhang T. (2023). Contrastive-based removal of negative information in multimodal emotion analysis. Neural Networks, 163, 524-538.
29. Wang J. et al. (2023). Towards trustworthy image super-resolution via symmetrical and recursive artificial neural networks. Pattern Recognition, 145, 109920.

30. Chen D., Zhao K., & Xu Y. (2023). Knowledge generation and distillation for road segmentation in intelligent transportation systems. IEEE Transactions on Intelligent Transportation Systems, 24(8), 8895–8907.
31. Xu J., Wei S., & Zhao L. (2024). Generalizable deepfake detection via spatial kernel selection and halo attention network. Pattern Recognition Letters, 176, 27-35.
32. Zhang Y., Wang Z., & Sun C. (2024). Deepfake detection via feature refinement and enhancement network. IEEE Access, 12, 14450–14462.
33. Liu Q. et al. (2024). Advancing federated domain generalization in ophthalmology: Vision enhancement and consistency assurance for multicenter fundus image segmentation. Medical Image Analysis, 98, 103075.
34. Singh A., Kaur S., & Verma P. (2023). A client-server-based recognition system: Non-contact single/multiple emotional and behavioral state assessment methods. Expert Systems with Applications, 222, 119744.
35. Huang L. et al. (2023). Dynamic spectral graph anomaly detection. Knowledge-Based Systems, 263, 110271.
36. Detecting-Depression-in-Tweets/sentiment_tweets3.csv at master · viritaromero/Detecting-Depression-in-Tweets · GitHub
37. Depression: Twitter Dataset + Feature Extraction (kaggle.com)
38. depression-detection/data/tweets_combined.csv at master · swcwang/depression-detection · GitHub
39. Twitter Depression Dataset (kaggle.com)
40. Reddit Mental Health Data (kaggle.com)
41. Isnaini Nurul Khasanah," Sentiment Classification Using fastText Embedding and Deep Learning Model", Procedia Computer Science, Volume 189, 2021, Pages 343-350, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2021.05.103.
42. Tomas Mikolov and Kai Chen and Greg Corrado and Jeffrey Dean , "Efficient Estimation of Word Representations in Vector Space" , 2013, https://arxiv.org/abs/1301.3781
43. Gardner-Lubbe S (2020) Linear discriminant analysis for multiple functional data analysis. J Appl Stat pp 1–17
44. Uddin MZ, Torresen J (2019) Activity recognition using smartphone sensors, robust features, and recurrent neural network. In: 2019 13th international symposium on medical information and communication technology (ISMICT), May 2019.