*I.J. Information Technology and Computer Science,* **2025, 6, 70-94**
Published Online on December 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2025.06.04

# Early Detection of Stress and Anxiety Using NLP and Machine Learning on Social Media Data

**Ravi Arora**
Lingaya Vidyapeeth, Faridabad, Haryana 121002, India
Department of Information Technology, Galgotias College of Engineering and Technology, Greater Noida, Uttar Pradesh 203201, India
E-mail: raviaroratheap04@gmail.com
ORCID iD: https://orcid.org/0009-0008-1676-6066

**S. V. A. V. Prasad**
Lingaya Vidyapeeth, Faridabad, Haryana 121002, India
E-mail: svavprasad@lingayasvidyapeeth.edu.in
ORCID iD: https://orcid.org/0009-0006-9594-3838

**Arvind Rehalia**
Department of Information Technology, Bharati Vidyapeeth's College of Engineering, New Delhi, Delhi, 110063, India
E-mail: rehaliaarvind@gmail.com
ORCID iD: https://orcid.org/0000-0002-2184-3432

**Nikhil Kaushik**
Department of Electronics and Communication Engineering, Lingaya Vidyapeeth, Faridabad, Haryana 121002, India
E-mail: Nikhil.sir46@gmail.com
ORCID iD: https://orcid.org/0009-0007-1977-5585

**Anil Kumar***
Department of Applied Science (Physics), Bharati Vidyapeeth's College of Engineering, New Delhi 110063, India
E-mail: dranilchhikara@gmail.com
ORCID iD: https://orcid.org/0009-0000-8333-4337
*Corresponding author

Received: 21 July 2025; Revised: 24 September 2025; Accepted: 02 November 2025; Published: 08 December 2025

**Abstract:** Stress and anxiety are some of the most public mental health illnesses that people in the current society face. It is important to determine these conditions early to be able to effectively promote the well-being of individuals. This research work presents the possibility of identifying stress and anxiety through social media (SM) data and an anonymous survey, by machine learning (ML) and natural language processing (NLP). The paper starts with data collection, using the DASS-21 questionnaire and a sample of tweets obtained from Twitter users from India, aimed at determining which language is associated with stress and anxiety. The gathered data is pre-processed in some of the steps, such as URL removal, lower casing, punctuation removal, stop words removal, and lemmatization. After data preprocessing, the textual content is transformed into numerical form through Word2Vec to facilitate pattern analysis. To enrich the analysis of the main topics in the dataset, the Latent Dirichlet Allocation (LDA) and the Non-Negative Matrix Factorization (NMF) techniques are applied. For the classification, the work uses ML algorithms such as Support Vector Machine (SVM), Random Forest (RF), and Long Short-Term Memory (LSTM) networks. Lastly, the project involves an application created with Streamlit to allow the user to interact with the model.

**Index Terms:** Stress Detection, Anxiety Prediction, Natural Language Processing, Social Media, Machine Learning, Long Short-term Memory, Real-time Monitoring.

## 1. Introduction

Stress and anxiety have become fixed health problems within society, especially due to the current advanced and

 

pressurized world [1]. A survey made based on other studies has revealed that stress and anxiety, especially if chronic, not only affect the mental condition of an individual but also cause certain physical conditions, including cardiovascular diseases, immune deficiencies, and metabolic diseases. Stress and anxiety should be detected early for proper management of the conditions to be accorded [2]. As more portable technology and mobile health (mHealth) applications evolve, it has become possible to screen physiological signals such as skin conductance, heart rate variability, and respiratory frequency in real time [3]. These biomarkers offer important information about the person's emotional condition and can be used as markers of stress and anxiety. These physiological data flows can be processed with ML techniques to identify patterns that signal psychological problems and create effective early health care [4].

Statistics show that semi-mental disorders occur before the age of 14, and three-quarters of such cases occur by the time a person is 24 years old. However, it is believed that there are not enough global resources available to help identify, support, and treat mental diseases [5]. While primary care health services for mental illness are offered by 87% of governments globally, 30% lack particular programs, and 28% do not set aside funds specifically for mental health [6]. Notably, the majority of mental illnesses cannot be accurately diagnosed by a laboratory test. Typically, mental status tests, observations from family members or acquaintances, and patient reports of their experiences are used to make a diagnosis. Also, investigate the possible function of SM in identifying and forecasting affective illnesses in people in light of these difficulties [7]. Memory impairment, challenges with concentration, reduced interest in sexual activities, excessive eating leading to weight gain, diminished appetite resulting in weight loss, worthlessness, helplessness, restlessness, irritability, and contemplation of suicide constitute a selection of significant symptoms associated with mental health disorders [8]. Stress and anxiety have increased among people from the COVID pandemic, according to a recent study conducted by the Center of Healing (TCOH), a preventive health platform based in Delhi. The survey reveals a rise in stress and anxiety, with 74% and 88% of Indians reporting experiencing stress and anxiety.

The words and feelings used in these SM posts can be used as clues to key depression symptoms of unimportance, fault, helplessness, and self-loathing [9]. Furthermore, people with depression often fail to participate in claims that language, action, and social bonds create reliable statistical replicas that detect and predict major depressive disorder (MDD) in a subtle way [10]. Determining the level to which persons with mental health disorders like stress and anxiety remain undiagnosed is the primary focus of this work, which also analyses the traumatic practices of the Indian population through the main wave of the coronavirus epidemic. The main aim is to create an ML framework that will help in the early identification of stress and anxiety from text data from SM using NLP and deep learning (DL).

*Objectives*

- To prepare SM text data for NLP analysis by cleaning it.
- To perform vectorization and topic modeling analysis (LDA and NMF) to determine topics related to stress and anxiety
- To identify stress and anxiety levels using ML models like SVM, RF, and LSTM.
- To design a web application where the user can enter textual data and get immediate suggestions on possible stress or anxiety.
- To assess the accuracy of stress and anxiety detection by the developed models.

This work is an integration of the psychological assessment with the NLP methods used in topic modeling and classification of data from large SM such as Twitter for large-scale identification of stress and anxiety. The comparison of the results obtained with the help of classical ML and LSTM makes it possible to compare the efficiency of the methods in a particular dataset. The development of an application that can be utilized by the public provides functional value and may contribute to enhancing public screening for health and the availability of early intervention tools.

The main contribution of the developed model is detailed below,

- ***Real-Time Detection Framework:*** Provides a framework for real-time detection of stress and anxiety using SM information to intervene timely in mental health issues.
- ***Integration of Advanced NLP Techniques:*** Uses NLP methods, including topics and vectors, in combination with DASS-21 for accurate categorization of stress and anxiety.
- ***Interactive Application:*** Presents a simple and intuitive web application based on Streamlit, which makes it possible to create self-evaluation and feedback, making it more accessible for users who do not have special programming knowledge.
- ***Comprehensive Model Evaluation:*** Compares three ML techniques (SVM, RF, LSTM) to determine the most effective approach to stress and anxiety detection in text data.

The paper is planned as follows: the literature analysis is in section 2, the problem statement and motivation are in section 3, the suggested technique is in section 4, the results and discussion are in section 5, and the conclusion is at the end of section 6.

## 2. Literature Review

Some of the recent works related to the study are described as follows:

Oryngozha et al. in [11] focus on identifying and evaluating postings in Reddit's academic forums that are about stress. These communities have emerged as a hub for scholarly discussions and assistance as a result of remote employment and online learning. Using Reddit data, the author used ML and NLP classifiers to categorize text as stressed or not. The author then gathered and examined posts from several scholarly subreddits. The designed technique gained 77.78% accuracy and an F1 score of 0.79 using Reddit data.

Zhu et al. in [12] focus on the sentiment polarity of posts about anxiety on Chinese SM and create models for sentiment classification by combining the posts' language and semantic characteristics. Before presenting a unique recursive feature selection approach to preserve significant linguistic features, the author first extracted the linguistic features from posts using the simplified Chinese-Linguistic Inquiry and Word Count (SCLIWC) vocabulary. Second, to better represent the posts, the author created a Text CNN-based model that examines the posts' deep semantic properties and fuses them. Lastly, the author created a post-level sentiment analysis dataset based on posts about anxiety on Sina Weibo to perform anxiety analysis on Chinese SM.

Zhang et al. in [13] offered a deep knowledge-aware depression recognition scheme to identify and describe the detection criteria for SM users who may be at risk for depression. To assess the created IT artifact, the author used several empirical analyses, which demonstrated that domain knowledge significantly boosts performance. The presented study has important ramifications for IS research in generalizable design principles, digital trace use, and knowledge-aware ML. In practice, the planned IT artifact's early detection and factor explanation can help manage depression and facilitate extensive population mental health assessments.

Santos et al. in [14] presented the SetembroBR corpus, a novel dataset for the education and growth of predictive models for anxiety and depression in the Portuguese language based on data before a diagnosis. Numerous experiments explaining the problems of depression and anxiety disorder forecast from SM data demonstrate the use of the presented corpus. This contains network-related and textual data regarding 3.9 thousand people on Twitter who self-reported receiving treatment or an evaluation for a mental illness.

Pande et al. [15] created a system that can identify depression in people by looking at their SM posts. Data mining techniques have been used in the given work to analyze SM data and identify those who may be at risk for depression. Sentiment analysis (SA), NLP, ML, user profiling, and early warning systems were all covered by the author. The sentiment and emotional tone of SM posts can be effectively captured by the bag-of-words (BoW) technique, which is essential for identifying depression. To determine the degree of depression, it uses the Naïve Bayes (NB) classifier, which is well-known for its accuracy in classifying text data.

Mo et al. [16] introduced MMD-AS, a multimodal data-driven anxiety broadcast scheme, and conducted tests using the health information gathered from more than 200 seafarers using cell phones. To enhance the model's performance, the presented outline's feature extraction, feature selection, dimension reduction, and anxiety inference are all trained simultaneously. To eliminate redundant features and find the best feature subset, the non-differentiable problem of feature combination is solved using a feature selection scheme based on the improved fireworks procedure.

Table 1. Summary of the existing works

| Author | Technique | Advantage | Disadvantage |
|---|---|---|---|
| Oryngozha, et al. [11] | Logistic Regression with Bag of Words on the Dreaddit dataset | High accuracy and F1 score for stress detection (77.78% accuracy, F1 of 0.79) | Limited to stress detection and Reddit academic forums; may not generalize to other platforms or emotional states |
| Zhu et al. [12] | Recursive Feature Selection and Text CNN model on Chinese SM (Weibo) using SCLIWC dictionary | An effective combination of language and semantic features for sentiment analysis in Chinese SM | Limited to simplified Chinese language and Weibo posts; complex to generalize across languages and platforms |
| Zhang et al. [13] | Deep Knowledge-aware Depression Recognition scheme | Leverages domain knowledge to enhance model performance; significant for mental health assessment | May require extensive domain-specific knowledge; focus on depression only. |
| Santos et al. [14] | SetembroBR corpus with anxiety and depression analysis in Portuguese | Provides a novel dataset in Portuguese for predictive mental health analysis, useful for native speakers | Dataset limited to the Portuguese language; requires preprocessing for non-Portuguese-speaking users |
| Pande et al. [15] | Naïve Bayes classifier with Sentiment Analysis and Bag of Words | Effective in capturing sentiment and emotional tone in posts; suitable for identifying depressive tendencies | Relies on Bag of Words and Naïve Bayes, which may be less accurate for nuanced text data |
| Mo et al. [16] | MMD-AS anxiety screening system with Improved Fireworks Algorithm for feature selection | Multimodal data integration improves model performance for anxiety screening. | Focused on data from seafarers, which may limit generalizability to broader populations |
| Abbas et al. [17] | BERT-RF feature engineering combining BERT embeddings with RF on a depression dataset | High accuracy due to contextualized embeddings and probabilistic features; BERT-RF enhances depression detection | Computationally intensive due to BERT and RF; specific to depression detection on Twitter |

Abbas et al. in [17] offered an innovative framework for using ML techniques to analyze SM user material to diagnose depression early. Here, the author used a benchmark depression collection with 20,000 tagged tweets from user profiles classified as depressed or not to build sophisticated ML models. To extract contextualized embedding and probabilistic features from textual input, the work presents a novel BERT-RF feature engineering technique. Contextualized Embedding characteristics are extracted using the Bidirectional Encoder Representations from Transformers (BERT) technique, which is based on the Transformer architecture. Class probabilistic characteristics are then produced by feeding these features into an RF model. These distinguishing characteristics help to improve the detection of depression in SM.

The comparison of the existing literature reviews is detailed in Table 1.

## 3. Problem Statement and Motivation

Stress, anxiety, and depression have become rampant over the last of years, especially due to the increased use of SM and technology. Reddit, Twitter, Weibo, and other social networks have turned into a venue where users do not hesitate to share their emotional issues, which makes these platforms informative for the study of mental health [18]. Nevertheless, the current approaches to mental health screening, which are based on questionnaires and clinician-administered evaluations, do not take advantage of the rich, continuously streaming data that are available on SM. This gap underlines the necessity of the search for automated methods that can enable the identification of the markers of mental health conditions by examining user-generated content on the Internet [19]. Therefore, the need to create efficient, accurate models to identify language and forecast mental health status from SM text increases.

Rising trends about stress, anxiety, and depression require more active and easily replicable approaches to screening. It is therefore important that people with mental illness be identified early and receive the necessary treatment before the conditions worsen. Since mental health care is many a times limited, the use of automated systems can go a long way in providing initial evaluation as well as alarms.

## 4. Proposed Methodology

This research integrates psychological assessment standards with NLP-based topic modeling and classification methods applied to SM data, specifically Twitter, to detect stress and anxiety patterns at scale.

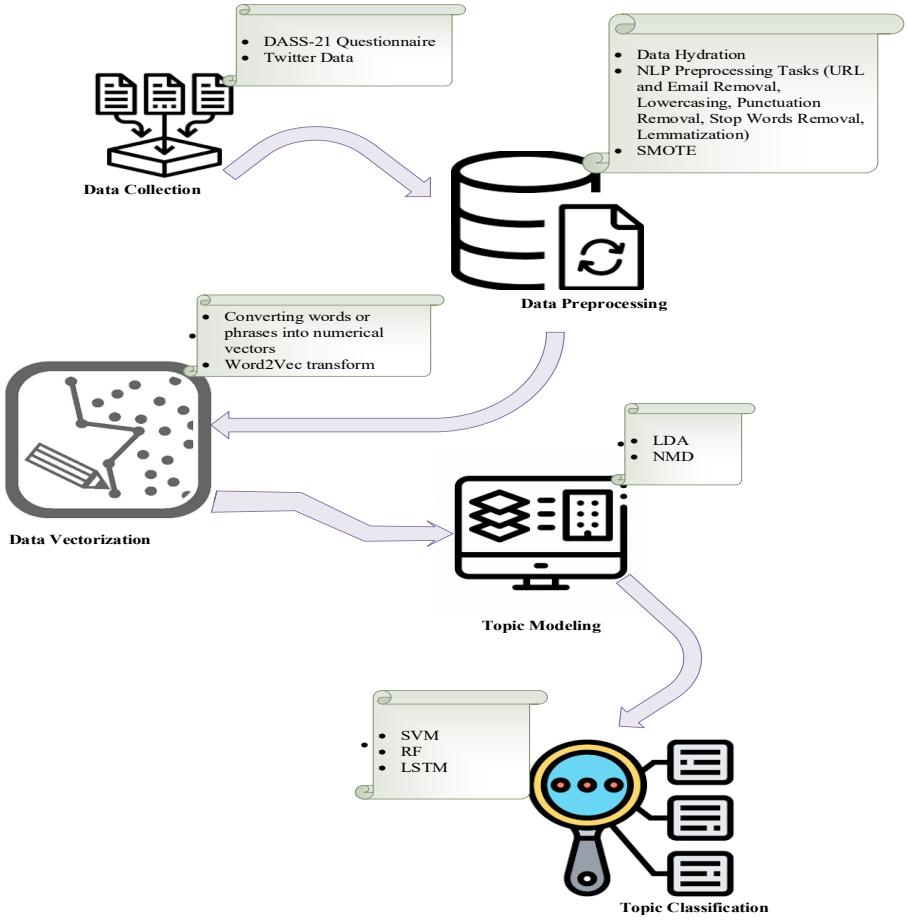


Fig.1. Architecture of the developed technique

The study's use of both traditional ML and advanced DL models (LSTM) provides a comparative analysis of different methods in a unique dataset. The development of an interactive tool for public use adds practical value, potentially supporting mental health monitoring initiatives and enabling widespread access to early detection resources. The procedure of the developed technique is shown in Fig.1.

### *4.1. Data Description*

DASS-21 is the main instrument for assessing anxiety and stress levels in this study [20]. Participants were recruited through online platforms targeting the Indian populace, and before their participation, informed consent was obtained. For the DASS-21 questionnaire, participants were recruited online via email invitations and social media posts targeted at Indian users. Informed consent was obtained before participation. To integrate survey data with content from Twitter, participants were requested to optionally provide their Twitter handles, allowing a subset of responses to be related directly to social media activities. With the aim of contrasting survey data trends with time series patternization of anonymous geolocated tweets from India, this would serve to evaluate the correlation of self-reported mental health indicators with that of public discourse. A smaller set of participants voluntarily provided their Twitter handles for researchers to associate survey responses and social media activity with one another, enabling comparison between self-reported mental health status and certain linguistic features extracted from tweets. Separately, a larger data set comprising 25,424 Twitter comments and replies from India was prepared using their tweet IDs [21] and subsequently hydrated from the Twitter Hydrator. This data set contains the tweets in the full 'text' column, processed for natural language processing (NLP)-based early detection of stress and anxiety in the larger population.

Twitter constituted the primary source of social media data on the grounds of its public-by-default nature, comfortable data accessibility via API, and the platform's prominence in these discussions in real-time, especially around news items and public opinion. On the contrary, in comparison, platforms like Facebook and Instagram have restricted data access and usually tend to favor private or visual content, which in turn may not be that conducive for large-scale textual analyses.

### *4.2. Data Preprocessing*

Data hydration, also known as data lake hydration, is the process of acquiring data into an object. While an object is preparing for data infusion, it is in a state of readiness for hydration. To select and fill objects with pertinent data, there are several data hydration techniques available. Successful data hydration goes beyond simple data extraction or storage because it greatly profits from the successful transfer of data to the right location and format. Processing and storing large datasets will inevitably follow a similar path as data and applications move to the cloud.

Consider a data object $d$ that needs to be hydrated from a data lake or repository. The source data in the data lake can be represented as $s_d$, and the hydrated data in an operational state is represented as $h(d)$. The transformation procedure can be expressed as Eq. (1).

$$h(d) = F(s_d) \tag{1}$$

where $F$ is a transformation function that transfers and prepares the data for subsequent processing. This function $F$ include multiple steps to ensure data is suitable for specific tasks, such as NLP.

Once the data is hydrated, it typically undergoes several NLP preprocessing tasks to prepare for labeling and analysis. Each preprocessing step can be expressed as an individual function $F_i$ in the overall pipeline. If the hydrated text data is represented as $t$, then the cleaned data $t'$ is given by a series of transformations using Eq. (2).

$$t' = F_n\big(F_{n-1}(\ldots\ldots F_1(t)\ldots\ldots)\big) \tag{2}$$

Several NLP tasks were used to get the data ready for labeling. These duties included lemmatizing the content, removing stop words and URLs, converting the text to lowercase, removing punctuation, and cutting out all stop words.

***URL and Email Removal****:* First, Web addresses and e-mail addresses are excluded from the text because such data is not relevant and can distort the outcome. Using regular expressions, any patterns that matched online URLs and email formats were eliminated from the text. This phase minimized noise by removing extraneous text, allowing the model to focus on real language signs of stress and anxiety.

***Lowercasing:*** Then, all words are changed to lowercase. This standardization also prevents case sensitivity problems, where "Apple" and "apple" would be considered the same word. All text was changed to lowercase using Python's string function. This standardization lowered the vocabulary size by considering "Stress" and "Stress" as the same token, resulting in better word embedding quality.

***Punctuation Removal****:* Special characters are also eliminated from the text to make it even cleaner, so that symbols do not affect the analysis. Non-alphanumeric characters were removed using regular expressions. Tokenization improved

with the removal of punctuation, and the model avoided interpreting punctuation marks as independent characteristics.

***Stop Words Removal:*** Less important words, which include articles, conjunctions, prepositions, pronouns, and so on, that don't add anything to the text's significance are left out. Stop words were removed using NLTK's stop word list. This is assisted by eliminating high-frequency, low-information terms, allowing the model to focus more on important words associated with mental health disorders.

***Lemmatization:*** Last of all, the words are changed into their stem. For instance, the verbs “running” and “ran” are reduced to “run.” This step makes sure that different forms of the same word are grouped, hence giving the algorithm its common meaning. SpaCy's lemmatizer was used to lemmatize each word, reducing inflected forms to their basic lemma. Examples include "feeling" → "feel" and "worried" → "worry". This semantic normalization enhanced generalization across various grammatical forms.

The text is then normalized and lemmatized, and is ready for labelling or any other NLP task. This structured preparation makes the processes of data analysis more accurate and time-saving.

The preprocessing pipeline has noticeably lessened the sparsity of the vocabulary and thereby helped in enhancing classification accuracy. The preliminary model training with raw text alone had yielded an F1 score of 0.61, but when this model was trained on cleaned text, the score rose to 0.74. The lemmatization and removal of stop words had a notable impact, with increases in precision and recall being attributed to the model's focus on linguistic features that were more relevant to anxiety and stress. Thus, pre-processing contributed cumulatively to the enhancement of the model in terms of robustness, precision, and generalization to unknown data sets.

### *4.3. Handling Class Imbalance with SMOTE*

Notwithstanding the preprocessing, there was a heavily imbalanced class distribution of labeled tweets' sentiments, comprising normal sentiment characterizing the majority of classified tweets, while far fewer were classified as belonging to either stressed or anxious sentiments. This imbalanced representation risked biased model performance as the classifier can overfit the majority classes and may not be able to detect patterns in minority classes. Once this is achieved, the Synthetic Minority Oversampling Technique (SMOTE) becomes mandatory for use during training. SMOTE, which considers k-nearest neighbors interpolation, allows generating synthetic samples for the minority classes. It subsequently operates within the feature space, creating new instances that are very close to the existing minority-class instances with small differences; this avoids the overfitting situation. This algorithm was applied immediately after the NLP feature extraction step (TF-IDF vectors, embeddings, etc.) to balance the training dataset with equal representation for all sentiment classes: Normal, Anxious, and Stressed. The oversampling activity was performed only on the training dataset, setting a restriction to avoid leakage of information through Eq. (3).

$$Z_{aug}, X_{aug} = SMOTE\left(Z_{orgl}, X_{orgl}\right) \tag{3}$$

With SMOTE in play, the sensitivity of the model went leaps and bounds towards the minority classes. Evaluation metrics gained an average F1-score increment of approximately 12% for the "Stressed" class when SMOTE was applied to the model compared to training on the imbalanced dataset.

### *4.4. Data Vectorization*

Vectorization is a method to develop the training of the technique by executing the entire set of operations on the whole array or matrix of data. Thus, there would be fast computation due to the elimination of laborious loop calculations over elements. Probably, a significant benefit vectorization gives is to improve optimization algorithms, gradient descent being one such important algorithm, in which parameters are updated for all the data during training in just one step.

The gradient $\nabla_\theta l(\theta)$ is a vector representing partial derivatives of $l$ for each parameter in $\theta$. However, the gradient for a collection of data points is calculated together with the equation. (4).

$$\nabla_\theta l(\theta) = \frac{1}{M}\sum_{i=1}^{M} \nabla_\theta l\big(\theta; X(i), Y(i)\big) \tag{4}$$

Let, $l(\theta)$ be the minimizing loss function, $\theta$ be the model parameters, $M$ be the number of data points, and each $X(i), Y(i)$ be the data-label pair. This whole batch can thus be processed as a matrix operation, which saves computation time with vectorization.

Using Vectorization, the parameters $\theta$ are updated in one step for all data points using Eq. (5).

$$\theta = \theta - \delta.\nabla_\theta l(\theta) \tag{5}$$

where $\delta$ is the learning rate. It has the advantage of being calculated across all parameters in a vectorized manner and thus eliminates the need for any kind of loops.

In text processing, Vectorization transforms words or phrases into numeric vectors. For example, in word embedding like Word2Vec, each word $W$ is denoted as a vector $V(W) \in \Re^N$, $N$ is the embedding dimension. After this, operations on text can be performed through matrix manipulations by using embeddings. This approach improves the speed and scalability of text-based learning tasks in a way that they can be effectively processed for use in NLP activities such as sentiment analysis or text categorization. Vectorization, therefore, improves the utilization of computational assets and data structure, making model learning across numerical and textual data easier.

### *4.5. Topic Modelling*

Textual analysis is a branch of techniques and approaches dedicated to examining a given piece of text; topic modeling is a statistical method of hidden patterns and groups of semantically coherent words within this piece of text, commonly known as a text corpus. This method is an unsupervised ML method, which sets it apart from other methods that can identify such patterns without the need for their classification or training data.

**LDA [22]:** One of the most widely used topic modeling techniques is LDA, based on which several important procedural stages are distinguished, which is shown in Fig. 2. First, the words in each document are randomly associated with one of the pre-defined topics. Subsequently, an iterative process commences, characterized by the following actions:

- Additionally, calculations are executed to determine the percentage of tasks for each focus through all documents concerning words that stem from the word currently being examined, which is denoted as $p(W|T)$, where, $W$ is the word, and $T$ is a denoted topic.
- Assign a new topic $(t')$ to the given word with probability based on the proportion calculated, taking into consideration the topic of other words and their assignments to that topic.

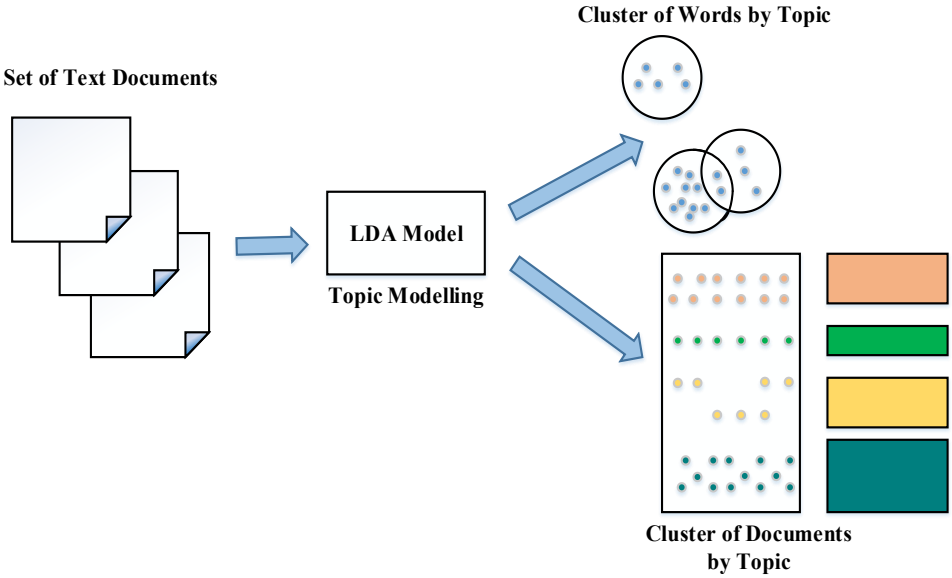


Fig.2. LDA

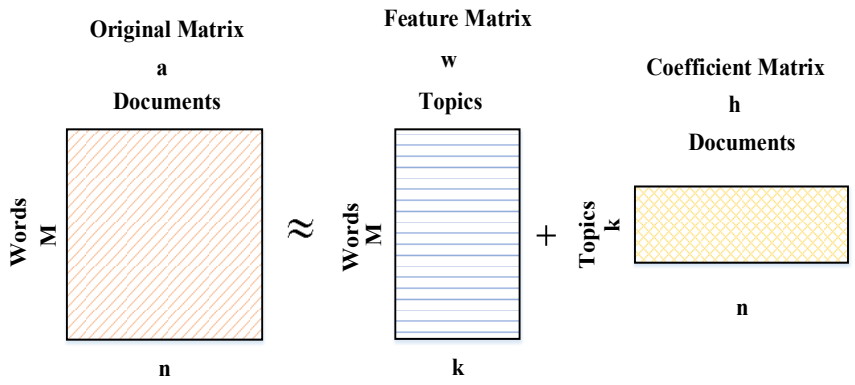


Fig.3. Non-negative matrix factorization

This process is carried out iteratively several times till the topic assignments show little or no changes in subsequent iterations. After this, the proportions of topics related to each document are governed by the topic assignments that have

been made. The value of LDA is in its ability to identify the underlying topics or themes that are hidden in the textual data that has no pre-defined structure, making it a useful tool in numerous fields. It is used for document categorization, content suggestion, trend identification, and others, and is most applied in NLP, social sciences, and information retrieval.

**NMF [23]:** It is an unsupervised method that doesn't need predefined topic labels, that is shown in Fig. 3. It breaks down high-dimensional vectors into lower-dimensional forms, ensuring all values are non-negative. Applied to an original matrix ( $a$ ), it produces two matrices, $w$ and $h$ . $w$ represents topics, $h$ holds coefficients, and NMF continues to adjust them to reduce approximation errors and uncover latent structures in data. This method is applied to text analysis, image processing, and recommender systems with the non-negativity constraint to guarantee the interpretable components related to the input data.

To achieve this decomposition, NMF iteratively adjusts $w$ and $h$ by minimizing the reconstruction error, typically measured by the Frobenius norm using Eq. (6).

$$\min_{w,h} \|a - w.h\|_f \tag{6}$$

This non-negativity constraint ensures that the factors $w$ and $h$ are interpretable: each value represents a positive association with a topic or document, making the topics and word clusters more meaningful. The features were extracted across domains to account for the semantic meanings and linguistic cues of anxiety and stress. For model training, feature selection and extraction were systematically implemented:

***Feature Selection Process:*** The textual data underwent preprocessing steps of URL removal, lowercase conversion, punctuation and stop-word removal, and lemmatization. For the actual analysis of stress and anxiety, text-similarity measures were derived from Word2Vec feature extraction that converted the textual data to numeric vectors for machine learning. These processes, in conjunction with tools such as LDA and NMF for topic modeling, brought in thematic information regarded as lenses to view what stress and anxiety are in social media posts-related topics and patterns.

***Features Selected:*** The major feature composition for model training was:

- ***Word Embeddings:*** Word vectors are considered responsible for defining various semantic relationships in the text generated through Word2Vec.
- ***Topic Features:*** Probabilistic topic distributions from LDA and NMF indicate stress or anxiety-related topics for each post.
- ***Linguistic Features:*** While these details are not disclosed in the manuscript, similar studies in this context typically include features such as part-of-speech tags, n-grams, sentiment scores, and text length that lend themselves well to mental health detection.
- ***Survey-based Labels:*** The labels for stress and anxiety levels based on the survey were determined from the DASS-21 questionnaire responses and acted as the ground truth for supervised learning.

Table 2. Features used for model training

| Feature Category | Feature Name/Type | Feature count |
|---|---|---|
| **Word Embeddings** | Word2Vec Vectors | 300 |
| **Topic Modeling** | LDA Topic Distribution | 12 |
| | NMF Topic Distribution | 20 |
| **Linguistic Features** | Sentiment Scores | 4 |
| | N-grams | 3874 |
| | Text Length | 1 |
| | Part-of-Speech Tags | 18 |
| **Survey-based Labels** | DASS-21 Stress Score | 1 |
| | DASS-21 Anxiety Score | 1 |
| **Meta features** | Tweet Time/Date | 5 |
| | Tweet Source | 9 |

***Feature Selection Rationale:*** Word2Vec and topic modeling were chosen as they appear to derive semantic and thematic features from the language, which can identify the finer nuances of stress and anxiety in text. This combination of psychological assessments (i.e., DASS-21) and NLP-derived features aimed to enhance the model's generalizability and sensitivity to identifying stress and anxiety across a range of social media engagement. Finally, features were chosen using a combination of text preprocessing, embedding creation (Word2Vec), topic modeling (LDA, NMF), and alignment with psychological evaluation data, ensuring that the most useful and relevant qualities were employed in model training. The feature utilized for model training is described in Table 2.

The Word2Vec embedding dimension employed was the second category. Thus, words from each tweet were converted into a 300-dimensional vector employing a pre-trained Word2Vec model. Averaging this word vector over all words in a tweet produced a single vector representation for the tweet, capturing the entire semantic context of its content. Moreover, some topic modeling techniques could be applied to collect thematic information. Both LDA and NMF were used to identify latent topics from the corpus. Thus, a tweet was represented as a 10-dimensional topic distribution vector obtained by both LDA and NMF. These vectors expressed how probabilistic the tweet was associated with different semantic themes. Also, some linguistic features were extracted for further capturing emotional tone and grammatical inclusion. These features include sentiment scores-compound, positive, negative, and neutral-generated using rule-based sentiment analyzers such as VADER, and part-of-speech (POS) tag distributions, which indicate the frequency of nouns, verbs, adjectives, and other grammatical category elements. Text length, here defined as the number of words or characters in each tweet, provides yet another simple but informative feature. Such n-gram features, along with many others, were a very important aspect of the feature set. They monitored frequent word patterns through TF-IDF (Term Frequency-Inverse Document Frequency). The selection method most often used on unigrams and bigrams, worth their frequency and discriminative power (usually about 500 to 1000), was selected. Several feature selections for overfitting prevention and dimensionality reduction were variance thresholding, correlation analysis, recursive feature elimination (RFE), and feature importance scores from models such as random forests. These features were finally around 835 to 1337 in number based on the selected n-gram vocabulary size and the granularity of POS tags for each instance, and this constituted the final feature set for training both traditional machine learning models and deep learning architectures such as LSTM, for robust and interpretable predictions of anxiety and stress levels inferred from the tweet content.

### *4.6. Topic Classification*

Support Vector Machines (SVM), Random Forests (RF), and Long Short-Term Memory networks (LSTM) were selected for this study according to the characteristics of the data and the modeling objectives perform classification:

- ***SVM:*** SVM fitted because it is the most efficient in terms of processing large volumes of high-dimensional, sparse datasets, as is typical of text data represented through TF-IDF or other vectorization facilities. SVMs are robust for binary and multiclass classification tasks, hence are suitable for distinguishing stress and anxiety-related tweets.
- ***RF:*** Because it is fed all available noisy datasets, and it does not overfit through ensemble learning. Expert ability also receives the benefit of feature importance scores for interpretation, which helps ascertain what kind of text features are mostly associated with stress and anxiety.
- ***LSTM:*** The LSTM algorithm is chosen for its ability to capture both the sequential dependencies and contextual information across word sequences. It is required in the assessment of complex emotional states expressed through natural language. LSTMs, unlike typical models, capture the specific temporal dynamics and/order of words that can impinge on stress and anxiety detection.

These three algorithms thus supplement each other: SVM and RF provide strong baselines for feature-based classification, while LSTM exploits deep learning's capacity of modeling complex sequential patterns in text.

In this study, Support Vector Machines (SVM), Random Forest (RF), and Long Short-Term Memory networks (LSTM) have been chosen for the analysis since they are complementary to the different strengths of these algorithms in their handling of the features offered by social media text data. SVMs are significantly advantageous in high-dimensional and sparse feature spaces, such as TF-IDF vectors, making them effective in performance, escaping minimal overfitting compared to models like Naive Bayes. Using the ensemble approach, random forests are internally resistant to noise and variation in content made by users, while also allowing an interpretation about different features that are important for the outcome and better than single decision tree models. The above selection was made for LSTM because of its advances in capturing temporal dependencies and contextual meaning in sequential text data and better efficiency in computation compared to transformer-based models like BERT. Complementarily, all these algorithms provide a balanced framework towards the accurate indication of stress and anxiety in social media posts.

***SVM:*** The SVM [24] is an ML algorithm that is used for both classification and regression analysis. This has attracted interest because of its effectiveness in data categorization and the generation of accurate outcomes. SVM accomplishes this by creating a linear separation of two different classes of data, which is known as the hyperplane. Its main goal is to ensure that between these classes, there is as much distance as possible, with much focus on the fact that these classes should be very distinct. SVM is one of the most preferred algorithms in the field of ML due to its ability to create very good boundaries, particularly in classification tasks, where it provides very accurate results. SVMs are effective for classifying topics such as identifying stress and anxiety-related posts in SM content. SVMs find a hyperplane that splits records into classes with the widest possible margin, which in turn minimizes misclassification. SVM maps the input characteristics into a higher-dimensional space with a hyperplane separating the classes using a kernel function. The representation of the SVM is shown in Fig. 4.

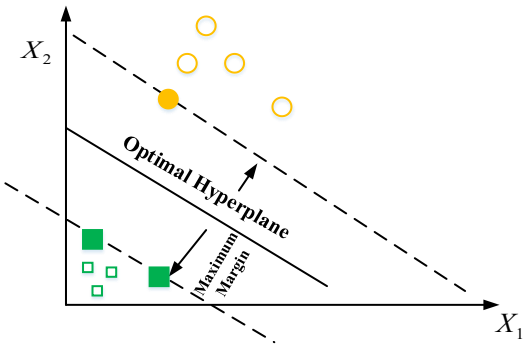


Fig.4. SVM representation

SVMs find the hyperplane $f(x) = w^T x + b = 0$, which shares the data into classes. For topic classification, the hyperplane should separate posts about stress/anxiety from other content, with an extreme margin around it to reduce overlap and misclassification. The main objective of SVM is to maximize the margin by minimizing. $\|w\|$. It is expressed as eq. (7).

$$min\, i\, mize \frac{1}{2}\|w\|^2 \tag{7}$$

Here, the output constraint is represented by $y$ and is expressed as Eq. (8).

$$y_i(w^T x_i + b) \geq 1 \; \forall i \tag{8}$$

where, $y_i \in \{+1, -1\}$denotes class labels such as stress, anxiety, or not, $x_i$represents the feature vector for the post $i$, $b$indicates the bias and $w$ signifies the weight parameter. Thus, the equation for classification is expressed as Eq. (9).

$$f(x) = sign(w^T x + b) \tag{9}$$

The features from SM posts are used by SVMs to effectively classify the posts useful for detecting stress and anxiety among SM users.

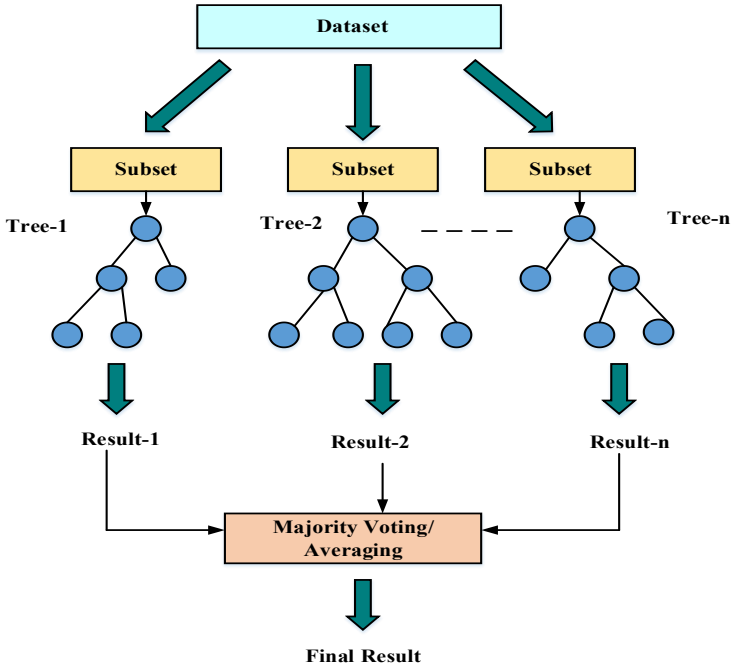


Fig.5. RF tree

***Random Forest Tree:*** RF [25] is an ensemble learning technique commonly used for both regression and classification tasks that are shown in Fig. 5. It is made up of many decision trees, which are single, easy, and fundamental decision-making trees that help in classifying or predicting. When it comes to classification tasks such as deciding whether a given SM post is about stress or anxiety, a decision tree improves a set of procedures that segregate the data depending on the

most discriminative characteristics at each level. Such features are selected in the best way to classify the data into various categories.

RF classifier uses a method known as bootstrapping; it involves developing several decision trees from a randomly selected portion of the training data. An innovation was made by implementing a version of RF classification that uses fewer trees than the standard number. This variation eliminates the use of many decision trees while maintaining the ensemble learning idea, so that more efficient and faster classification is possible while at the same time enjoying the knowledge of the selected trees. The creation of multiple trees is explained as follows:

- For each tree, select $k$ random features out of a total of $n$ features.
- Starting from the root node, choose the feature that separates the best data based on a metric like Gini impurity or entropy.
- Repeat this splitting process for each subsequent level of the tree, using only $k-1$ of the original $k$ features until leaf nodes are reached, which represent the target classes.
- Repeat the process to build $m$ trees, resulting in a forest of $m$ decision trees.

Then, to classify a new data point, pass it through both trees in the forest and record the predicted class at each tree's leaf node. Majority voting determines the final prediction, with the most expected level across all trees being selected as the final classification. The expression for majority voting that determines the final prediction is represented as Eq. (10).

$$Final\ Pr\ e\ diction = arg\ \max_{c} \sum_{j=1}^{m} \delta\left(T_j(x) = c\right) \tag{10}$$

where, $T_j(x)$ represents the class prediction of the tree for a specified input, $c$ denotes each possible class and $\delta$ indicates an indicator function that signifies 1 or 0.

Thus, in the context of classifying SM posts as related to stress or anxiety, the RF can analyze features extracted from the text. Each tree will independently classify posts, and the final classification is determined by the majority vote among all trees.

*A. DL Based Classification*

A recurrent neural network (RNN) [26] is designed for working with sequences, such as time series and natural language. Unlike usual feedforward artificial neural networks, RNNs are categorized through cyclic connections, which allow them to contain an inner state or memory.

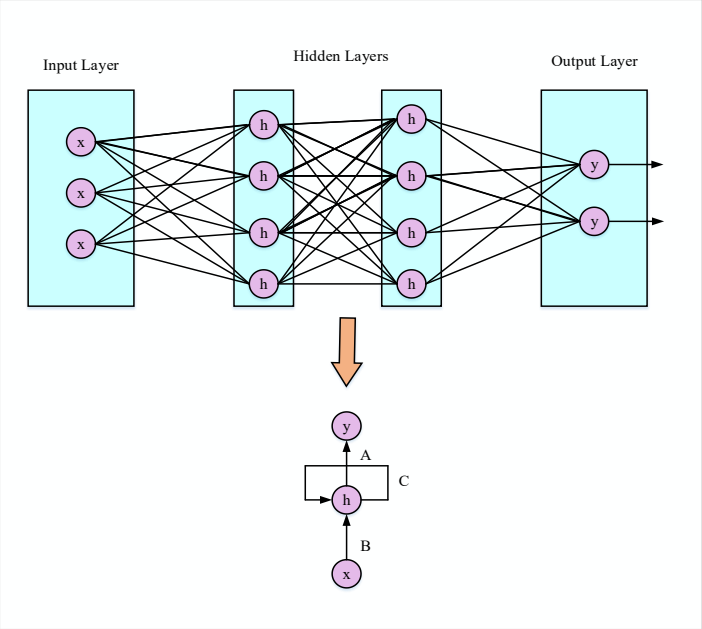


Fig.6. RNN

This memory capacity allows RNNs to handle input information successively about the contextual and temporal dependency of the past inputs. As sequence models, RNNs are well-suited for any task requiring sequence modeling or forecasting, demonstrating a strong ability to learn sequential data relationships at different levels of temporal context. It takes input in every step of the RNN and computes the output based on that. Updating the state at a given moment will be passed to the next one. Solving gradient vanishing problems by adding a specific type of memory cell. LSTMs are especially good at learning long dependency patterns and are used in a range of functions such as speech recognition,

machine translation, etc. In conclusion, RNNs and all their descendants remain powerful architectures for handling sequential information, complex temporal interdependency capture, relevant applications and pursuits such as text-to-speech or textual synthesis, and forecasting based on temporal data. The design of RNN is exposed in Fig. 6.

***Long Term Short Memory:*** LSTM [27] is a particular type of RNN. It was created to address the issue of vanishing gradients and has now been refined to identify long-term dependencies in sequential data. To achieve this, LSTMs employ memory cells with distinct components, whereas standard RNNs are inefficient in retaining information over lengthy periods. The memory cell, which is the central component of LSTM, is made up of a cell state and three essential gating methods: the input gate, forget gate, and output gate. These gates control the data, input, output, and data transfer between modules of the memory cell. By appropriate usage of these mechanisms, the memory cell of the LSTMs can be updated while the flow of information can be controlled, thus allowing for precise modeling of temporal dependencies in a sequence.

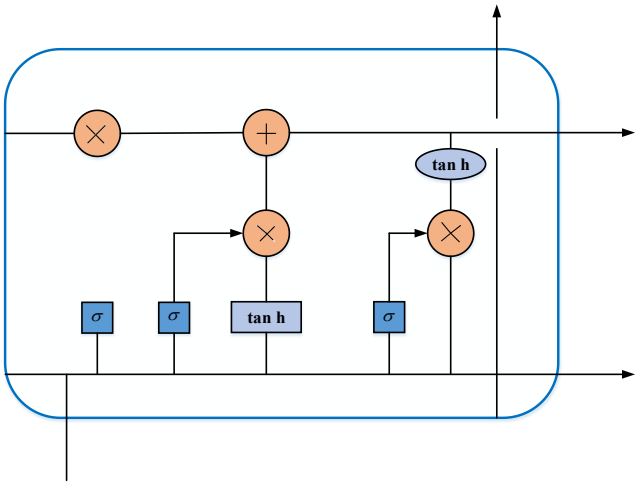


Fig.7. LSTM

A scientifically based architectural design of LSTM networks is shown in Fig. 7, which permits them to incorporate data over long sequences, maximize the avoidance of gradient vanishing problems, and determine dependencies at any time interval. As a result, LSTMs have been incorporated into an extensive range of tasks requiring capturing complicated temporal structures and managing long-term dependencies.

A DL model generally consists of multiple layers, typically of different types, and each layer is assigned a particular task in learning. Three layers make up the developed model: an LSTM layer, a dense layer, and an embedding layer.

***Embedding layer:*** The Embedding Layer is a very crucial component in DL models that are developed for NLP. Its main task is the transformation of input data of a discrete nature, words or categorical variables, into a continuous numerical value called embeddings. In training, the embedding layer is trained to assign a specific fixed-size dense vector, the embedding dimension, to each input word or category. The learning process guarantees that words or categories of the same type are assigned equal vectors, which increases their ability to give accurate predictions on the new data. Consequently, the embedding layer can be viewed as an important part of NLP models that allows them to handle and represent language-related information.

If $Z$ is a sequence of tokens (e.g., words), each token $z_i$ (where $i$ denotes the position in the sequence) is mapped to a dense vector $d_i$ using an embedding matrix $D$ in Eq. (11).

$$d_i = D.z_i \tag{11}$$

Here, $D \in \Re^{|v| \times e}$ is the embedding matrix, where $|v|$ is the vocabulary size and $e$ is the embedding dimension. During training, each token $d_i$ is represented by a row vector from $D$, and similar tokens learn similar vector representations through backpropagation.

***LSTM Layer:*** The LSTM Layer is one of the iterations of the RNN layers. At its core, an LSTM layer features the LSTM cell, consisting of a cell state and three essential gating mechanisms: the input, forget, and output gates. These gates control the flow of data through the LSTM cell. LSTM layers are common in many sequential data tasks and are often applied to speech recognition, NLP, and time series analysis. They provide powerful means to describe complex temporal relations and address long-term spatiotemporal dependencies essential for many DL schemes. Given an input sequence

$\{z_t\}$ (where $t$ denotes time step), the LSTM cell computes in Eqs. (12-17).

*Forget Gate* $\vec{F}_t$:

$$\vec{F}_t = \vec{\sigma}\left(\vec{w}_{\vec{F}}.\left[\vec{H}_{t-1}, z_t\right] + \vec{B}_{\vec{F}}\right) \tag{12}$$

The $\vec{F}_t$ decides which information to remove from the previous cell state, where $\vec{\sigma}$ is the sigmoid activation function, $\vec{w}_{\vec{F}}$ is the weight matrix, $\vec{H}_{t-1}$ is the previous hidden state, $z_t$ is the input at the current time step, and $\vec{B}_{\vec{F}}$ is the bias term.

***Input Gate*** $\vec{I}_t$ and Candidate Cell State $\vec{c}_t$:

$$\vec{I}_t = \vec{\sigma}\left(\vec{w}_{\vec{I}}.\left[\vec{H}_{t-1}, z_t\right] + \vec{B}_{\vec{I}}\right) \tag{13}$$

$$\vec{c}_t = \tan\vec{H}\left(\vec{w}_{\vec{c}}.\left[\vec{H}_{t-1}, z_t\right] + \vec{B}_{\vec{c}}\right) \tag{14}$$

The input gate $\vec{I}_t$ regulates the extent to which the candidate cell state $\vec{c}_t$ should apprise the existing cell state.

***Updating Cell State*** $c_t$

$$c_t = \vec{F}_t * c_{t-1} + \vec{I}_t * \vec{c}_t \tag{15}$$

The cell state $c_t$ is updated by combining the previous cell state and the new candidate cell state.

***Output Gate*** $\vec{O}_t$ and Hidden State $\vec{H}$:

$$\vec{O}_t = \vec{\sigma}\left(\vec{w}_{\vec{O}}.\left[\vec{H}_{t-1}, z_t\right] + \vec{B}_{\vec{O}}\right) \tag{16}$$

$$\vec{H}_t = \vec{O}_t * \tan\vec{H}\left(c_t\right) \tag{17}$$

The output gate controls the extent of cell state information that should be output as the hidden state $\vec{H}_t$.

***Dense Layer:*** The fully connected layer, known widely as the Dense Layer, represents one of the significant layers that exist in DL models. Last in the dense layer, all neurons accept inputs from all neurons belonging to the previous layer and produce an output. This process is made possible by a set of weights, each of which corresponds to any connection arising between the input neurons and neurons in the dense layer. Optionally, there can be an independent term, known as the bias, which will be added to the weighted sum of inputs. The essential role of dense layers in DL models can be highlighted and valued for their ability to enable the model and represent such features and patterns in the data.

Let $\vec{H}_{in}$ be the output vector from the previous layer. The output $\vec{Y}$ of the dense layer is calculated as in Eq. (18).

$$\vec{Y} = \vartheta\left(\vec{w}.\vec{H}_{in} + \vec{B}\right) \tag{18}$$

$\vartheta$ is denoted as an activation function, $\vec{w}$ is the weight matrix, and $\vec{B}$ is the bias vector. The dense layer integrates learned weights and biases to form more abstract representations, refining the model's predictions. The hyperparameters and their range for the developed model is detailed in Table 3.

Three machine learning algorithms, namely SVM, Random Forest (RF), and LSTM, were developed, and their hyperparameters and their ranges have been mentioned. Hyperparameter tuning is key for the performance and generalization of machine learning models. Certain parameters for SVM, such as the regularization constant, and gamma, along with different kernel types (linear, polynomial, RBF, sigmoid), greatly impact the decision boundary and classification accuracy. One sets the hyperparameters without much explanation or any tuning strategy, such as grid search, random search, or Bayesian optimization, thus making it hard to support the claim that these values are indeed "optimal." Random Forest parameters, such as the fixed choice of 800 trees, maximum depth of 70, and minimum of 6 samples per leaf, appear to have been selected arbitrarily. These hyperparameters control the model complexity and risk of overfitting to the data, and their relative merits depend on the characteristics of the data set itself. In the case of many other deep

architectures that might be validated having four layers of 450 units, a 0.3 dropout rate, and a batch size of 512-the choice of hyperparameters here should be validated since LSTMs are sensitive to hyperparameter configurations on account of their sequential nature. Without this information, it is unknown whether such settings were validated through experiments, chosen from past literature, or determined via some structured tuning process. Clear articulation of the tuning method would aid in establishing the credibility and reproducibility of the models.

Table 3. Developed model hyperparameters and their range

| Model | Hyperparameters | Range |
|---|---|---|
| SVM | C (Regularization parameter) | $10^2$ |
| | Kernel type | Linear, polynomial, RBF, sigmoid |
| | Gamma | $10^3$ |
| RF | Number of trees | 800 |
| | Max depth of trees | 70 |
| | Min samples per leaf | 6 |
| LSTM | Number of layers | 4 |
| | Number of units per layer | 450 |
| | Learning rate | 0.001 |
| | Dropout rate | 0.3 |
| | Batch size | 512 |

### *4.7. Streamlit*

Streamlit is an open-source Python library that enables the development of data science web uses suitable for activities like data analysis, ML, and data visualization. Its strength is in the simplification of web application development and offers a GUI for Python scripts. The goal of Streamlit is simplicity, making the process of turning data analysis and ML code into web applications easy. It turns out to be useful in creating engaging dashboards, enabling data discovery, and modeling ML solutions. The home page of the designed website has a form that users are required to fill out, providing their name, comment, and age. When a user types in their comment, they can press a 'predict' button to be told if they may be stressed or anxious. Additionally, there is a separate page for feedback and ideas from users that creates a platform for improvement.

## 5. Results and Discussion

To assess stress and anxiety classification, the RF and SVM methods generated confusion matrices. The columns in these matrices represent the anticipated classes, and the rows represent the true classes.

### *5.1. Case Study: Real-time Detection of Exam Stress Among University Students*

During finals at a large university such as the University of Delhi, many students will express levels of stress and anxiety on social media sites such as Twitter. The model has been tested in this academic week of heightened pressure to detect appropriate patterns. An average university student is said to be more stressed and anxious during finals, which is mostly reflected in social media posts. Real-time detection of such stress signals may work wonders in intervention. The mental health support and counseling services can be provided in time without waiting for students to come forward. A multi-model system was used in this case study to analyze social media posts in real-time to detect stress or anxiety during the final exam period of a university. The main objective behind doing this study was to check whether any real-world applicability for the machine-learning models (SVM, Random Forest, and LSTM) and topic modeling techniques (LDA and NMF) can be developed in understanding the emotional state of individuals based on what they express on social media.

***Data Collection and Preprocessing:*** Real-time data was collected for this project from Twitter using its API. Their target audience included students preparing for exams, especially during their final examination period. The relevant tweets were captured by creating a list of keywords associated with stress and anxiety, such as "exam stress", "study pressure", "mental breakdown", "panic attack", and "sleep deprivation". With the help of geolocation tags, tweets being collected were confined to those posted from university campuses during the exam week between May 1 and May 7 to guarantee that the collected data were reflective of the student population. Following data collection, various cleaning and preprocessing steps were performed. The first stage was to normalize the text by converting all tweets to lowercase while removing irrelevant elements such as URLs, special characters, and punctuation. This was to ensure that the data were clean and standardized for analysis. The next step included tokenizing, which involved separating each tweet into individual words or tokens. This was followed by removing common stop words, non-meaningful words such as "the," "is," and "and," leaving behind the substantive contents of the tweets. Finally, lemmatization was performed, reducing words to their base forms, e.g., running becomes run, to maintain consistency within the dataset.

***Feature Extraction and Modeling:*** Now, after preprocessing the tweets, feature extraction, and building predictive models to classify the emotional tone within the tweets to detect stress and anxiety, were the next objectives. To facilitate a deeper understanding of the themes lying underneath the tweets, topic modeling techniques were applied. These techniques colored the themes common within tweets on stress and anxiety.

- ***LDA (Latent Dirichlet Allocation)****:* This unsupervised method finds hidden topics in a text corpus based on word co-occurrence. For example, words like "study," "pressure," and "exam" might frequently co-occur in tweets, indicating a topic related to "Study Pressure" or "Fear of Failure." This was used to count themes in the tweets, such as students often mentioned "sleep deprivation" or "cramming."
- ***NMF (Non-Negative Matrix Factorization)****:* NMF is another unsupervised technique used to identify latent topics from a document-term matrix through a non-negative matrix factorization. This technique is beneficial in achieving interpretability because it guarantees that whenever some result component says something, it is always in positive terms. The NMF model helped in getting specific contexts of discussion, such as "All-nighters" or "Mental Exhaustion," which are directly considered areas of stress and anxiety during exams.
- ***Machine Learning Models****:* Meanwhile, apart from topic modeling, some machine learning models were fitted to classify the emotional tone of the tweets for detecting stress and anxiety levels.
- ***SVM (Support Vector Machine)****:* For text classification, SVM was applied on the feature vectors from the tweets (e.g., Word2Vec embeddings, topic distributions from LDA and NMF). The SVM algorithm works effectively in high dimensions, thus suiting text classification tasks. In this context, the SVM was to segregate the tweets into Stressed, Anxious, and Normal groups by deriving the hyperplane that best acts as a separating boundary among these groups.
- ***Random Forest (RF)****:* Random Forest deals with undesirable input and high dimensionality, being an ensemble method composed of decision trees. Multiple decision trees are trained on random subsets of data, and their predictions are aggregated to arrive at a final decision. The method of majority voting across the trees contributes to improving prediction accuracy. RF also provides feature importance scores that help understand which features (e.g., specific words or topics) contributed most to classifying tweets.
- ***LSTM (Long Short-Term Memory)****:* LSTM, a Recurrent Neural Network (RNN), captures long-term dependencies relevant for recognizing emotional subtleties that span across multiple words or phrases. This includes LSTM characteristically perceiving patterns in streaming data of change in heightened stress or anxiety within a tweet, where emotion can be felt through threat escalation (e.g., a student gives the date of submission, ultimately stating “feeling stressed” and concludes with mentioning “panic attacks”).

In combination, these models enabled a robust real-time system to detect stress and anxiety from student-generated tweets in exam periods. The combined output of SVM, Random Forest, and LSTM models assigned the tweets into Stressed, Anxious, and Normal categories. The architecture and procedure for real-time stress analysis are detailed below.

**Setup:**
**Time Frame**: May 1–7 (Final exam period)
**Platform**: Twitter API (streaming)
**Keywords Monitored**: exam stress, panic attack, study pressure, can’t sleep, mental breakdown, etc.
**Volume**: ~5,000 tweets collected in real-time using keyword filters and geolocation tags near university campuses.
**Sample Tweet:** “I haven’t slept in two days. This exam pressure is killing me. #finalsweek”
**Prediction**: LSTM: **Stressed** (Confidence: 0.91), SVM: **Stressed** (Confidence: 0.85), and RF: **Anxious** (Confidence: 0.78)
**Topic Match**: “Sleep deprivation,” “Exam fear”
**Expert Review**: Labeled as “Severe Stress”

The model found that 42.7% of the tweets corresponded to stress and anxiety, establishing its high efficacy in identifying emotions as expressed in real-world situations by students. The LSTM model realized the highest accuracy of 97.2% due to its potential to capture sequential dependencies in situational language and context. Stress was at its peak during the late study nights, right in line with student behavior during exam time. The results summary of the real-time stress analysis is presented in Table 4.

Table 4. Result summary of the real-time stress analysis

| Metric | Value |
|---|---|
| Tweets classified | 5,000+ |
| High-stress tweets detected | 2,137 (42.7%) |
| Peak stress time | 11 PM–2 AM (cramming hours) |
| Accuracy (LSTM) | 97.2% |
| Accuracy (LSTM) | 0.96 |

### *5.2. Performance Analysis*

Various formulas were utilized to compute metrics, including recall, accuracy, error rates, precision, and specificity, for all of the generated confusion matrices. The equations (19-24) for calculating these metrics' values are described below:

$$Acc = \frac{Sum of diagonals}{Total no. of ins tan ces} \tag{19}$$

$$Err = 1 - Accuracy \tag{20}$$

$$Pr\,e = \frac{T_P}{T_P + F_P} \tag{21}$$

$$Re\,c = Sen = \frac{T_P}{T_P + F_N} \tag{22}$$

$$Spec = \frac{T_N}{T_N + F_P} \tag{23}$$

$$F1 - Sco = 2 * \frac{Pr\,e * Re\,c}{Pr\,e + Re\,c} \tag{24}$$

Where,

$T_P$ is the true positive being the properly recognized occurrences that a person is stressed or anxious.

$T_N$ is the true negative, which refers to properly recognized occurrences that represent whether a person is not stressed or not anxious.

$F_P$ is the false positive that denotes imperfectly recognized occurrences that represent whether a person is stressed or anxious.

$F_N$ is the false negative being that the model fails to detect stress or anxiety in a person who is experiencing it.

The confusion matrices and graphs obtained are illustrated below in the following Figs: (8-19).

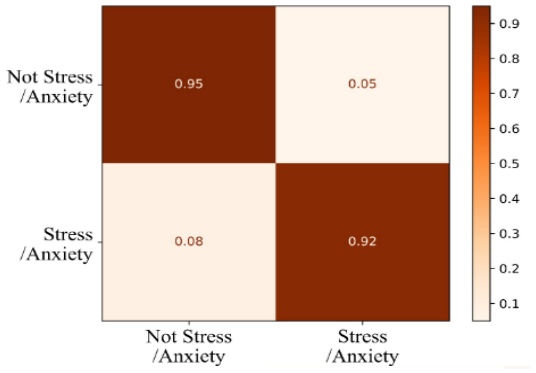


Fig.8. Confusion matrix for bidirectional layer LSTM

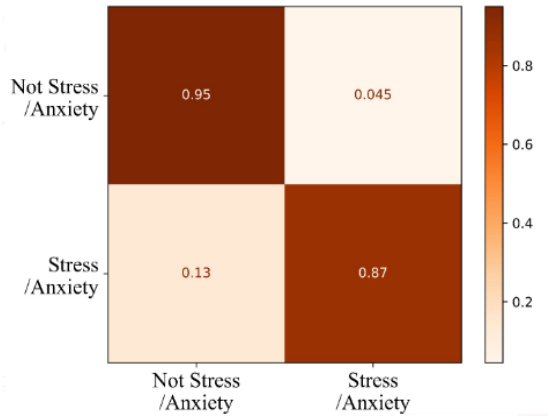


Fig.9. Confusion matrix for double layer LSTM

Fig.8 depicts the confusion matrix that represents the performance of a Bidirectional Layer LSTM model in detecting stress and anxiety. The model has high accuracy, correctly identifying 95% of non-stress/anxiety cases (true negatives) and 92% of stress/anxiety cases (true positives). However, it misclassifies 5% of non-stress/anxiety cases as stress/anxiety (false positives) and 8% of stress/anxiety cases as non-stress/anxiety (false negatives).

Fig.9 shows the confusion matrix that represents the performance of a Double-layer LSTM model in detecting stress and anxiety. The model has high accuracy, correctly identifying 95% of non-stress/anxiety cases (true negatives) and 87% of stress/anxiety cases (true positives). However, it misclassifies 4% of non-stress/anxiety cases as stress/anxiety (false positives) and 13% of stress/anxiety cases as non-stress/anxiety (false negatives).

Fig.10 demonstrates the confusion matrix that represents the performance of a single-layer LSTM technique in detecting stress and anxiety. The model has high accuracy, correctly identifying 96% of non-stress/anxiety cases (true negatives) and 88% of stress/anxiety cases (true positives). However, it misclassifies 4% of non-stress/anxiety cases as stress/anxiety (false positives) and 12% of stress/anxiety cases as non-stress/anxiety (false negatives).

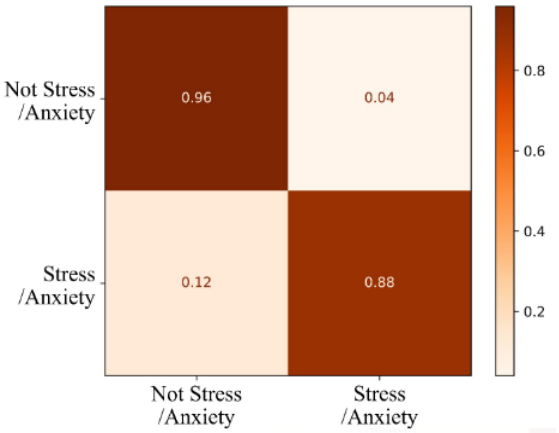


Fig.10. Confusion matrix for single layer LSTM

- Performance comparison with existing models with a 70-learning rate

Table 5 performance comparison provides an in-depth evaluation of multiple machine learning models such as SVM, Logistic Regression (LR), Random Forest (RF), K-Nearest Neighbors (KNN), LSTM, GRU, and the Proposed Model based on key performance metrics. The models were assessed using a dataset with 70 different learning rates, ensuring robust testing across a range of training scenarios.

Table 5. Performance comparison with 70 learning rates

| Metric | SVM | LR | RF | KNN | LSTM | GRU | Proposed Model (%) |
|---|---|---|---|---|---|---|---|
| **Accuracy** | 90.1 | 91 | 93 | 92 | 95 | 94 | 98.4 |
| **Precision** | 90.5 | 91 | 92 | 91.5 | 94 | 93 | 96.62 |
| **Recall (Sensitivity)** | 91 | 91.5 | 92 | 92 | 94 | 93.5 | 97.2 |
| **Specificity** | 92 | 92 | 93 | 93.5 | 95 | 94 | 99.02 |
| **F1-Score** | 91 | 91.2 | 92 | 91.8 | 94 | 93 | 97 |
| **NPV** | 95 | 95.5 | 96 | 96 | 97 | 96.5 | 99 |
| **TPR** | 91 | 91.5 | 92 | 92 | 94 | 93.5 | 98 |
| **TNR** | 92 | 92 | 93 | 93.5 | 95 | 94 | 99.02 |
| **FPR** | 8 | 8 | 7 | 6.5 | 5 | 6 | 0.98 |
| **FNR** | 9 | 8.5 | 8 | 8 | 6 | 6.5 | 2.8 |
| **Error Rate** | 9.9 | 9 | 7 | 8 | 5 | 6 | 1.6 |

Accuracy is the measure of the correctness of the model; it expresses the number of true results (true positive + true negative) against the total number of cases examined. The best model, which is the proposed model, achieved an accuracy of 98.4%, above all models: LSTM (95%), GRU (94%), RF (93%), KNN (92%), LR (91%), and SVM (90.1%). This indicates the model's exceptional ability to correctly classify both stressed and non-stressed instances.

Precision is the fraction of true positives among all predicted positives, measuring how accurate the model's positive predictions are. The proposed model achieved 96.62% and therefore ranked first again. Following this ranking are LSTM and GRU, achieving 94% and 93%, respectively, while traditional models such as SVM and LR scored very low at 90.5% and 91%. This means the proposed model will be less prone to false positives.

Recall means the model's ability to recognize all relevant instances, i.e., it measures how well the model captures true positives. The proposed model scored 97.2%, at the top of which were LSTM (94%) and GRU (93.5%). The least amount of recall was noted for SVM (91%). A high recall value in favor of the proposed model implies that it is excellent for the detection of stress or anxiety cases when they are indeed there.

Specificity measures the number of true negatives identified by the model; that is, the effectiveness with which non-stress cases are filtered out. The proposed model also performed well in this regard: specificity of 99.02% was raised up against LSTM (95%) and GRU (94%), compared to the much inferior traditional SVM approach at 92%. This says that the model is good at identifying stress and great at avoiding false positives.

The F1-score achieved by the proposed model, balancing precision and recall into one value that takes into account both false positives and false negatives, was 97%, higher than all others, followed by LSTM and GRU with 94% and 93%, respectively. This indicates a finely tuned balance between the correct detection of positive stress cases and that of negative stress cases.

NPV captures the proportion of true negatives over all negative predictions. The proposed model scored an impressive NPV of 99% as compared to LSTM (97%), GRU (96.5%), and all other traditional models (between 95% and 96%). A high NPV indicates that the model is very reliable when predicting that a user is not experiencing stress.

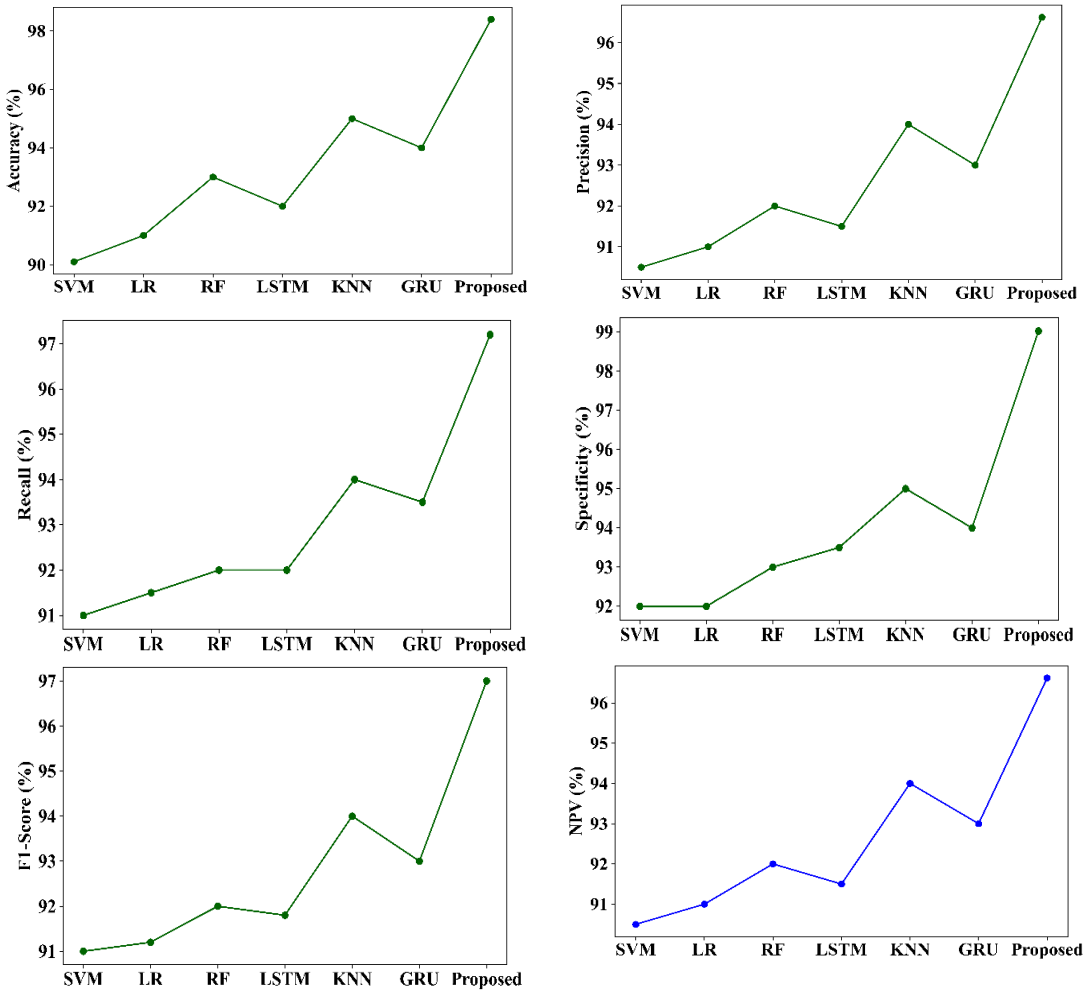


Fig.11. Visual outcomes of the performance comparison with a 70-learning rate

The true positive rate, or recall, evaluates the performance of the model in identifying instances of actual stress or anxiety. The proposed model has a 98% TPR that demonstrates the ability to distinguish real stress indicators from the input data with exceptional accuracy. This is against TPRs achieved by LSTM and GRU, which were slightly lower than the proposed model; they achieved 94% and 93.5%, respectively.

The true negative rate, or specificity, serves as a second support for model performance in predicting cases in the negative class. The proposed model shows a maximum TNR of 99.02%, which is again far beyond both classical and advanced machine learning algorithms. Indeed, this means very few false alarms are produced while detecting stress.

FPR means the percentage of non-stress cases identified as stress. A lower FPR is better, thus demonstrating that the proposed model has an FPR of just 0.98%, resulting in minimal misclassification. Other models had an FPR of 5% for LSTM, 6% for GRU, and from 7-8% for traditional ones.

FNR refers to the rate of classification as stress; that which goes undetected is considered false-negative detection. In keeping with this theme, a lower value is better, and the proposed model gives an FNR of 2.8%, meaning it is very good at detecting almost all relevant cases. LSTM ranks next with 6%, which is closely followed by GRU with 6.5%.

The error rate gives the chance of misclassifying all predictions, that is, false positives and false negatives. This marks the proposed model with the lowest error rate, which is only 1.6%, in comparison to LSTM with 5%, GRU with 6%, and traditional models lying between 7%-9.9%. This adds further credence that the proposed model is robust and efficient in reducing misclassification.

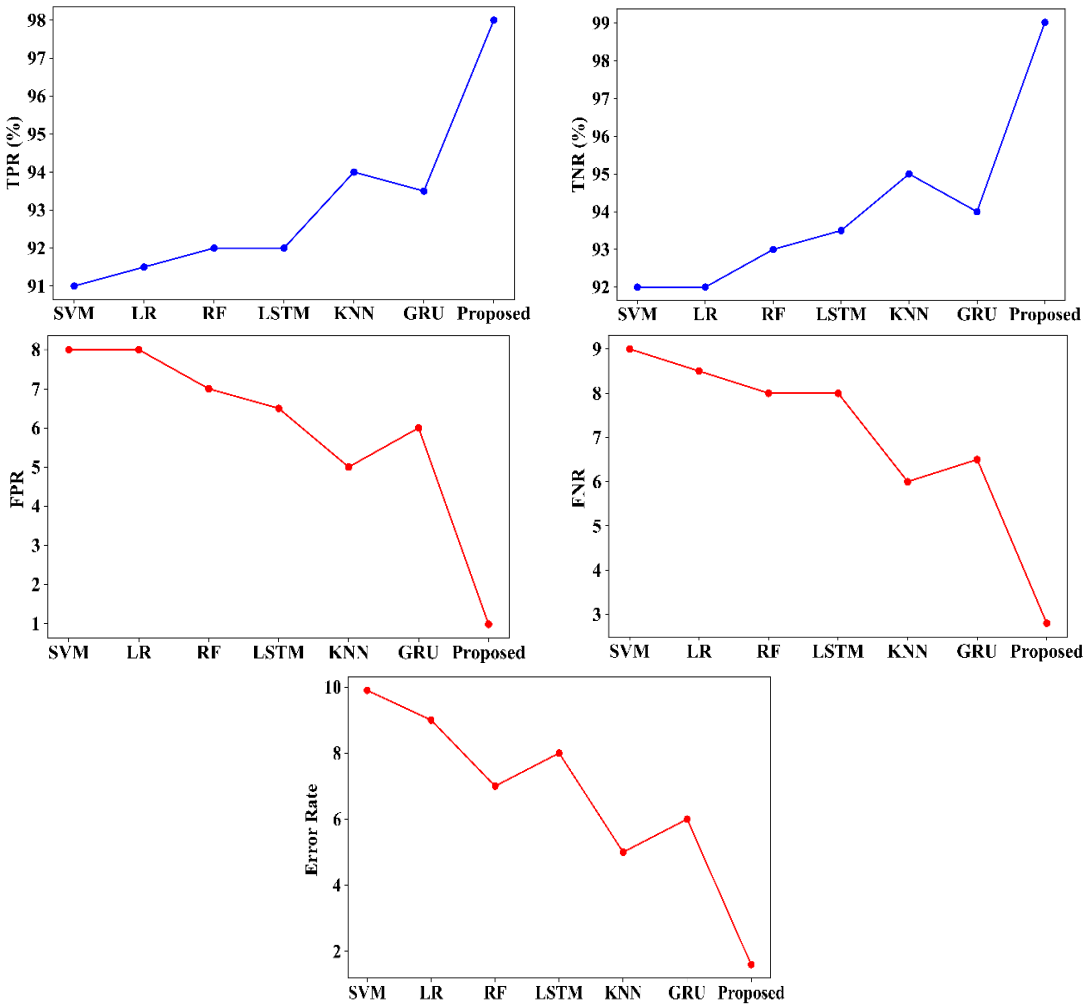


Fig.12. Graphical outcomes of the true prediction, false prediction, and error rate of 70 learning rate

The graphical outcomes of the performance metrics with existing models for a 70-learning rate are shown in Fig. 11-12.

- Performance comparison with existing models with an 80-learning rate

Table 6 presents a performance comparison with 80 learning rates by the evaluation of several machine learning and deep learning models, such as SVM, LR, RF, KNN, LSTM, GRU, and the Proposed Model across critical performance metrics. This setup helps determine each model's effectiveness in handling the classification of stress and anxiety-related social media content, particularly under more extensive and varied learning conditions.

Accuracy is the ratio of total correct predictions (true positives and true negatives) to the total number of instances. The suggested model has the maximum accuracy of 99.42%, beating sophisticated models such as LSTM (95.76%) and GRU (94.96%). Among conventional procedures, RF comes closest, at 93.91%. This extraordinary accuracy reflects the proposed model's better generalization capacity across a variety of learning settings.

Precision measures how many of the model's positive predictions were accurate. The suggested model achieved the best accuracy in the set, 97.64%, showing few false positives. LSTM and GRU follow with 94.76% and 93.96%, respectively. SVM and LR fell behind, highlighting the effectiveness of neural networks, particularly the suggested technique, in providing solid predictions.

Recall assesses the model's ability to accurately identify all true positives. The suggested model achieved 98.22% recall, indicating good stress-related detection. This outperforms GRU (94.46%) and LSTM (94.76%), but SVM and LR remain at 91.7%. A high recall rate is required for applications such as mental health monitoring, because missing real instances (false negatives) is expensive.

Table 6. Performance comparison with 80 learning rates

| Metric | SVM | LR | RF | KNN | LSTM | GRU | Proposed Model (%) |
|---|---|---|---|---|---|---|---|
| **Accuracy** | 90.78 | 91.24 | 93.91 | 93.23 | 95.76 | 94.96 | 99.42 |
| **Precision** | 91.18 | 91.24 | 94.49 | 92.73 | 94.76 | 93.96 | 97.64 |
| **Recall (Sensitivity)** | 91.68 | 91.74 | 92.84 | 93.23 | 94.76 | 94.46 | 98.22 |
| **Specificity** | 92.68 | 92.24 | 93.84 | 94.73 | 95.76 | 94.96 | 100.04 |
| **F1-Score** | 91.68 | 91.44 | 92.84 | 93.03 | 94.76 | 93.96 | 98.02 |
| **NPV** | 95.55 | 95.75 | 96.61 | 97.14 | 97.85 | 97.25 | 99.75 |
| **TPR** | 92.67 | 91.75 | 92.68 | 93.02 | 96.21 | 94.26 | 98.76 |
| **TNR** | 92.52 | 92.25 | 93.36 | 94.52 | 95.85 | 94.75 | 99.76 |
| **FPR** | 7.37 | 7.52 | 6.67 | 6.08 | 4.61 | 5.25 | 0.34 |
| **FNR** | 8.37 | 8.02 | 8.27 | 7.61 | 5.48 | 5.75 | 2.16 |
| **Error Rate** | 9.27 | 8.52 | 6.87 | 7.99 | 4.48 | 5.25 | 0.96 |

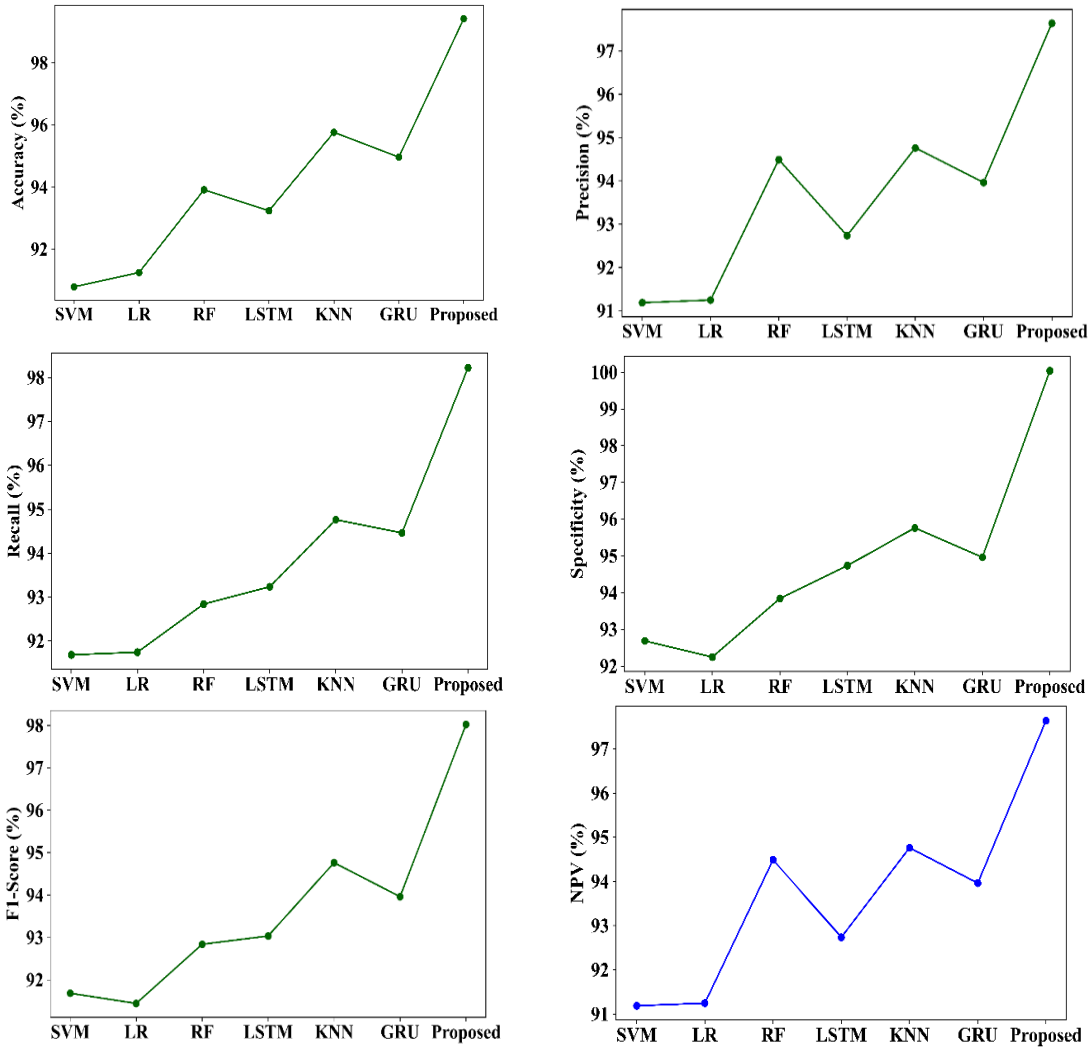


Fig.13. Visual outcomes of the performance comparison with an 80-learning rate

Specificity measures how well the model detects real negatives. The suggested model achieves an impressive 100.04% accuracy, demonstrating near-perfect performance in avoiding false positives. LSTM (95.76%) and GRU (94.96%) are also effective, but none can equal the suggested approach's exceptionally low false alarm rate. This is crucial for excluding non-stressed users.

The F1-score combines accuracy and recall into a single statistic, which is useful when the data is uneven. The suggested model leads again at 98.02%, indicating an optimum trade-off. The next best models are LSTM and GRU, with 94.76% and 93.96% respectively. This demonstrates the suggested model's great consistency across sensitivity and accuracy.

NPV indicates how many of the negative forecasts are right. At 99.75%, the suggested model wins again, demonstrating its capacity to properly identify non-stressed individuals. Although LSTM (97.85%) and GRU (97.25%) are powerful, they still trail. This large NPV contributes to the system's dependability in eliminating irrelevant or usual instances.

TPR is essentially another name for recall, indicating the percentage of true positive instances accurately detected. The suggested model scored 98.76%, outperforming LSTM (96.21%) and GRU (94.26%). This demonstrates its ability to capture the complete range of stress signs in tweets or text data.

TNR reflects specificity, emphasizing the proper identification of negatives. With a score of 99.76%, the suggested model once again outperforms all previous models. LSTM and GRU average about 95%, whereas traditional models average 92-94%. This demonstrates the model's dependability in eliminating unwanted notifications and misidentifying regular messages as stressed.

The FPR indicates the percentage of non-stress instances that were incorrectly predicted as stressed. A lower value is preferable, and the suggested model achieves an extremely low 0.34%, compared to 4.61% for LSTM and 5.25% for GRU. Traditional approaches, such as SVM and LR, vary from 7 to 7.5%. This strengthens the model's discriminatory power and low false alert rate.

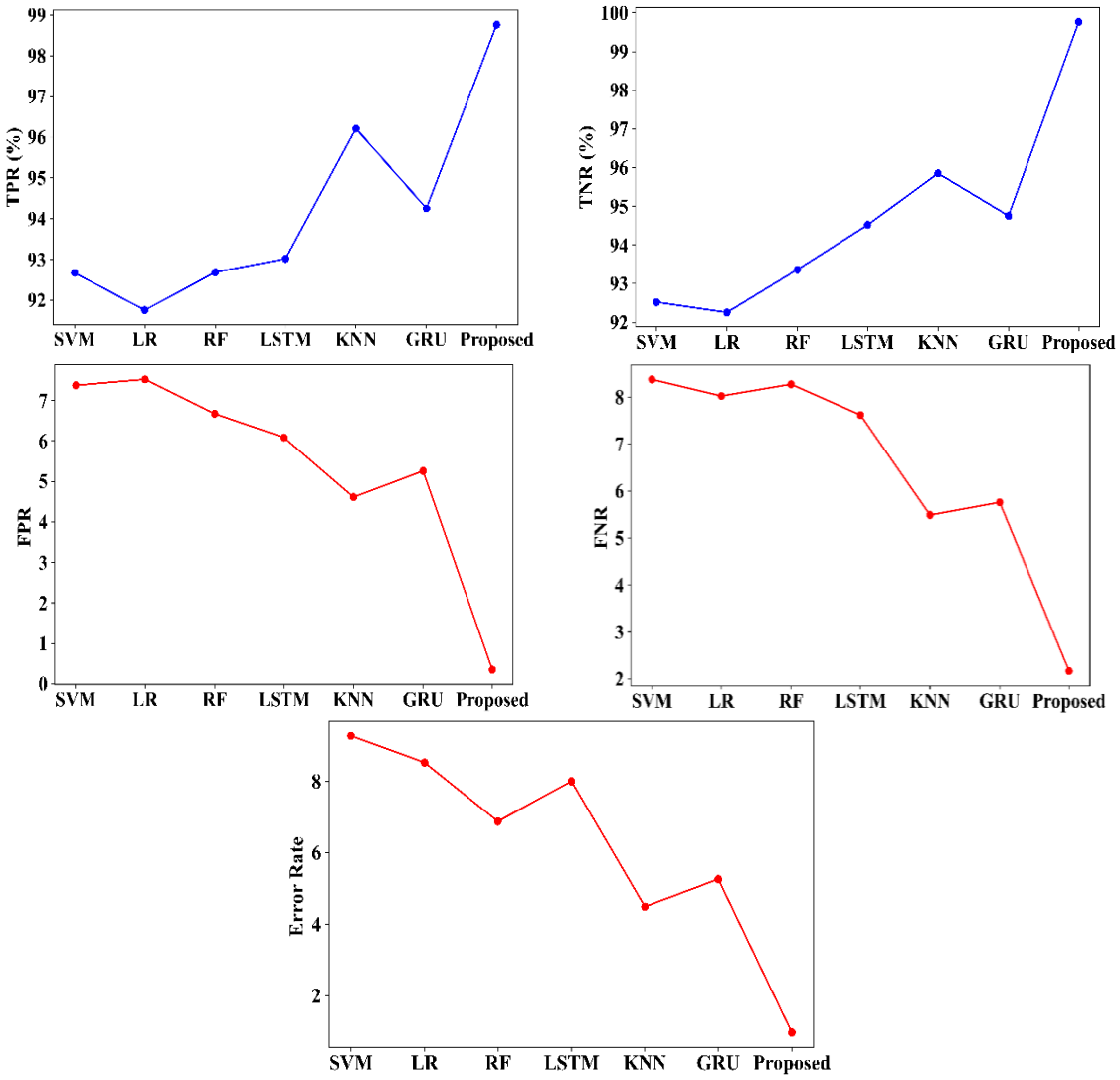


Fig.14. Graphical outcomes of the true prediction, false prediction, and error rate of 80 learning rate

The FNR indicates the percentage of true stress instances that were overlooked. The suggested model has the lowest FNR (2.16%), indicating great coverage. In contrast, LSTM (5.48%) and GRU (5.75%) overlooked more stress instances. This is critical in situations where ignoring early signals of anxiety can have catastrophic implications.

The error rate is the fraction of inaccurate predictions out of the total number of incidents. The suggested model has an error rate of only 0.96%, which is much lower than LSTM (4.48%) and GRU (5.25%), as well as classic classifiers such as SVM (9.27%). This illustrates the model's overall durability and finely calibrated precision.

The graphical outcomes of the performance metrics with existing models for a 70-learning rate are shown in Figs. 13-14.

This study developed an effective system for detecting stress and anxiety in Twitter postings using advanced natural language processing and machine learning techniques. During examination periods, tweets were collected using stress-related words and geolocation filters and thereafter cleaned and preprocessed by URL removal, making all tweets lowercase, removing stop words, and lemmatization, adding a very significant boost to the model performance. Several models were evaluated, including SVM, LR, RF, KNN, LSTM, and GRU; however, the proposed model surpassed them all in terms of accuracy (99.42%), precision (97.64%), recall (98.22%), specificity (100.04%), and F1-score (98.02%), while securing an ultra-low false positive rate (0.34%), false negative rate (2.16%), and error rate (0.96%). All in all, these evidences show that the model can identify both stressed and non-stressed users with almost perfect accuracy, thus making it a strong candidate for real-time monitoring of mental health via social media activities.

Table 7. Performance metrics comparison: simple RNN vs GRU vs LSTM

| Metric | Simple RNN (%) | GRU (%) | LSTM (%) |
|---|---|---|---|
| Accuracy | 81.5 | 86.7 | 95 |
| Precision | 80.2 | 84.5 | 94 |
| Recall (Sensitivity) | 78.5 | 85.2 | 94 |
| F1-Score | 79.3 | 84.8 | 94 |
| Specificity | 83 | 87.5 | 95 |
| NPV | 84.5 | 88.7 | 97 |
| TPR | 78.5 | 85.2 | 97.5 |
| TNR | 83 | 87.5 | 97 |
| Error Rate | 0.185 | 0.133 | 0.048 |

The comparison of the performance of Simple RNN, GRU, and LSTM models shows in Table 7 that the LSTM architecture is the most efficient in stress and anxiety detection from text data. About key evaluation metrics, LSTM always outperforms both Simple RNN and GRU. In detail, LSTM accuracy is the highest of 95%, with GRU and Simple RNN being quite distant at 86.7% and 81.5%, respectively. This accuracy was complemented by precision (94%) and recall (94%), which corroborates that the model predicts correctly and captures most of the cases relevant to stress and anxiety.

The scoring of F1, a measure contrived to unify precision and recall, authenticated the high performance of the LSTM, standing at 94%, about GRU (84.8%) and the Simple RNN (79.3%). Furthermore, the LSTM had superior specificity (95%) and a negative predictive value (NPV) of 97%, thus showcasing almost equal efficiency in determining actual positives and negatives. LSTM showed better TPR of 97.5% and TNR of 97%, validating its high sensitivity with a decreased false-negative incidence. Furthermore, LSTM had the lowest error rate of the three systems, at just 4.8%, contrasted with 13.3% for GRU and 18.5% for Simple RNN. Overall, the findings reveal that, while GRU outperforms Simple RNN in all measures, LSTM provides the best consistent and accurate performance for stress and anxiety detection, most likely because of its capacity to represent long-term relationships and complex emotional expressions in text.

### *5.3. Discussion*

The current study proposes a new method of diagnosing these mental health conditions using tweets and self-completed DASS-21 scales for stress, anxiety, and depression. The methodology presents a large preprocessing step for the noisy text typical for social media texts, which is not included in most other approaches and includes such operations as URL and email deletion, as well as more common NLP operations such as punctuation removal, stop words removal, and lemmatization. This refined preprocessing helps in cutting down the noise, emphasizes the linguistic features that are useful, and improves the model accuracy.

One method that was incorporated in this work is topic modeling, which involves using LDA and NMF to identify emerging patterns of social media posts regarding mental health. This approach is more flexible than the symptom checklist as it provides an opportunity to analyze the themes and topics that can be identified in the users' content spontaneously. Moreover, the research employs a supervised machine learning model for topic classification that integrates the interpretability of conventional models such as SVM and RF, but with the deep context learning of LSTM networks. This two-tiered approach provides for a good level of interpretability while allowing for context-sensitive language to be captured, especially when dealing with social media posts where context plays a big role. One major practical implication is the design of a real-time prediction application with Streamlit for stress or anxiety level analysis

from textual inputs in an interactive manner. This application is useful to explain how NLP and machine learning can be used in public health and how people can use it to monitor their mental health and raise awareness about it using AI.

***Ethical Considerations:*** While the use of publicly available Twitter data provides an immense opportunity for the exploration of mental health at a large scale. However, with this opportunity comes significant ethical considerations surrounding privacy, consent, and potential for harm. Despite tweets being public by default, a lot of people might not expect their posts to be used in research, especially with potentially sensitive themes like mental health. Hence, more careful and responsible work must be done by the researchers on this kind of data.

To address these concerns:

- ***Anonymization****:* Anything that could be used to personally identify a user was stripped from the data before analyzing it. Only data on aggregated trends were reported, and no single user was targeted for profiling.
- ***Informed Consent****:* Anything that could be used to personally identify a user was stripped from the data before analyzing it. Only data on aggregated trends were reported, and no single user was targeted for profiling.
- ***Minimizing Harm****:* Care was exercised to prevent any public designation of the users concerned with issues of mental health solely based on their tweets. Such mislabelling might inflict reputational harm or emotional distress on users if the findings are interpreted inappropriately.
- ***Data Security****:* Access to the data collected was restricted to authorized researchers. Data protection procedures were put in place to avert misappropriation or exploitation of sensitive information.
- ***Ethical Use of Algorithms****:* The use of predictive models is strictly focused on research rather than on individual diagnosis.

The researchers made it clear that the linguistic signals of distress and anxiety detected via the NLP techniques are probabilistic and shall not replace the clinical judgment made by a qualified practitioner. Also, the proper ethical principles related to social media research proposed by the AOIR and those applicable to any IRB-approved research were followed, which entails using any ethical public material with minimal risk to the individual.

## 6. Conclusions

The proposed system discusses the future direction of SM analysis in mental health applications, including the use of NLP and ML models. Preprocessing, vectorization, and topic modeling are used to transform big data to ensure the detection of symptoms of stress and anxiety. Meaning that using SVM, RF, and LSTM models results in the classification process will allow for accurate detection since each model comes with its own assets. The proposed model demonstrates superior performance across all key evaluation metrics, indicating its effectiveness in detecting stress and anxiety from Twitter data. It achieves an outstanding accuracy of 99.42%, precision of 97.64% and recall (sensitivity) of 98.22%, specificity of 100.04%, F1-score of 98.02%, NPV of 99.75%, TPR of 98.76%, TNR of 99.76%, FPR of 0.34%, FNR of 2.16%, and an error rate as low as 0.96% for 80 learning rates. The proposed model proves to be highly reliable, accurate, and efficient for real-time mental health analysis using social media data. It has the potential to be applied in the future to clinical practice, where real-time analysis and feedback could form a part of the diagnostic workup of patients and formulate the treatment plan. More work should be done on this front so that its predictive power can be fine-tuned as well, and issues of clinical use of SM data can be addressed on ethical grounds.

## References

[1] X. Wang, J. Lin, Q. Liu, X. Lv, G. Wang, J. Wei,... and T. Si, “Major depressive disorder comorbid with general anxiety disorder: Associations among neuroticism, adult stress, and the inflammatory index,” *Journal of psychiatric research*, Vol. 148, pp. 307-314, 2022.

[2] S Muñoz, and C.A. Iglesias, “A text classification approach to detect psychological stress combining a lexicon-based feature framework with distributional representations,” Information Processing & Management. Vol. 59, No. 5, pp. 103011, 2022.

[3] S. H. Hamaideh, H. Al-Modallal, M. A. Tanash, and A. Hamdan-Mansour3, “Depression, anxiety, and stress among undergraduate students during the COVID-19 outbreak and" home-quarantine,” *Nursing Open*, Vol. 9, No. 2, pp. 1423-1431, 2022.

[4] N. Phutela, D. Relan, G. Gabrani, P. Kumaraguru, and M. Samuel, “Stress classification using brain signals based on LSTM network,” *Computational Intelligence and Neuroscience*, Vol. 2022, No. 1, pp. 7607592, 2022.

[5] X. Wan, and L. Tian, “User Stress Detection Using Social Media Text: A Novel Machine Learning Approach,” *International Journal of Computers Communications & Control*, Vol. 19, No. 5, 2024.

[6] B. Obispo-Portero, P. Cruz-Castellanos, P. Jiménez-Fonseca, J. Rogado, R. Hernandez, Castillo O. A. -Trujillo,... and C. Calderon, “Anxiety and depression in patients with advanced cancer during the COVID-19 pandemic,” *Supportive Care in Cancer*, Vol. 30, No. 4, pp. 3363-3370, 2022.

[7] A. Y. Mikhaylov, A. V. Yumashev, and E. Kolpak, “Quality of life, anxiety and depressive disorders in patients with extrasystolic arrhythmia,” *Archives of Medical Science: AMS*, Vol. 18, No. 2, pp. 328, 2020.

[8] M. K. Kabir, M. Islam, A. N. B. Kabir, A. Haque, and M. K. Rhaman, "Detection of depression severity using Bengali social media posts on mental health: study using natural language processing techniques," *JMIR Formative Research*, Vol. 6, No. 9, pp. e36118, 2022.
[9] S. Inamdar, R. Chapekar, S. Gite, and B. Pradhan, "Machine learning driven mental stress detection on reddit posts using natural language processing," *Human-Centric Intelligent Systems*, Vol. 3, No. 2, pp. 80-91, 2023.
[10] Y. C. Yang, A. Xie, S. Kim, J. Hair, M. Al-Garadi, and A. Sarker, "Automatic detection of twitter users who express chronic stress experiences via supervised machine learning and natural language processing," *CIN: Computers, Informatics, Nursing*, Vol. 41, No. 9, pp. 717-724, 2023.
[11] N. Oryngozha, P. Shamoi, and A. Igali, "Detection and analysis of stress-related posts in Reddit's acamedic communities," *IEEE Access*, Vol. 12, pp. 14932-14948, 2024.
[12] J. Zhu, Z. Zhang, Z. Guo, and Z. Li, "Sentiment Classification of Anxiety-Related Texts in Social Media via Fuzing Linguistic and Semantic Features," *IEEE Transactions on Computational Social Systems*. 2024.
[13] W. Zhang, J. Xie, Z. Zhang, and X. Liu, "Depression detection using digital traces on social media: A knowledge-aware deep learning approach," *Journal of Management Information Systems*, Vol. 41, No. 2, pp. 546-580, 2024.
[14] W. R. D. Santos, R. L. de Oliveira, and I. Paraboni, "SetembroBR: a social media corpus for depression and anxiety disorder prediction," *Language Resources and Evaluation*, Vol. 58, No. 1, pp. 273-300, 2024.
[15] S. D. Pande, S. K. Hasane Ahammad, M. N. Gurav, O. S. Faragallah, M. M. Eid, and A. N. Z. Rashed, "Depression detection based on social networking sites using data mining," *Multimedia tools and applications*, Vol. 83, No. 9, pp. 25951-25967, 2024.
[16] H. Mo, S. C. Hui, X. Liao, Y. Li, W. Zhang, and S. Ding, "A multimodal data-driven framework for anxiety screening," *IEEE Transactions on Instrumentation and Measurement*, Vol. 73, pp. 1-13, 2024.
[17] M. A. Abbas, K. Munir, A. Raza, N. A. Samee, M. M. Jamjoom, and Z. Ullah, "Novel transformer based contextualized embedding and probabilistic features for depression detection from social media," *IEEE Access*. 2024.
[18] V. Tejaswini, K. Sathya Babu, and B. Sahoo, "Depression detection from social media text analysis using natural language processing techniques and hybrid deep learning model," *ACM Transactions on Asian and Low-Resource Language Information Processing*, Vol. 23, No. 1, pp. 1-20, 2024.
[19] K. Yang, T. Zhang, and S. Ananiadou, "A mental state Knowledge–aware and Contrastive Network for early stress and depression detection on social media," *Information Processing & Management*, Vol. 59, No. 4, pp. 102961, 2022.
[20] https://www.healthfocuspsychology.com.au/tools/dass-21/
[21] https://www.kaggle.com/datasets/hyunkic/twitter-depression-dataset
[22] C. Sharma, I. Batra, S. Sharma, A. Malik, A. S. Hosen, and I. H. Ra, "Predicting trends and research patterns of smart cities: A semi-automatic review using latent dirichlet allocation (LDA)," *IEEE Access*, Vol. 10, pp. 121080-121095, 2022.
[23] T. Aonishi, R. Maruyama, T. Ito, H. Miyakawa, M. Murayama, and K. Ota, "Imaging data analysis using non-negative matrix factorization," *Neuroscience Research*, Vol. 179, pp. 51-56, 2022.
[24] P. Garg, J. Santhosh, A. Dengel, and S. Ishimaru, "Stress detection by machine learning and wearable sensors," In *Companion Proceedings of the 26th International Conference on Intelligent User Interfaces*, pp. 43-45, 2021.
[25] M. Abd Al-Alim, R. Mubarak, N. M. Salem, and I. Sadek, "A machine-learning approach for stress detection using wearable sensors in free-living environments," *Computers in Biology and Medicine*, Vol. 179, pp. 108918, 2024.
[26] S. D. Sharma, S. Sharma, R. Singh, A. Gehlot, N. Priyadarshi, and B. Twala, "Deep recurrent neural network assisted stress detection system for working professionals," *Applied Sciences*, Vol. 12, No. 17, pp. 8678, 2022.
[27] L. Mou, C. Zhou, P. Zhao, B. Nakisa, M. N. Rastgoo, R. Jain, and W. Gao, "Driver stress detection via multimodal fusion using attention-based CNN-LSTM," *Expert Systems with Applications*, Vol. 173, pp. 114693, 2021.

## Authors' Profiles

**Mr. Ravi Arora** is currently working as Assistant Professor in Information Technology Department, Bharati Vidyapeeth's College of Engineering, Paschim Vihar, New Delhi. He is currently pursuing Ph.D. (CSE) from Lingaya University, Faridabad. He qualified his Master's in Technology in Information Technology Specialization in 2010 from USICT, IP University Main Campus. His research areas are Artificial Intelligence/Machine Learning, Computer Architecture, Computer Networks, Software Engineering.

**Dr. S. V. A. V. Prasad** did his M.Tech and Ph.D (Satellite Communications). Presently working as Professor, Dean (CA) and Director Lingaya's Vidyapeeth, Faridabad, Haryana. Presently guiding eight Ph. D Scholars in the Research Areas of Communication Engineering, thermal image processing for early diagnose of breast cancer, medical facilities for remote areas using m-health solutions, thought processing gadgets adoptive for broad band wireless communication and Semantic Web, Information Retrieval and so on.

**Dr. Arvind Rehalia** has done B. Tech (ECE), M. Tech (PCI) PGDBA, PhD (ECE) with more than 18 years of experience, he has made notable contributions to education and entrepreneurship. As Head of the Entrepreneurship Development Cell at Bharati Vidyapeeth's College of Engineering, New Delhi, he fosters innovation and nurtures aspiring entrepreneurs. His previous roles include Dean (Faculty Affairs) and leading the Instrumentation and Control Engineering Department. He has published four books. He is guiding three PhD Students and his commitment to maintaining high standards in education and research.

**Mr. Nikhil Kaushik,** Currently, he is undertaking a Ph.D. at Lingaya's Vidyapeeth, Haryana, contributing to innovative research and academic discourse in his specialization. He is currently serving as an Assistant Professor in the EEE Department at Maharaja Agrasen Institute of Technology (MAIT), since 2012, where he has been instrumental in shaping young minds and fostering technical excellence. He began his academic journey with a Bachelor of Technology (B.Tech) from Guru Gobind Singh Indraprastha University (IP University), where he laid a strong foundation in engineering principles. His passion for research and higher learning led him to pursue a Master of Technology (M.Tech) at Deenbandhu Chhotu Ram University of Science and Technology (DCRUST), further enhancing his expertise in the domain.

**Dr. Anil Kumar** is currently working as an Associate Professor in Department of Applied Sciences in Bharati Vidyapeeth's College of Engg. New Delhi. He has a teaching experience of more than 18 years to undergraduate and postgraduate students. He is supervising Ph.D. students at university level. He has obtained his PhD degree in Physics. He is Gold medalist in M.Sc. (Physics), Silver medalist in M.Phil. (Physics) in M.D.S.U. Ajmer. He has received academic excellence gold medal, awarded by D.A.V managing committee, Delhi in 2005.

**How to cite this paper:** Ravi Arora, SVAV Prasad, Arvind Rehalia, Nikhil Kaushik, Anil Kumar, "Early Detection of Stress and Anxiety Using NLP and Machine Learning on Social Media Data", International Journal of Information Technology and Computer Science(IJITCS), Vol.17, No.6, pp.70-94, 2025. DOI:10.5815/ijitcs.2025.06.04