Natural Resources for Human Health
eISSN: 2583-1194

Original Research

Received 26 May 2026
Revised 22 June 2026
Accepted 20 July 2026

Natural Resources for Human Health 2026, Vol. 6(7s): 1092–1108

KEYWORDS:
Natural language processing, depression detection, machine learning, text analysis, data mining.

Detecting Depression from Tweets through Topic Modeling and Lexicon-based Sentiment Fusion

Priyanka Srivastava[1], Dr. Mohammad Suaib[2]

[1]Department of CSE, Integral University, Lucknow, India
priyanka.lecturer.gpl@gmail.com
[2]Department of CSE, Integral University, Lucknow, India
suaib@iul.ac.in

ABSTRACT: Mental disorders such as depression constitute the largest share of concerns in mental well-being across the globe, having a severe influence on an individual's emotional, social and professional lives. With the increased application of social networks in the modern age, opportunities have opened up to detect the signs of depression among people using various tools based on textual content shared on these platforms. This paper describes a hybrid NLP approach towards detecting depression through analysis of the textual corpus of tweets based on the usage of LDA, VADER and TF-IDF features. The method includes a complete process of text preprocessing, consisting of tokenization, lemmatization, stop-word removal, punctuation elimination and de-noising to create a meaningful text corpus. The resulting topics generated using LDA, sentiments expressed through VADER and statistical weight of words captured using TF-IDF are used to generate feature vectors. The constructed feature matrix is further used to train and test a range of machine learning/deep learnimg models, such as SVM, LR, KNN, RF and LSTM. The evaluation of the proposed approach is carried out based on publicly available Twitter depression dataset consisting of over one million labeled tweets. As a result, it is shown that the suggested approach can capture the themes, emotions and context connected to depression. The experiments carried out using a number of different models have revealed that SVM provides the highest values of classification accuracy (95.1%). On the other hand, LSTM achieves accuracy of 96.5%.

1. INTRODUCTION

Depression is considered one of the most common and serious mental illnesses globally. Millions of people are affected by it within their personal, social and professional spheres. Traditionally, diagnosis of the disease is conducted using clinical interviews and subjective self-reporting, which are constrained by social stigma and lack of full disclosure. Hence, timely detection of depressive symptoms plays a vital role in ensuring prompt treatment. Due to the popularity of such social media platforms as Twitter, there is an extensive repository of tweets with behavioral and linguistic peculiarities of depressive nature, which can serve as a source of information for computational healthcare monitoring.

Notable advancements in sentiment and emotion analysis have been made by utilizing transformer models, graph neural networks and multimodal learning. Still, there remain significant challenges in dealing with noisy texts, code-switched language, and domain-specific data in addition to ensuring transparency and reliability of the model. For this reason, this paper proposes a hybrid sentiment analysis method that includes such techniques as Latent Dirichlet Allocation for extracting topics from the text, Valence Aware Dictionary and Sentiment Reasoner for estimating polarity of the text and TF-IDF as a way of assigning importance to the features.

The proposed research contributes to the field of computational healthcare monitoring by applying natural language processing tools in order to detect signs of depressive illness at the early stages. While the existing works mostly focus on

deep learning black-box models, the proposed method pays special attention to classification and explainability of the results. Additionally, the current paper shows how social media data can be used to advance the area of mental healthcare and public health care planning.

One of the important contributions of the paper is an integrated sentiment analysis framework that uses different types of features for depression detection in social media data. As a result of experimental analysis, it has been found that the combination of LDA topic models, VADER sentiment analysis and TF-IDF features achieves better classification accuracy compared to independent use of these techniques. The Support Vector Machine classifier reached 95.1% accuracy. Additionally, an approach to preprocessing and feature selection has been proposed in order to ensure robustness of the system.

The proposed system can be applied to real-time social media monitoring for identifying users with depressive signs in order to enable professionals working in mental healthcare, caregivers and policymakers to develop necessary intervention strategies. The methodology can be applied to various fields such as public opinion monitoring, customer feedback analysis and behavioral analytics. Also, the system has high interpretability, which makes it suitable for decision support systems.

2. RELATED WORK

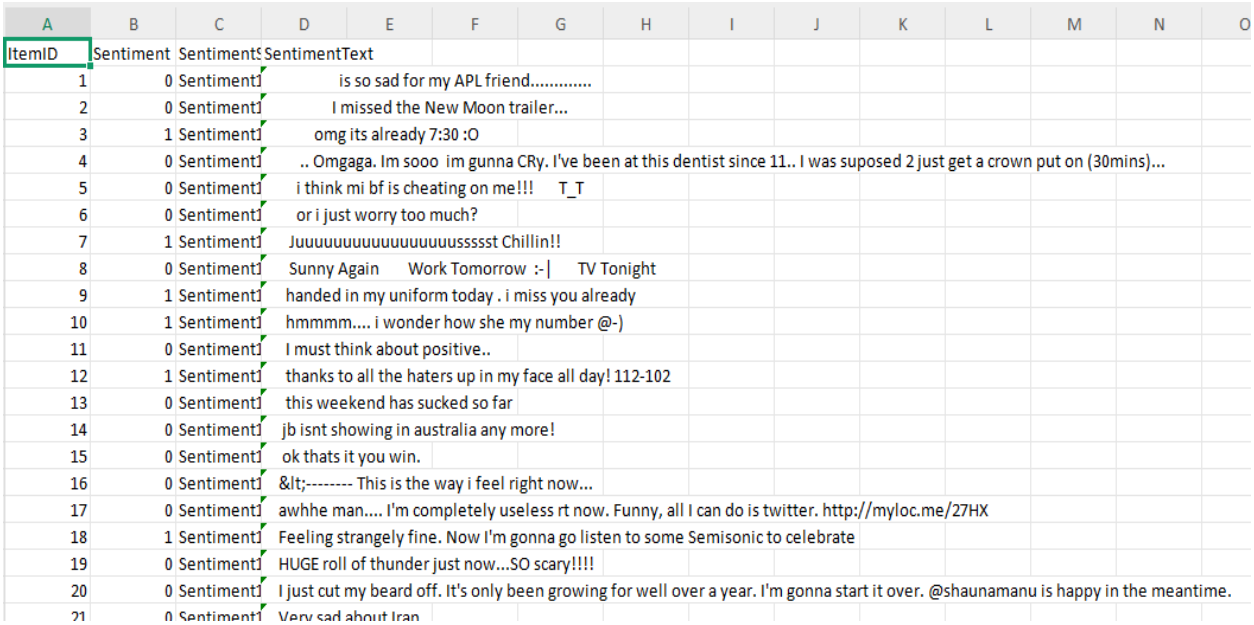
In their paper published in 2024, Naglik has introduced ASTE-Transformer, a novel dependency-aware transformer architecture designed to learn joint models of aspect, opinion, and sentiment triad relationship in the context of Aspect-Sentiment Triplet Extraction [1]. Avoiding the pipeline approach helped reduce error propagation and achieve state-of-the-art F1 scores on SemEval benchmark datasets. Continuing the trend of joint reasoning, Bodke has introduced PASTEL, a polarity-aware framework that consists of two parts: the generation of candidate triplets and verification of polarity consistency with a large language model. The framework showed impressive improvements in precision on noisy social media datasets [2]. In addition, Peng has introduced a Deep Relationship Enhancement Network, which uses iterative refinement of aspect–opinion interactions to provide more accurate results in extracting triplets [3]. Along with the mentioned frameworks, a syntax–semantics Graph Convolutional Network, which is able to combine syntactic dependency structure and semantic similarity graphs to propagate contextual information, can be considered [4]. Another example of such an improvement in triplet extraction is AWA-GCN, a framework that has used language-specific graph representation for triplet and quadruple extraction [5].

As another step toward advancing the existing methods, a framework for cross-lingual quadruple extraction, which is equipped with domain and language adapters, has been introduced in 2025 to provide good performance in diverse environments [6]. For the sake of more accurate extraction of aspect-opinion interaction, Piao et al. have used fused graph neural networks together with a biaffine attention mechanism to get more accurate relevance scores of pairwise relations and avoid mismatches in sentences with several aspects [7]. The topic of emotion recognition is another area where the development is ongoing not only in terms of text but also in the case of multimodal input. The vision transformer (ViT) model with a temporal convolution (ViTCN) component introduced in 2024 can jointly process spatial and temporal information in videos to recognize emotions with competitive benchmark performance [8]. Moreover, Rasool et al. have proposed nBERT, a transformer model fine-tuned for psychotherapy transcripts, which has proven to be effective for detection of subtle clinical emotions [9]. In addition, Rezapour et al. have performed a comparative study of transformer models' architectures for textual emotion classification and have found considerable differences in explanations' stability in attention-based and gradient-driven interpretability methods [10].

For SemEval-2024 Task 10, Creangă et al. have introduced the ISDS-NLP system, which utilizes contextual transformers and multilingual augmentation techniques to recognize emotions in conversational text [11]. In 2025, an anonymous author has combined a CNN-based feature extraction component with transformer encoders to recognize emotions in English-Hindi code-mixed social media posts [12]. Furthermore, the field of opinion extraction has been developing beyond single-target tasks. In particular, Zhao et al. have proposed a multi-target opinion extraction framework, which is based on the integration of graph convolution and sequence-labeling methods to extract opinions concerning multiple targets [13]. Zhu et al. have introduced a sentence-guided grid-tagging method for aspect-sentiment quadruples extraction, in which tokens are represented in two-dimensional grid allowing the joint decoding of aspects, opinions, categories, and sentiments [14]. In the medical domain, an anonymous author has developed a framework for integration of large language models and interpretable

rule-based classifiers to extract explainable aspect-sentiment pairs from clinical notes under strict transparency requirements [15].

Furthermore, an NLPCC 2023 span-based extraction framework enhanced with contrastive learning, in which negative sampling provided better separation between valid and invalid aspect–opinion pairs, has been introduced [16]. A RoBERTa-BiLSTM-Attention hybrid model proposed by unnamed researchers in 2025 has shown great performance in monitoring of the evolving sentiments across social media streams for public opinion analysis [17]. To overcome the problem of lack of Chinese-language resources, Lu et al. have proposed an iterative weak supervision technique that is able to generate automatically large-scale review dataset for triplet extraction, thus reducing the cost of annotations substantially [18]. In terms of visual emotion recognition, Kalsum et al. have developed a lightweight deep convolutional neural network to robustly recognize facial expressions despite illumination changes and demographic variation. This network can be deployed on mobile devices and in edge computing systems [19]. Similarly, Venkatesan et al. have used DeepFace embeddings together with downstream classifiers to recognize emotions from facial images, providing an example of practical image-based emotion analysis framework [20] while drawing attention to ethical issues related to cross-cultural interpretation and fairness.



ItemID	Sentiment	SentimentS	SentimentText
1	0	Sentiment1	is so sad for my APL friend.....
2	0	Sentiment1	I missed the New Moon trailer...
3	1	Sentiment1	omg its already 7:30 :O
4	0	Sentiment1	.. Omgaga. Im sooo im gunna CRy. I've been at this dentist since 11.. I was suposed 2 just get a crown put on (30mins)...
5	0	Sentiment1	i think mi bf is cheating on me!!! T_T
6	0	Sentiment1	or i just worry too much?
7	1	Sentiment1	Juuuuuuuuuuuuuuuussssst Chillin!!
8	0	Sentiment1	Sunny Again Work Tomorrow :-| TV Tonight
9	1	Sentiment1	handed in my uniform today . i miss you already
10	1	Sentiment1	hmmmm.... i wonder how she my number @-)
11	0	Sentiment1	I must think about positive..
12	1	Sentiment1	thanks to all the haters up in my face all day! 112-102
13	0	Sentiment1	this weekend has sucked so far
14	0	Sentiment1	jb isnt showing in australia any more!
15	0	Sentiment1	ok thats it you win.
16	0	Sentiment1	<-------- This is the way i feel right now...
17	0	Sentiment1	awhhe man.... I'm completely useless rt now. Funny, all I can do is twitter. http://myloc.me/27HX
18	1	Sentiment1	Feeling strangely fine. Now I'm gonna go listen to some Semisonic to celebrate
19	0	Sentiment1	HUGE roll of thunder just now...SO scary!!!!
20	0	Sentiment1	I just cut my beard off. It's only been growing for well over a year. I'm gonna start it over. @shaunamanu is happy in the meantime.
21	0	Sentiment1	Very sad about Iran

Figure 1. Screen shot of CSV file view of twitter sentiment labelled dataset used under this article for depression detection model.

3. METHODOLOGY

The designed sentiment analysis framework integrates several feature extraction approaches to provide a full-fledged representation of text for further processing by machine learning algorithms. As a first step, the raw text corpus is preprocessed in several steps that include tokenization, part-of-speech tagging, lemmatization, punctuation stripping, removal of stop-words, and filtering of too short and long tokens. In addition, unnecessary data related to the domain is filtered out such as HTML entities and special symbols. Next, a Bag-of-Words (BoW) model is constructed that reflects the distribution of terms in the text corpus. The feature extraction procedure is applied from three angles. Firstly, Latent Dirichlet Allocation (LDA) is employed to compute topic distributions that reveal latent thematic structures of documents. Secondly, lexicon-based approach to sentiment analysis is implemented with the use of Valence Aware Dictionary and Sentiment Reasoner (VADER). This technique produces scores reflecting emotions contained in textual data. Finally, TF-IDF values are calculated for the top-K most frequent terms. It means that those words that carry important information regarding particular documents but are rarely mentioned in the corpus are selected. Extracted features are compiled in a feature matrix that is later used for training supervised machine learning models such as SVM to predict sentiments and related issues (depression, etc.).

Natural Resources for Human Health
eISSN: 2583-1194

3.1 Data Used

The data considered in this work for model development and conducting experiments consisting of tweets that is labeled based on the features relaed to sentiment. This dataset, referred to the "twitter_sentiment", and free access in public from the Kaggle website. The size of this dataset is about 157 MB, and it consists of a total of 1,048,575 tweet entries. These tweets are assigned binary labels, where 0 refers to the Depressed class and 1 refers to Not Depressed class. Of the total dataset, 554,570 (52%) tweets belong to the Depressed class. This dataset has been developed particularly for conducting research related to depression detection from Twitter content. There are two main categories of tweets contained in the dataset: first, normal tweets which do not show signs of depression and second, tweets having textual features which indicate depressive behavior as shown in Figure 1 below.

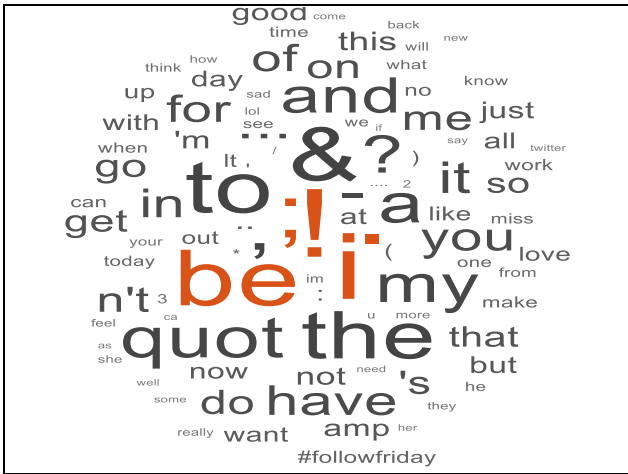


Figure 2a. Unique words in Raw data.

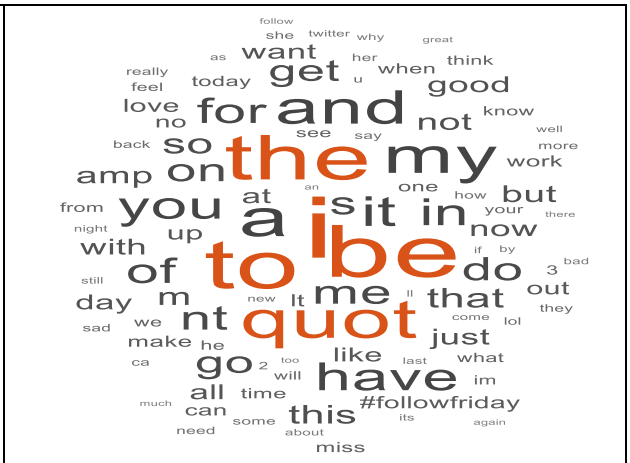


Figure 2b. Unique words after removing punctuation from raw data.

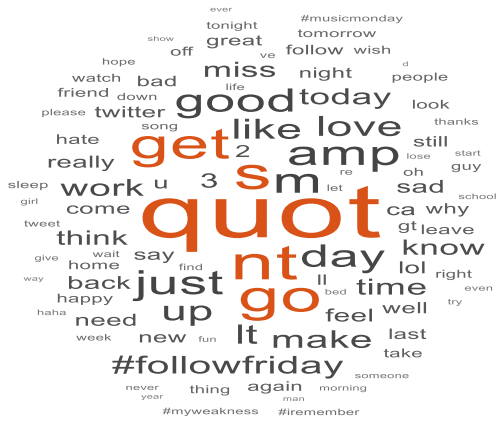


Figure 2c. Unique words after removing stop words from raw data.

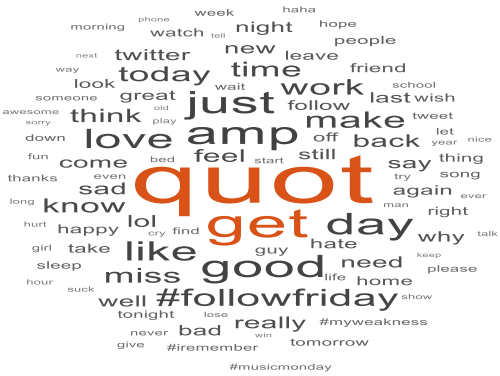


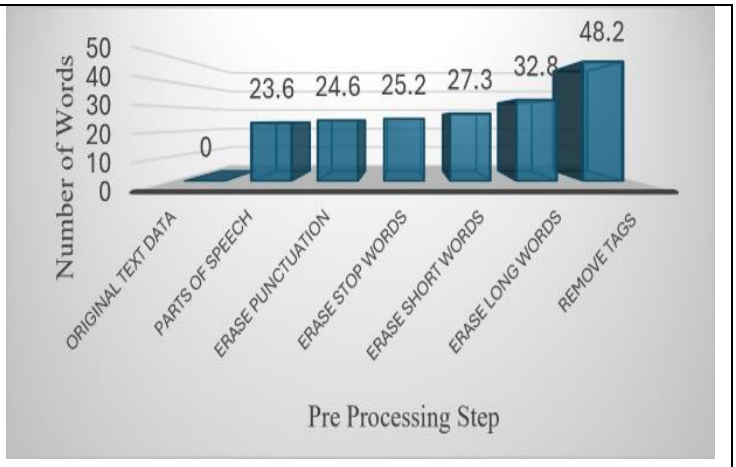
Figure 2d. Unique words after removing short/long words from raw data.

Natural Resources for Human Health
eISSN: 2583-1194

3.2 Data Preprocessing

Table 2. Reduction of unwanted words during preprocessing steps

Pre Processing Step	Number of Words	Percent Reduction
Original text data	22434	0
Parts of speech	17144	23.6
Erase punctuation	16923	24.6
Erase stop words	16787	25.2
Erase short words	16303	27.3
Erase long words	15085	32.8
Remove tags	11630	48.2



The sequence-based visualizations displayed in Figures 2a – 2d show that punctuation symbols such as ";", ".", "", and "!" take up a considerable amount of textual content from the dataset. Upon the elimination of those punctuation marks, frequently repeated words like "i", "to", "be", and "quot" prevail; however, such words do not provide any useful information when detecting depression patterns. In addition, further preprocessing based on stop-word elimination is shown in Figure 2c, where frequently repeated tokens such as "quot", "get", "go", "nt", "s", "2", "3", and "u" are revealed. In addition, tokens consisting of 2 or less letters as well as too long phrases and URLs of more than 100 characters are also excluded since they contain negligible amounts of useful information. The processed and original unprocessed word cloud visualizations are provided in Figure 3.

The new processed dataset includes more valuable words such as "good", "like", "feel", and "love", which help to find sentiment features. Such words can be useful for forming discriminant features for machine learning models. After filtering out unnecessary words, the clean Bag-of-Words (BoW) representation is formed. It includes the vocabulary of words and their frequencies of appearance in Twitter comments. This BoW representation is then used for calculating the number of repetitions for each unique word from the dataset. For analysis of vocabulary usage, the top 100 most frequently appearing words for both label values 0 (Depressed) and 1 (Not Depressed) are found. Those word lists will help to reveal linguistic differences between depressed and non-depressed people. The obtained lexical information will then be used for developing a set of features and machine learning model, as it is explained in the following sections.

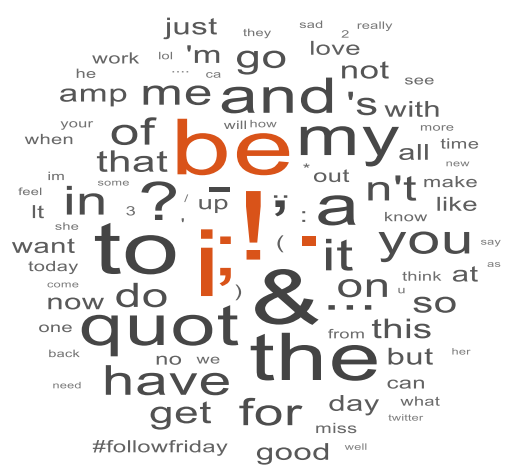


Figure 3a. Raw Data

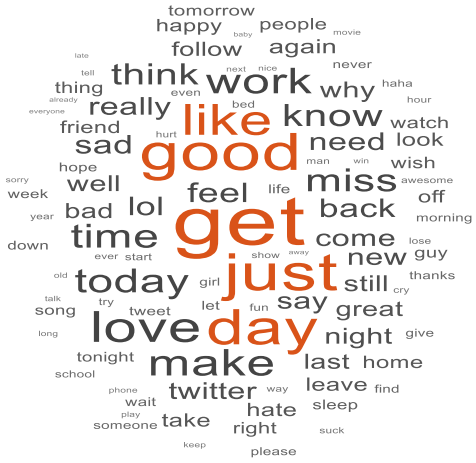


Figure 3b. Final Cleaned data

Natural Resources for Human Health
eISSN: 2583-1194

3.3 Feature extraction

Once irrelevant textual content is removed, the resulting cleaned dataset is then used for the creation of three feature representations, which include: (a) Latent Dirichlet Allocation (LDA), (b) Term Frequency-Inverse Document Frequency (TF-IDF), and (c) Valence Aware Dictionary and Sentiment Reasoner (VADER). Each of the feature representations captures certain unique aspect of the textual content and when used together, they create a more informative representation of Twitter data. Below are procedures for obtaining these features from the cleaned sentiment dataset.

LDA is probabilistic generative modelling technique that seeks to uncover latent thematic structure present in large sets of text documents. Based on this model, documents are considered as a mixture of several latent topics where each topic is a probability distribution over the vocabulary. Uncovering these latent themes allows documents with common themes to be represented by concise vector of probability of topics. LDA model uses Dirichlet distribution to obtain the proportion of topics in each document and distribution of words in topics in order to generate textual content probabilistically. For predetermined categories:

$$Dir(\theta \mid \alpha) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$$

θ_i: Probability of category I,$\Gamma(\cdot)$: Gamma function, α_i= Prior concentration parameter,

The parameter αcontrolling the sparsity where low value of α i.e. α<1 producing sparse distributions and highlevelis producing uniform distribution.

Multinomial Distribution models: It is showing that the outcomes count are lying across multiple categories.For Knumber of categories:

$$Multinomial(x) = P(x_1, \ldots, x_K) = \frac{M!}{x_1!\, x_2! \ldots x_K!} \prod_{i=1}^{K} p_i^{x_i}$$

M: Total number of trials, x_i: category count of i, p: i^{th}category probability.

In this work calculated of LDA is performed from Gibb's sampling formulae as the probability that word i is belonging to topic k:

$$P(z_i = k \mid z_{-i}, w_i) \propto \frac{n_{dk}^{-i} + \alpha}{\sum_k n_{dk}^{-i} + K\alpha} \times \frac{n_{kw}^{-i} + \beta}{\sum_w n_{kw}^{-i} + V\beta}$$

Where:distribution of document-topic: $\theta_d \sim Dirichlet(\alpha)$

Generation of word from topic: $w_{dn} \sim Multinomial(\phi_{z_{dn}})$

Where: ϕ_k= word distribution for topic k

(b) **VADER (Valence Aware Dictionary and Sentiment Reasoner):**ADER is an approach to sentiment analysis that relies on lexical means to determine the emotional polarity conveyed in the text data. This approach assigns sentiment scores on four levels, such as positive, negative, neutral, as well as a combined one, thus allowing for a detailed analysis of emotional expression. Due to its efficiency in analyzing short text messages, social media posts, and non-formal text writing, VADER has gained much popularity as a tool for performing sentiment analysis on user-generated data. The sentiment scores calculated using VADER can be viewed as important input features for the machine learning model because they reflect emotional characteristics and polarity of the respective documents. For the valence score vi of the i^{th} word:

S: Base valence score: It calculatesstrength of total sentiment for all sentiment-bearing words: $S = \sum_{i=1}^{N} v_i$
B: Booster adjustment increasing or decreasingintensity of sentiment by modifier words;$S' = S + B; B = \pm 0.293$

N: Negation adjustment for reversing or weakeningpolarity of sentiment byterms of negation:$S' = S \times N$; $N = -0.74$

C: Capitalization emphasis increasingstrength of sentiment for uppercase words:$S' = S + C$and$C = 0.733$

EP: Exclamation emphasis increasingintensity of emotion bycountingexclamation marks:

$$EP = \min(N_!, 4) \times 0.292$$

QP: Question mark emphasis for adjustingintensity of sentiment on the basisof question marks:

$$QP = \begin{cases} 0.18 \times N_?, & N_? \leq 3 \\ 0.96, & N_? > 3 \end{cases}$$

Contrastive conjunction rule used for reducing or emphasizing sentiment before or after "but": $S_{before} = 0.5S$ *and* $S_{after} = 1.5S$

Total valence score arecombining all adjusted contributions of sentiment as a raw score: $x = \sum_{i=1}^{N} v_i + B + C + EP + QP$

Compound score normalization converts raw sentiment score in between −1 and +1:$compound = \frac{x}{\sqrt{x^2+\alpha}}; \alpha = 15$

Positive/Negtive/Neutral sentiment score measures the proportion of positive sentiment in the text.

$Positive = \frac{\sum positive\ valence}{Total\ valence}$; $Negative = \frac{\sum negative\ valence}{Total\ valence}$; $Neutral = \frac{Neutral\ Tokens}{Total\ Tokens}$

Final constraint ensures that all sentiment proportions sum to unity: $Positive + Negative + Neutral = 1$

(c) **TF-IDF (Term Frequency-Inverse Document Frequency)**:TF-IDF refers to the statistical approach used to calculate the importance of a term within a document relative to the whole corpus. In the case of the TF component, the frequency of the term used within a particular document is measured, while the IDF component lowers the weight of words that appear too frequently in many documents. This process helps to determine which terms have higher discriminative power in the process of differentiating one document from another. TF-IDF gives prominence to terms with high discriminatory power. The resultant weights of these terms serve as useful inputs in machine learning-based classification and sentiment analysis tasks.

TF measures how frequently a term appears in a document: $TF(t,d) = \frac{f_{t,d}}{\sum_k f_{k,d}}$

IDF measures how important a term is across all documents:$IDF(t) = \log\left(\frac{N}{df_t}\right)$

Smoothed IDF avoids division by zero and stabilizes the weighting: $IDF(t) = \log\left(\frac{1+N}{1+df_t}\right) + 1$

TF-IDF score calculates the weighted importance of a term in a document: $TFIDF(t,d) = TF(t,d) \times IDF(t)$

3.4 Machine Learning (ML) model development:

The feature vectors obtained from each of the comments in Twitter are combined to create an aggregate feature matrix, which is made up of three feature vectors. This feature matrix is referred to as the input dataset X while the labels from each tweet are given to the output dataset Y. In order to train the machine learning models and test them, the whole data set is randomly partitioned into two portions, the training and testing datasets, with a split ratio of 70% for training and 30% for testing.

The training dataset, is fed to the machine learning algorithms in order to develop classification models that can predict if a given text instance is depressed or not. This classification is done based on the feature vector representation of the tweets. To explore the efficiency of different learning techniques, four machine learning models are developed and analyzed: Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbors (KNN), and Random Forest (RF).

(a) Random Forest

Random Forest (RF) is used widely as a supervised machine learning algorithms that is applied for both purposes of classification and regression. It is used with the concept of ensemble learning. The multiple decision trees are created and executed together for improvement of the predictive accuracy of a system under machine learning while decreasing the overfitting risk. The algorithm was developed by Leo Breiman in 2001 and became one of the most popular machine learning approaches because of its stability, flexibility, and good predictability. Random Forest has been used in many areas including medicine, meteorology, image recognition, fraud detection, and IoT-based applications.

A large set of decision trees is built during the training stage and the decisions made by them are combined into one. The combination of multiple decisions allows increasing the robustness of the model and improving the generalization ability compared to a single decision tree that is sensitive to the variations in the training set and can have a high variance.

Training of a Random Forest algorithm starts from a data set which looks like:

$$D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$$

Where x refers to the input set of features and y refers to the output features. In Random Forest, several bootstrap samples are created from the original dataset using random sampling with replacement. Each of these bootstrap samples is used for building individual decision trees. In contrast to the regular decision trees that consider all features present in the dataset while splitting, only some random features are selected in Random Forest. This helps in reducing the correlation among different trees and increases their diversity. The best splitting criteria in each case are calculated using impurity measures like the Gini Index:

$$Gini(t) = 1 - \sum_{i=1}^{c} p_i^2$$

where indicates the likelihood of class occurrence. The smaller the Gini impurity value is, the cleaner the node is, implying that within that particular node, most samples belong to one class. After the ensemble of trees has been trained already, the final prediction can be obtained through the aggregation of predictions of all trees. In classification tasks, the class that receives the most votes is selected based on a majority vote method, while for regression tasks, the estimated value (s) of each individual tree is averaged:

$$\hat{y} = \frac{1}{B}\sum_{b=1}^{B} T_b\,(x)$$

where B is the number of decision trees in the ensemble and $T_b(x)$ is the estimate of the b^{th} decision tree. Random Forest obtains the total prediction based on the output provided by each tree in the forest. As a result of this procedure, it obtains better normalization of the predictions obtained by using one tree.

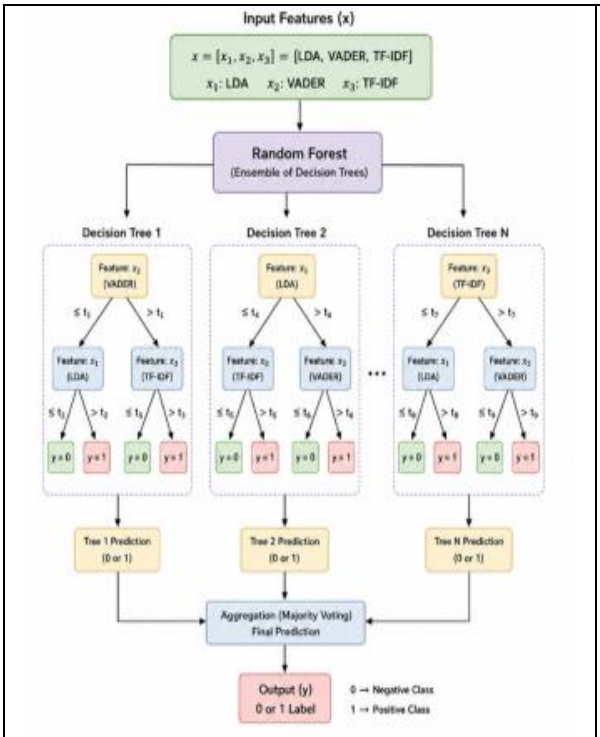


Figure 4a. Model working forML classifier.as random forest

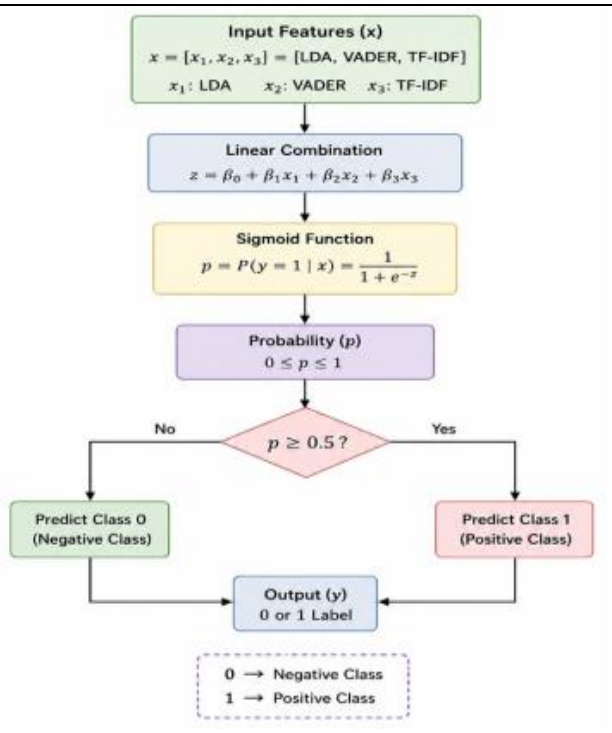


Figure 4b. Model working for ML classifier.as random forest Logistic regression.

Natural Resources for Human Health
eISSN: 2583-1194

(b) Logistic Regression

Logistic Regression is an approach of machine learning that used under supervised learning. The primary aim of the algorithm is to provide a probability score based on the observations that it has been trained on. It can classify observations as belonging to a Yes/No or True/False Class and is better suited for binary classification problems. Unlike, linear regression, where there are no limits to the output, in Logistic Regression there is a sigmoid function involved in the calculation restricting the predicted probability value to a range of 0 and 1. Given the ease of implementation, computation, and interpretation, it has found its applications in several domains such as medical diagnosis, fraud detection, sentiment analysis, and risk prediction. The initiation of the Logistic Regression is done with a linear function as follows:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

where $x_1, x_2, \ldots, x_n$ represent input variables and β_0 is the intercept, $\beta_1, \beta_2, \ldots, \beta_n$ are model coefficients. The value that are obtained is passed through the sigmoid activation function for convertingthemas form ofprobability:

$$P(y = 1) = \frac{1}{1 + e^{-z}}$$

The sigmoid function is applied in Logistic Regression to transform the model's linear output into probabilities ranging from 0 to 1. In cases of binary classification, the output showing the probability of greater than 0.5 is assigned to class 1 while results with probability lower than 0.5 belong to class 0. Model parameters are determined through the Maximum Likelihood Estimation (MLE) that finds values of parameters based on their consistency with the data observed during training and thus helps to improve classification results.

Logistic Regression is good at classification tasks when the underlying data allows the distinctions being made between the classes going along a straight line. The computational latency involved in the algorithm implementation is low and the computational complexity is also low, thus making the approach viable for numerous classification tasks. Additionally, Logistic Regression is able to resist overfitting in small and medium datasets and its coefficients show how much each individual variable contributes to the outcome. However, when it comes to highly nonlinear data or data with more complex distribution even Logistics Regression has its limitations.

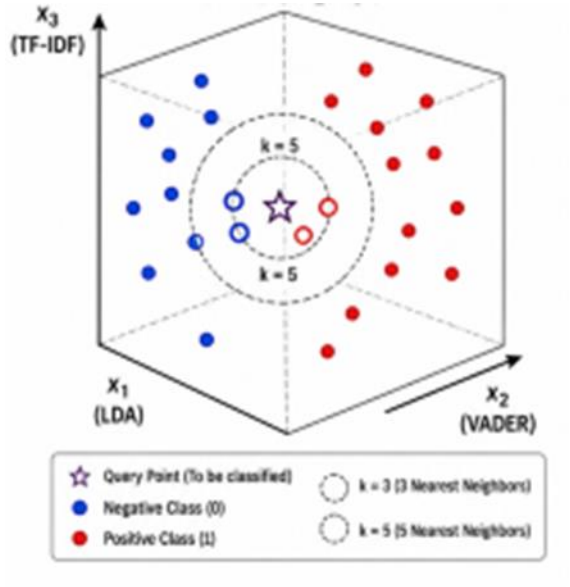


Figure 4c. Working model of K-NN as ML classifier.

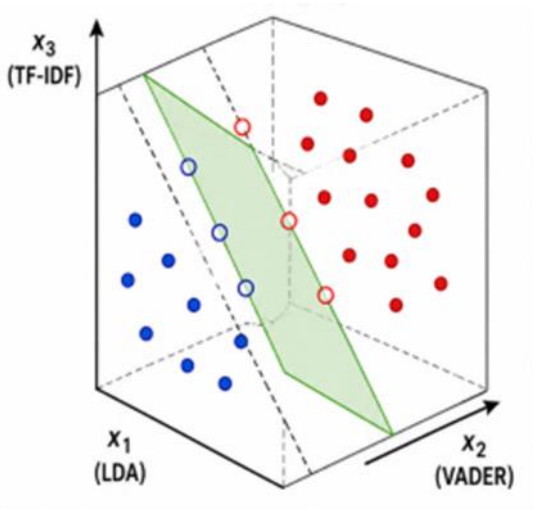


Figure 4d. Working model of SVM as ML classifier.

Natural Resources for Human Health
eISSN: 2583-1194

(c) K-Nearest Neighbors (KNN)

KNN is an approach of supervised learning that is extensively used for classification and regression problems. This is a non-parametric instance-based learning technique that performs predictions by the similarity of the observations. The algorithm identifies the closest training samples for a new sample and uses them to make the prediction. The popularity of KNN is due to its ease of use and challenging delivery. KNN stores training samples in the feature space rather than building a predictive framework. When a new input sample is given to the algorithm, KNN computes its distance with respect to already existing training samples.

Euclidean distance is mostly used to measure this distance.

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

where x_i and y_i are the values of the respective attributes for two points. In these computations, the total number of features is represented by n. Once the distances have been determined, KNN chooses K instances with the lowest distances. For classification purposes, the output class is determined by voting, whereas for regression purposes the output is the average of the outputs of K instances obtained.

The effectiveness of KNN algorithm depends heavily on the value determined for K. A small value for K may make the classifier too responsive to noise and lead to overfitting, while a large value for K may oversmooth the data structure and reduce classification accuracy. Since KNN does not require additional training, it can be easily implemented, but it has a high computation cost for large datasets since it needs to perform distance evaluations for every prediction.

Table 4: Top Most Frequently Used Hundred Words

Case: Normal					**Case: Depressed**			
*Bold Fonts: Common Words, **Yellow Highlights: Positive Words					*Bold Fonts: Common Words, **Yellow Highlights: Negative Words			
Again	god	Man	Start		**again**	**guy**	**morning**	**Still**
Always	**good**	**Miss**	**Still**		already	**happy**	**need**	Stupid
amazing	**great**	**Morning**	**Take**		another	hate	**never**	Suck
awesome	**guy**	Movie	**Talk**		away	**home**	**new**	**Take**
Baby	haha	Music	**Tell**		**back**	**Hope**	next	**Talk**
Back	**happy**	**Need**	Thank		bad	Hour	**night**	**Tell**
Bed	**home**	**Never**	Thanks		**bed**	Hurt	**off**	**Thing**
Call	**hope**	**New**	**Thing**		break	Iphone	**old**	**Think**
Come	hot	Nice	**Think**		**come**	**Just**	**people**	Though
Cool	**just**	**Night**	**Time**		cry	**Keep**	phone	**Time**
Day	**keep**	**Off**	**Today**		damn	**Know**	**play**	**Today**
Down	**know**	**Old**	**Tomorrow**		**day**	**Last**	**please**	**Tomorrow**
Enjoy	**last**	**People**	**Tonight**		die	**Leave**	rain	**Tonight**
Ever	late	**Play**	Tweet		**down**	**Let**	**really**	Try
everyone	**leave**	**Please**	**Twitter**		even	**Life**	**right**	**Twitter**
Feel	**let**	Ready	**Wait**		**ever**	**Like**	sad	**Wait**
First	**life**	**Really**	**Watch**		**feel**	**Lol**	**say**	**Watch**
Follow	**like**	**Right**	**Way**		find	Long	**school**	**Way**
Friend	listen	**Say**	**Week**		**friend**	**Look**	**show**	**Week**
Fun	live	**School**	Weekend		beat	Lose	sick	**Well**

Funny	**lol**	**Show**	**Well**		**get**	**Love**	**sleep**	**Why**
Game	**look**	**Sleep**	**Why**		**girl**	**Make**	**someone**	Win
Get	lot	Smile	**Work**		**give**	Mean	something	Wish
Girl	**love**	**Someone**	World		**good**	**Miss**	soon	**Work**
Give	**make**	Song	Yay		**great**	Mom	sorry	Year

(d) Support Vector Machine (SVM)

Support Vector Machine (SVM) is a ML algorithm that falls under the category of supervised learning. It works best for classification problems, particularly for binary classification where SVM's goal is to discover an appropriate decision boundary separating two classes from one another. Since SVM is great in generalization and dealing with high-dimensional input spaces, it is applicable in such fields as image classification, text classification, medical diagnosis, handwriting recognition, and bioinformatics.

The methodology underlying SVM is being able to find a hyperplane that allows maximizing the distance between classes. The nearest samples to this hyperplane are called support vectors and those samples are used to come up with the final decision boundary. The mathematical representation of SVM for the dataset with the input vector x and the class y is given below:

$$f(x) = w^T x + b$$

where w represents the weight vector and b denotes the bias term. The optimal hyperplane is determined by maximising the separation between the support vectors and the decision boundary. The corresponding margin is defined as:

$$Margin = \frac{2}{\| w \|}$$

SVM defines this process as an optimisation problem that functions by reducing classification errors and increasing the margins. If the data is not separable by using a linear decision boundary, SVM can also use kernel functions to convert data from empirical space to a feature space that has a higher dimension. The kernel functions that are used in SVM are linear, polynomial, radial basis function (RBF), and sigmoid kernels. The kernel transformation is given as follows:

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

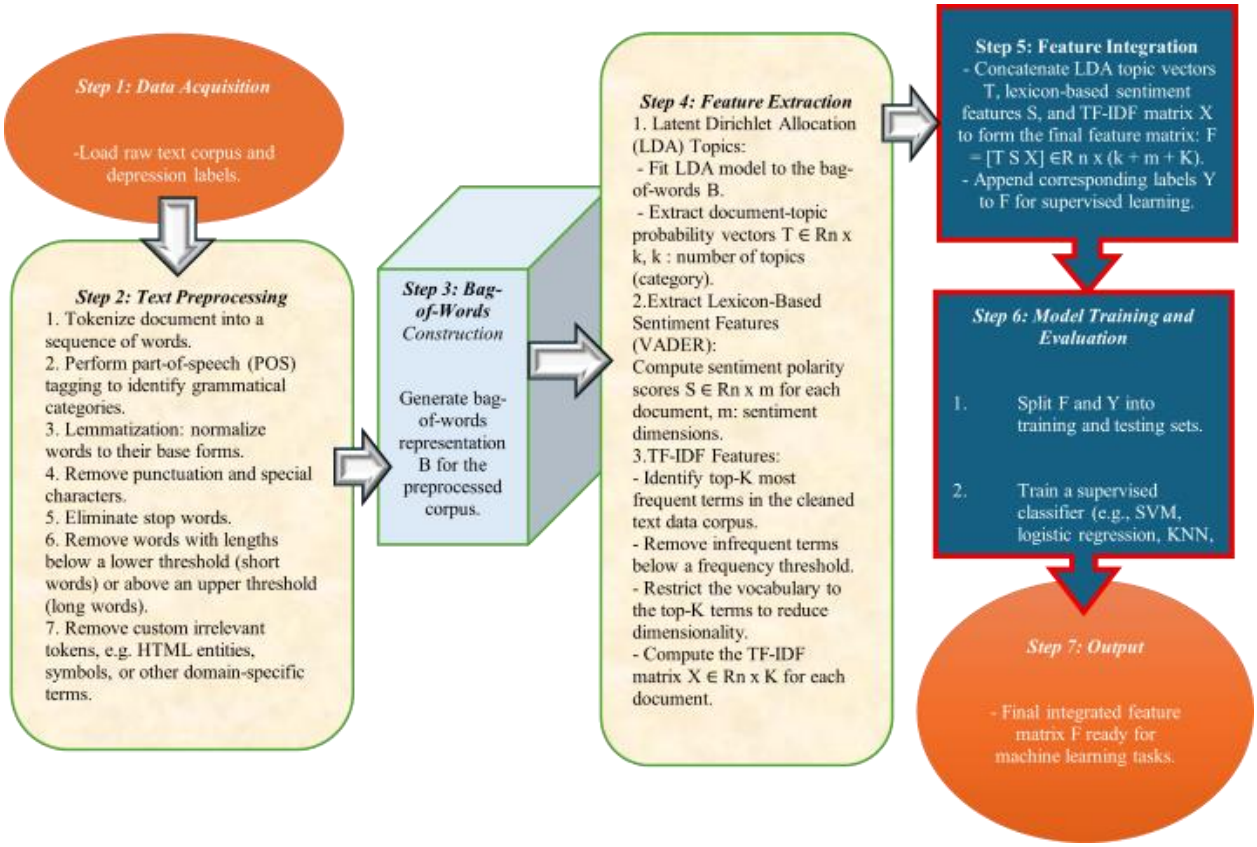
where φ indicates the mapping function.

In most cases, SVM works with a great classification accuracy and shows the good results on complex features' datasets with a small number of samples. Due to the high dimensions of the feature space, SVM is less vulnerable to overfitting. However, it is necessary to mention that when dealing with extremely high volumes of data, a significant growth in complexity of training can be observed, while selecting the kernel functions and their parameters will require further efforts. Despite of the mentioned constraints, SVM is still considered to be one of the effective ML algorithms.

Abbreviations	**Symbols**	
TF-IDF: Term Frequency-Inverse Document Frequency LDA: Latent Dirichlet Allocation VADER: Valence Aware Dictionary and Sentiment Reasoner POS: Part-of-Speech BOW: Bag-of-Words ML: Machine Learning SVM: Support Vector Machine	D: Input text corpus d_i: i^{th} document in the corpus Y: Label vector for all documents y_i: Label for i^{th} document F: Final feature matrix B: Bag-of-words matrix T: Document-topic probability matrix of LDA	S: Lexicon-based sentiment score matrix X: TF-IDF feature matrix n: Number of documents k: Number of topics in LDA m: Number of sentiment dimensions K: Number of top TF-IDF terms

Natural Resources for Human Health
eISSN: 2583-1194

Machine Learning Algorithm:



3.5 Deep Learning

Deep learning is a branch of machine learning that is based on fundamental concepts of artificial neural networks. It is using multiple layers that mimic the structure of a human brain for the task of data analysis. This helps AI in learning complex patterns from large volumes of unstructured data. Technologies behind the voice recognition, computer vision, and generative AI are powered by deep learning. There are two main types of dep learning processes [26 - 29]:

•CNNs (Convolutional Neural Networks): It is a very efficient for the application based on analysis of spatial dat. Commonly used in images data through the use of filters in pattern recognition.

•RNNs (Recurrent Neural Networks): They are efficiently used in the sequential data processing loke texts and time-series because they possess internal memory to process data in sequence.

The key foundation of deep learning is the use of neural networks. Deep learning is designed following the structure of the brain, which comprises layers of "neurons" in charge of processing information. Deep learning is based on the representation of data by increasingly complex levels of representations, from simple edges to complex objects. In contrary to shallow neural networks, deep learning involves the use of four or more layers in the process of data analysis. The models are more accurate when they are supplied with large amountities.

3.5.1 Recurrent Neural Networks (RNNs)

RNN are an artificial intelligence technique used to analyze sequential data (text, speech, time-series), by creating an internal memory that helps to store information from previous inputs. The RNN uses the same operation on each member of a sequence, which helps in influencing their predictions by past information.

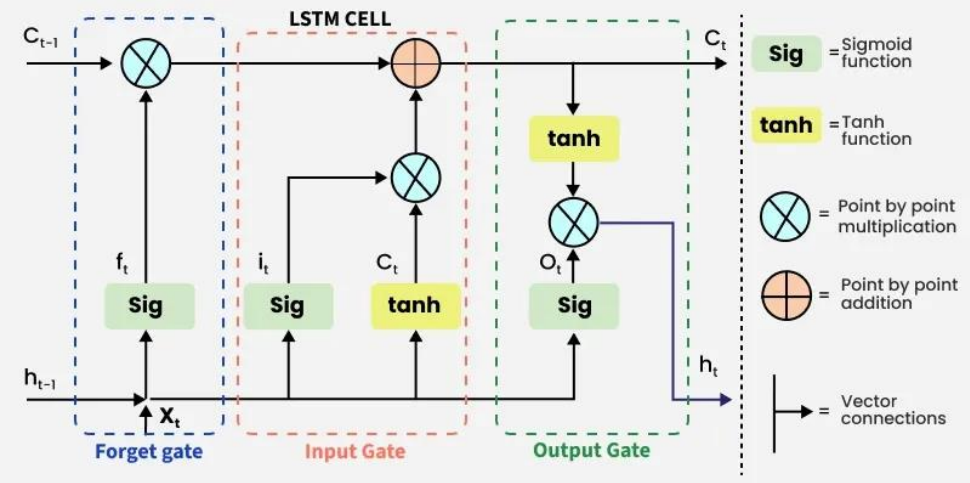
Natural Resources for Human Health
eISSN: 2583-1194

3.5.2 Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) networks are a variant of recurrent neural networks that overcome the challenge of vanishing gradient, thus they are used to learn dependencies from sequential data. It incorporates three gates (input gate, forget gate, and output gate) and special memory cell for information flow.

It is suitable for:

- Handles Long Term Dependencies: Retains information over the long period
- Memory Cells: Information storage and updating
- Better than RNN: Overcomes short term memory problem
- Application: Used in languages translation



Deep Learning Algorithm

Step 1: Data Acquisition - Load the raw text corpus and corresponding sentiment/depression labels.

- Select the subset of documents for processing if necessary.

Step 2: Text Preprocessing

Step 3: Bag-of-Words Construction

Step 4: Feature Extraction

1. Latent Dirichlet Allocation (LDA) Topics:

- Fit an LDA model to the bag-of-words B.

- Extract document-topic probability vectors T ∈ Rn x k, where k is the number of topics.

2. Lexicon-Based Sentiment Features (VADER):

Compute sentiment polarity scores S $\in R_{nxm}$ for each document, where m represents sentiment dimensions.

3. TF-IDF Features:

- Identify the top-K most frequent terms in the cleaned corpus.

- Remove infrequent terms below a frequency threshold.

- Restrict the vocabulary to the top-K terms to reduce dimensionality.

- Compute the TF-IDF matrix X ∈ $R_{n \times K}$ for each document.

Step 5: Feature Integration

- Concatenate LDA topic vectors T, lexicon-based sentiment features S, and TF-IDF matrix X to form the final feature matrix: $F = [T\ S\ X] \in R_{n \times (k + m + K)}$

- Append corresponding labels Y to F for supervised learning.

Step 6: Model Training and Evaluation

1. Split F and Y into training and testing sets.

2. Train a classifier using the training set by LSTM.

3. Evaluate performance using accuracy, precision, recall, or F1-score on the test set.

Step 7: Output

- Final integrated feature matrix F ready for machine learning tasks.

- Classification performance metrics for model evaluation.

4. RESULTS

The above-mentioned methodology is described in the concise form of an algorithm. The proposed algorithm allows creating a structure for representation of MATLAB implementation of the method which is based on in-built functions provided by the Signal Processing, NLP, and Machine Learning toolboxes. In accordance with the proposed methodology, four different ML models are created based on the training dataset, and several experiments are performed for each of them using different parameter configurations.

Two parameters are studied further in order to decrease the volume of data necessary and control the complexity of the created model. The first is the number of infrequent words and the second is the number of top words used based on the unique vocabulary created from the cleaned bag-of-words dataset. Different numbers of infrequent words, which can be equal to 5 and 10 and represent 2 variants, and different numbers of top words which can be equal to 30, 50, and 70 and represent 3 variants are considered. Hence, six experimental cases are created to study the influence of these two parameters on the performance of the ML model (Case ID 1-6). The accuracy results for the corresponding cases are represented in Table 5. Each column presents the ID of one case whereas the three rows present the number of infrequent words, the number of top words, and accuracy (%) correspondingly. The minimum value of accuracy obtained in the experiments equals 86%, and the maximum value equals 95%. The highest accuracy of 95.1% is achieved for Case ID 4 when the values of the number of infrequent words and top words are set to their maximum values, i.e. 10 and 70.

The performance of the created model for the selected case (Case ID 4) is estimated using several additional metrics calculated based on the elements of the confusion matrix. The results of different combinations of the number of infrequent words and top words are summarised in Table 6 using precision, recall, F1-score, and accuracy metrics. Among the compared models, SVM demonstrates the best performance using the Gaussian kernel and its kernel scale being 7.3. The second highest performance is observed for KNN with 10 neighbours, Euclidean distance metric, and squared inverse distance weighting.

Table 5: Accuracy comparison at different cases in terms of number of infrequent words and top words

Case Id:	1	2	3	4	5	6
Number of infrequent words	5	5	5	10	10	10
Number of top words	30	50	70	30	50	70
Accuracy (%)	86	92	93.7	95.1	94.8	94.2

Table 6: Summary of best results obtained after testing different experimental cases for Number of Infrequent words =10, Number of Top-words=50, features=54.

	Precision	Recall	F1-Score	Accuracy	Parameters
SVM	98.8 %	89.4 %	93.9 %	95.1 %	Kernel: Gaussian, Kernel Scale=7.3
Logistic regression	89.6 %	88.6 %	89.0 %	90.9 %	Linear equation
KNN	94.7%	91.0%	92.7%	94.1%	Number of neighbors=10, Distance= Euclidian, Distance weight= squared inverse
Random Forest	82.9%	68%	74.7%	77%	Number of splits=9999, Number of learners=30, Learning rate=0.1
LSTM	98.8%	90.2%	94.3%	96.2%	3 Layer

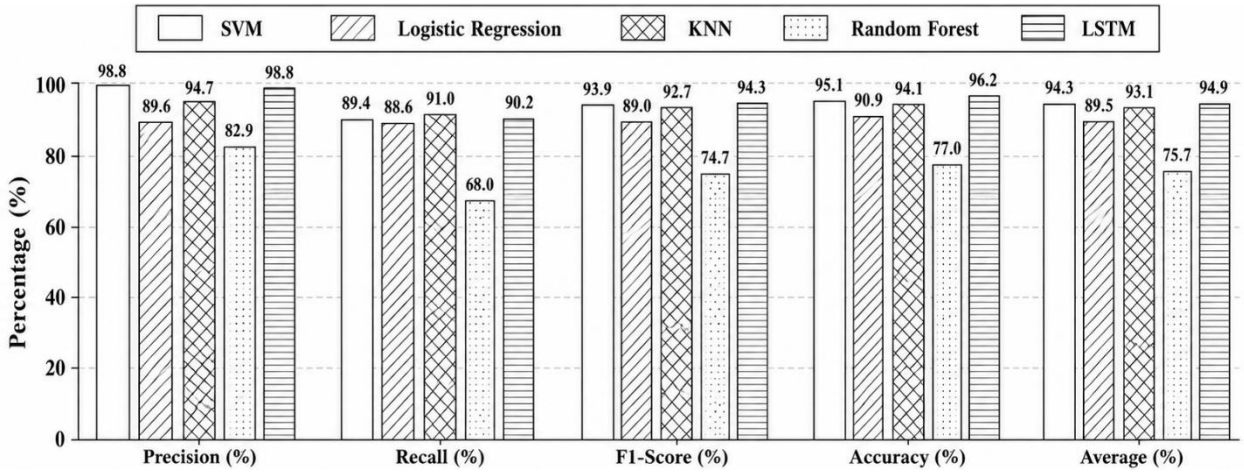


Figure 5. Bar chart representation of collective demonstration of precision, recall, F1-score and accuracy.

Table 7. Comparison to latest research works based on NLP for sentiment analysis by text data.

Technique	Precision	Recall	F1-measure	Accuracy
SABSA [22]	0.858	0.839	0.8485	0.837
SentiVec [21]	0.877	0.858	0.8675	0.861
Ngram+TF-IDF+SVM [23]	0.866	0.846	0.856	0.844
SEML [25]	0.854	0.837	0.8455	0.838
MTMVN [24]	0.817	0.789	0.803	0.792
LSTM (Proposed)	0.988	0.902	0.943	0.962

5. CONCLUSION

In this study, an integrated framework is proposed for detecting depression from Twitter posts, in which the combined use of Latent Dirichlet Allocation (LDA) based topic modeling, sentiment analysis via VADER tool, and TF-IDF technique has been done to capture latent thematic knowledge, polarity of users' emotions, and words' significance within a post respectively, thus leading to a representation of the user's behavior and his/her psychological status. The process of preprocessing has been executed extensively in order to discard any type of irrelevant textual elements and noisy information.

In this study, the fused feature vector has been used to train multiple Machine Learning and Deep Learning algorithms such as Support Vector Machine (SVM), Logistic Regression, KNN, Random Forest, and LSTM for classifying the depressed

versus non-depressed status of the users. According to the results of the experiment, the proposed integration method has provided significant improvements for the depression detection task as compared to common text representation approaches. SVM and LSTM algorithms have given better results among the tested ones where SVM has provided 95.1% accuracy, 98.8% precision, 89.4% recall, and 93.9% F1-score for the classification task. The LSTM provides 96.2% accuracy, 98.8% precision, 90.2% recall, and 94.3% F1-score

This study proves the effectiveness of the combined usage of probabilistic topic model, sentiment lexicon-based scoring, and statistical text representation approaches for depression detection problem. Compared to other methods, the proposed framework provides high interpretability level of the algorithm's outputs due to the fact that most of the modern Deep Learning techniques have black box nature, which is not convenient for healthcare-oriented decision-support systems.

ACKNOWLEDGEMENT

All the authors of this article are very thankful to the Computer Science and Engineering Department and Research and development centre of Integral University for the guidance and support for this research work. We acknowledge the communication of this article for purpose of publication under the Manuscript communication number (MCN): IU/R&D2026-MCN0004916.

REFERENCES

[1] Naglik, I., & Lango, M. (2024). ASTE-Transformer: Modelling dependencies in aspect-sentiment triplet extraction. *Proceedings of the Association for Computational Linguistics (ACL).*

[2] Bodke, A., Zhang, Y., & Lee, H. (2025). PASTEL: Polarity-aware sentiment triplet extraction with LLM-as-a-judge. *IEEE Transactions on Affective Computing*, Advance online publication.

[3] Peng, J., & Su, B. (2024). Aspect sentiment triplet extraction based on deep relationship enhancement networks. *Knowledge-Based Systems*, 295, 111380.

[4] Zhang, J., Xu, S., Gao, X., & Tang, Z. (2025). Aspect Sentiment Triplet Extraction with Syntax-Semantics Graph Convolutional Network. *International Journal of Computational Intelligence Systems*, *18*(1), 167.

[5] Haocheng Xi, Yutong Wang, Xinyu Wang, Zhitao Li, Bochun Liu, Yihan Wang, Chen Ni (2024). AWA-GCN: Enhancing Chinese sentiment analysis with graph convolutional networks. *Information Processing & Management.*

[6] Çekinel, R. F. (2025). *Multilingual, Multimodal and Explainable Approaches for Automated Fact-Checking Problem* (Doctoral dissertation, Middle East Technical University (Turkey)).

[7] Piao, Y., & Zhang, J. (2024). Text triplet extraction with fused graph neural networks and improved biaffine attention. *Applied Intelligence*, 54(6), 6570–6584.

[8] Zakieldin, K., Khattab, R., Ibrahim, E., Arafat, E., Ahmed, N., &Hemayed, E. (2024). Vitcn: Hybrid vision transformer with temporal convolution for multi-emotion recognition. *International Journal of Computational Intelligence Systems*, *17*(1), 64.

[9] Rasool, Z., Khan, S., & Ahmad, F. (2025). nBERT: Harnessing NLP for emotion recognition in psychotherapy. *Frontiers in Psychology*, 16, 1234567.

[10] Rezapour, M. (2024). Emotion detection with transformers: A comparative study of explanation methods. *Computers in Human Behavior*, 154, 107228.

[11] Creangă, C., & Dinu, L. P. (2024). ISDS-NLP at SemEval-2024 Task 10: Transformer networks for emotion recognition in conversations. *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval)*, 145–152.

[12] Patankar, S., & Phadke, M. (2025). A CNN-transformer framework for emotion recognition in code-mixed English–Hindi data. *Discover Artificial Intelligence*, *5*(1), 160.

[13] Zhao, X., Li, J., & Wang, H. (2025). Multi-target opinion words extraction using graph convolutional networks. *Neurocomputing*, 570, 127899.

[14] Zhu, H., Chen, Y., & Liu, Q. (2023). Sentence-guided grid tagging for aspect-sentiment quadruple extraction. *Information Sciences*, 626, 237–251.

[15] Zhang, Y., Wen, S., Zhu, Y., Li, Z., & Wang, X. (2025). Combining large language models with interpretable models for explainable aspect-based sentiment analysis in the medical domain. *Journal of King Saud University Computer and Information Sciences*, *37*(7), 1-19.

[16] Yang, J., Dai, F., Li, F., & Xue, Y. (2023, October). Span-based pair-wise aspect and opinion term joint extraction with contrastive learning. In *CCF International Conference on Natural Language Processing and Chinese Computing* (pp. 17-29). Cham: Springer Nature Switzerland.

[17] Deng, J., & Liu, Y. (2025). Research on sentiment analysis of online public opinion based on RoBERTa–BiLSTM–attention model. *Applied Sciences*, *15*(4), 2148.

[18] Lu, X., Chen, F., & Zhao, Y. (2024). Automatic construction of a Chinese review dataset for aspect sentiment triplet extraction via iterative weak supervision. *Data & Knowledge Engineering*, 149, 102215.

[19] Kalsum, T., Rauf, A., & Ali, N. (2023). A novel lightweight deep CNN model for human emotions recognition in diverse environments. *Multimedia Tools and Applications*, 82(13), 19751–19769.

[20] Venkatesan, S., Kumar, R., & Prasad, M. (2023). Human emotion detection using DeepFace and artificial intelligence. *Journal of Ambient Intelligence and Humanized Computing*, 14(5), 3123–3136.

[21] Zhu L, Li W, Shi Y, Guo K. SentiVec: learning sentiment-context vector via kernel optimization function for sentiment analysis. IEEE Trans Neural Netw Learn Syst. 2021;32(6):2561–72. https://doi.org/10.1109/tnnls.2020.3006531.

[22] Schouten K, van der Weijde O, Frasincar F, Dekker R. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. IEEE Trans Cybern. 2018;48(4):1263–75. https://doi.org/10.1109/tcyb.2017. 2688801.

[23] Ayyub K, Iqbal S, Munir EU, Nisar MW, Abbasi M. Exploring diverse features for sentiment quantification using machine learning algorithms. IEEE Access. 2020;8:142819–31. https://doi.org/10.1109/access.2020.3011202

[24] Bie Y, Yang Y. A multitask multiview neural network for end-to-end aspect-based sentiment analysis. Big Data Mining Analytics. 2021;4(3):195–207. https://doi.org/10.26599/bdma.2021.9020003

[25] Li N, Chow CY, Zhang JD. SEML: a semi-supervised multi-task learning framework for aspect-based sentiment analysis. IEEE Access. 2020;8:189287–97

[26] Sahay, S., Banoudha, A., & Sharma, R. (2014). On the use of ANFIS for predicting groundwater Levels in Hard Rock Area. Int. J. Res. Dev. Appl. Sci. Eng.

[27] On the use of ANFIS for predicting groundwater Levels in Hard Rock Area S Sahay, A Banoudha, R Sharma - Int. J. Res. Dev. Appl. Sci. Eng, 2014

[28] P. Pathak and M. M. Tripathi, "Hybrid machine learning models for post-COVID sentiment public health detection," Nanotechnology Perceptions, vol. 20, S11, pp. 104–119, 2024, doi:10.62441/nano-ntp.v20iS11.8.

[29] Singh, N., Singh, S. K., & Singh, R. (2025). Machine learning-based sentiment analysis for suicide prevention and mental health monitoring in educational institutions. *Journal of Neonatal Surgery*.