

# Detection Of Depression from Textual Data in Social Media Using Machine Learning

Malavika P<sup>1</sup>, Ashfaq K S<sup>1</sup> and Biju Skaria<sup>1</sup>

<sup>1</sup>Mar Athanasius College of Engineering, Kothamangalam, Kerala, India

*pmalavika18@gmail.com, ashfaqshafeek@gmail.com*

**Abstract.** Depression is a critical global mental health issue, with many cases remaining undiagnosed and untreated, especially in under-represented populations. As social media becomes a platform for emotional expression, it offers a valuable source of data for identifying mental health concerns. This study aims to develop a machine learning-based system to detect depression from social media posts by analyzing textual content. The proposed model utilizes N-gram and TF-IDF techniques for feature extraction and integrates classifiers such as Multilayer Perceptron (MLP), Random Forest (RF), and Decision Trees (DT) to improve detection accuracy. Trained on a dataset of over 10,000 depression-related posts, the system provides real-time detection of depressive symptoms and offers personalized support recommendations via a web-based interface developed using Flask. Previous research has shown that machine learning models, when combined with advanced feature extraction, can achieve over 90% accuracy in detecting depression. This project aims to create a scalable, accurate solution for identifying undiagnosed depression and providing mental health intervention by leveraging insights from social media activity, filling a crucial gap in timely mental health care.

## 1. INTRODUCTION

Depression is one of the leading causes of suicide worldwide. However, a significant percentage of depression cases go undiagnosed and untreated. According to the World Health Organization (WHO), depression is the most prevalent mental disorder, affecting over 300 million people globally. It is also responsible for more than two-thirds of suicides annually. Recent studies have demonstrated that messages posted by individuals with major depressive disorder on social media platforms can be analyzed to predict their likelihood of suffering from depression. Social media platforms, where people freely share their thoughts and feelings, offer a valuable source for monitoring health issues and trends. For instance, posts on platforms like Twitter and Facebook enable researchers to investigate multiple aspects of mental health. Specifically, studies have shown that tweets from individuals with major depressive disorder can be used to predict future episodes of depression.

A recent survey indicated that an increasing number of people, particularly teenagers and young adults, are turning to social media to express their feelings of depression. However, related work in this domain often relies on specific keywords like 'depression' and 'diagnose' when utilizing the data. In reality, social media users suffering from depression are unlikely to use such words directly due to the social stigma surrounding the condition. As a result, many affected individuals turn to less formal resources, such as social media, to seek support. This project aims to develop a machine learning model for detecting depression from textual data on social media. By analyzing posts without relying on explicit keywords, the project seeks to identify patterns and signals indicative of depression, providing a tool for early detection and intervention.

## 2. METHODOLOGY

### 2.1 Dataset Description

The dataset used for this project was sourced from Kaggle, specifically the "Depression Analysis" dataset by BB eye. It contains 10,314 records with three columns: text of the social media post or comment, a unique identifier (ID), and a class label indicating whether the content is depressive (1) or non-depressive (0) as shown in Figure 1. This dataset was chosen to train supervised learning models for detecting depression-related content without relying on explicit keywords.

	Unnamed: 0	message	label
0	106	just had a real good moment. i missssssssss hi...	0
1	217	is reading manga <a href="http://plurk.com/p/mzp1e">http://plurk.com/p/mzp1e</a>	0
2	220	@comeagainjen <a href="http://twitpic.com/2y2lx">http://twitpic.com/2y2lx</a> - <a href="http://...">http://...</a>	0
3	288	@lapcat Need to send 'em to my accountant tomo...	0
4	540	ADD ME ON MYSPACE!!! <a href="http://myspace.com/LookThunder">myspace.com/LookThunder</a>	0

**Figure 1** Depression Analysis Dataset

## 2.2 Data Pre-processing

Data preprocessing was essential to prepare the dataset for analysis. The following steps were performed:

### 2.2.1 Data Cleaning:

The dataset was checked for missing values, duplicates, and irrelevant features. It was found to be free of duplicates and missing values.

An irrelevant column named "Unnamed: 0" was dropped, ensuring only the required columns ("message" and "label") were retained.

### 2.2.2 Text Preprocessing Steps:

Removal of Extra Spaces, Mentions, and Hyperlinks: Regular expressions were used to clean the text.

Removal of Punctuation, Numbers, and Capitalization: Punctuation and numerical values were removed, and text was converted to lowercase.

### 2.2.3 Tokenization:

The processed text was split into individual words (tokens).

### 2.2.4 Stemming:

The Porter stemming algorithm was used to reduce words to their base forms.

### 2.2.5 Stopword Removal:

Common English stopwords, including custom Twitter-specific terms, were removed to reduce noise.

### 2.2.6 Update Dataset:

A new column ('processed\_messages') was added to store the cleaned and preprocessed text.

## 2.3 Feature Extraction

TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique is employed to convert a collection of text documents into a numerical representation that is suitable for machine learning tasks. This code utilizes the `TfidfVectorizer` class from `scikit-learn` to perform TF-IDF feature extraction on a dataset of processed messages.

Configuration parameters used:

`ngram_range = (1, 2)`: To capture both single words and two-word combinations.

`max_df = 0.75`: To filter out words appearing in more than 75% of the documents.

`min_df = 5`: To exclude terms that appear in fewer than 5 documents.

max\_features = 10,000: To limit the vocabulary size.

The resultant TF-IDF matrix was converted into a Pandas DataFrame for easy inspection and analysis.

## **2.4 Machine Learning Models**

Three machine learning algorithms were chosen for a comparative study based on their popularity and suitability for text classification tasks. We use Random Forest, Decision Tree, Multilayer Perceptron (MLP) algorithms for depression analysis.

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. It can be used for both Classification and Regression problems in ML. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Decision tree is a tree-like structure where each internal node tests on attribute, each branch corresponds to attribute value and each leaf node represents the final decision or prediction. The decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems.

A Multilayer Perceptron (MLP) is a type of artificial neural network used for supervised learning tasks, such as classification and regression. It consists of an input layer, one or more hidden layers, and an output layer, where each layer is made up of neurons. These neurons are fully connected between layers and use activation functions like ReLU or Sigmoid to learn complex, non-linear relationships in the data. The network is trained using backpropagation, where the weights are adjusted iteratively to minimize the error between predicted and actual outputs. MLPs are versatile and can be applied to a wide range of problems.

## **2.5 Model Training**

The dataset was divided into training (80%) and testing (20%) sets. The models were trained on the training set (features: X, labels: y). Each model was tuned and fit to the training data, followed by prediction on the test data.

### **2.5.1 MLP Model:**

The model consists of 100 neurons in the hidden layer. The maximum number of iterations is set to 300 for training, and the random\_state is set to 42 to ensure reproducibility. The classifier is fit to the training data (X\_train, y\_train), and predictions are made on the test data (X\_test).

The model achieved an accuracy of 99.10% on the training data.

### **2.5.2 Decision Tree:**

The model is initialized with a set random\_state of 42, which ensures that the results are reproducible by controlling the randomness in the training process. The classifier is fit to the training data (X\_train, y\_train), and predictions are made on the test data (X\_test).

The model achieved an accuracy of 99.49% on the training data.

### **2.5.3 Random Forest:**

A Random Forest Classifier, which is an ensemble learning method based on multiple decision trees is used to train the model. The model consists of 300 decision trees. The maximum depth of each tree is none and random\_state is set to 42 to ensure reproducibility. The classifier is fit to the training data (X\_train, y\_train), and predictions are made on the test data (X\_test).

The model achieved an accuracy of 99.49% on the training data.

## **2.6 Model Evaluation**

In machine learning, performance metrics refer to how well an algorithm performs depending on various criteria such as accuracy, precision, recall, and F1 score.

### 2.6.1 Accuracy

The percentage of correct test data predictions referred to as accuracy. It is easy to calculate by dividing the number of forecasts by the number of correct guesses.

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

### 2.6.2 Precision

The precision score is used to assess the model's correctly counting genuine positives among all positive predictions.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

### 2.6.3 Recall

The recall score used to assess the model's performance in terms of accurately counting true positives among all actual positive values.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

### 2.6.4 F1 Score

The F1-score is the harmonic mean of precision and recall score, and utilized as a metric in situations when choosing either precision or recall score can result in a model with excessive false positives or false negatives.

$$\text{F1 Score} = \frac{2 (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}}$$

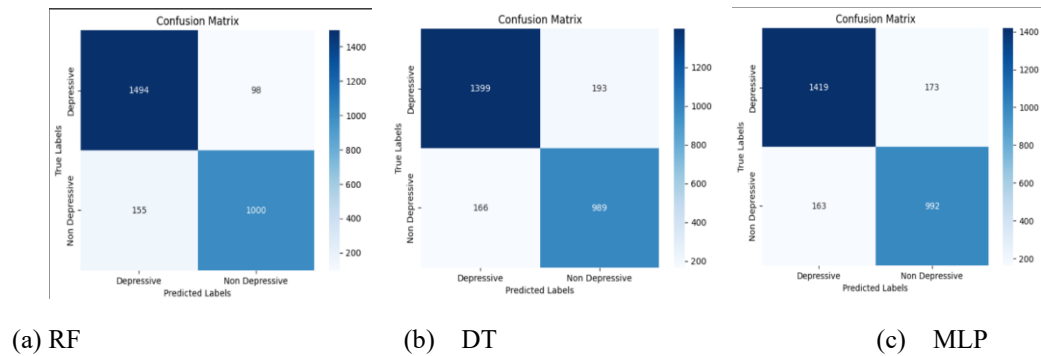
## 3. RESULTS AND DISCUSSIONS

We apply the selected algorithms to our dataset. The algorithms are then compared using various performance metrics. We can see from Table 1 that the Random Forest algorithm has the highest accuracy of any algorithm. After Random Forest, Multilayer Perceptron followed by Decision Tree.

**Table 1** Evaluation of algorithms

Algorithm	Accuracy	Precision	Recall	F1-score
Random Forest Classifier	91	91	91	91
Decision Tree	87	87	87	87
Multilayer Perceptron	88	87	88	87

However, accuracy does not always provide good performance metrics to compare algorithms, other metrics are also considered. Random Forest has 91% accuracy, precision, recall, and F1 score. Decision Tree has 87% accuracy, precision, recall, and F1 score. Multilayer Perceptron has 88% accuracy, precision, recall, and F1 score. From the result, we can say that Random Forest gives us the best prediction for our dataset. From the confusion matrix, as shown in Figure 1, we can also say that Random Forest gives us the best prediction, and Decision Tree gives us the poorest prediction in this case. As a result, we can conclude that for our chosen dataset, Random Forest is the best classification algorithm. Overall, RF is better as RF has fewer wrong predictions compared to DT and MLP and RF gives more correct predictions.



**Figure 2** Confusion matrix of different algorithms a) RF, b) DT and c) MLP

## 4. CONCLUSION

The detection of depression from textual data in social media has become an increasingly significant problem due to its potential to offer timely interventions. Depression, if undetected, can lead to severe complications, making early detection essential. Detecting depressive behaviours through text-based analysis, using machine learning models, is crucial for improving mental health care.

After extracting relevant features from social media texts using TF-IDF text vectorization technique, we applied various machine learning classifiers to our dataset. Feature extraction played a key role in enhancing the performance of these classifiers.

After analyzing the data, we discovered that Random Forest consistently outperformed other classification algorithms, delivering an accuracy score of 91 percent. Multilayer Perceptron (MLP) and Decision Tree (DT) achieved accuracies of 88 percent and 87 percent, respectively. Random Forest's superior performance, combined with its relatively low computational complexity, makes it an ideal choice for depression detection in social media text, contributing to the development of an efficient automatic prediction system.

## REFERENCES

- [1] Tadesse MM, Lin H, Xu B, Yang L. 2019. Detection of depression-related posts in reddit social media forum. *Ieee Access*. 7:44883-93.
- [2] Chiong R, Budhi GS, Dhakal S, Chiong F. 2021. A textual-based featuring approach for depression detection using machine learning classifiers and social media texts. *Computers in Biology and Medicine*. 135:104499.
- [3] Islam MR, Kabir MA, Ahmed A, Kamal AR, Wang H, Ulhaq A. 2018. Depression detection from social network data using machine learning techniques. *Health information science and systems*. 6:1-2.
- [4] Eye BB 2020 Depression Analysis. 1 edn., Kaggle [Depression Analysis](#).
- [5] Mishra V, Garg T (2018) A systematic study on predicting depression using text analytic. *Journal of Fundamental and Applied Sciences* 10 (2):293-307. doi:10.4314/jfas.v10i2.21
- [6] Edo-Osagie O, De La Iglesia B, Lake I, Edeghere O (2020) A scoping review of the use of Twitter for public health research. *Computers in Biology and Medicine* 122:103770. doi:10.1016/j.combiomed.2020.103770.
- [7] S. Paul, S. K. Jandhyala, and T. Basu, "Early detection of signs of anorexia and depression over social media using effective machine learning frameworks," in *Proc. CLEF*, Aug. 2018, pp. 1–9.
- [8] Deng X, Li Y, Weng J, Zhang J (2019) Feature selection for text classification: A review. *Multimedia Tools and Applications* 78 (3):3797-3816. doi:10.1007/s11042-018-6083-5
- [9] Breiman L (2001) Random forests. *Machine Learning* 45 (1):5-32. doi:10.1023/a:1010933404324
- [10] Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition. *IEEE Access* 7:158820-158846. doi:10.1109/access.2019.2945545.