Title

Comparing Traditional Machine Learning and Large Language Models: An Application to Mental Health Text Classification

Author

Yang, Zhangfeifan

Publication Date

2024

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Comparing Traditional Machine Learning and Large Language Models: An Application to Mental Health Text Classification

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Applied Statistics and Data Science

by

Zhangfeifan Yang

2024

ABSTRACT OF THE THESIS

Comparing Traditional Machine Learning and Large Language Models:
An Application to Mental Health Text Classification

by

Zhangfeifan Yang
Master of Applied Statistics and Data Science
University of California, Los Angeles, 2024
Professor Yingnian Wu, Chair

Mental health conditions profoundly affect individuals worldwide, yet their early detection and diagnosis remain complex. This thesis investigates the application of machine learning and large language models (LLMs) for classifying mental health conditions based on textual data. Traditional models, including Logistic Regression, Support Vector Machines (SVM), and Random Forest, were evaluated alongside the fine-tuned Llama 3.1-8B LLM. Preprocessing steps, such as text cleaning and vectorization using Term Frequency-Inverse Document Frequency (TF-IDF), facilitated effective feature extraction. The Llama 3.1-8B achieved superior performance, with an accuracy of 86%, compared to 76% for traditional models, while also excelling in capturing nuanced linguistic patterns. However, traditional models demonstrated advantages in interpretability and computational efficiency. This study underscores the potential of LLMs in advancing automated mental health assessments while emphasizing the importance of ethical considerations and model transparency in real-world applications.

The thesis of Zhangfeifan Yang is approved.

Nicolas Christou

Frederic R. Paik Schoenberg

Yingnian Wu, Committee Chair

University of California, Los Angeles

2024

To my family and friends,

whose unconditional support, encouragement, and belief in me

have made this journey not only possible but meaningful,

and to my beloved cat, Huihui,

who has been by my side for 12 years.

Despite his recent health challenges, Huihui has given me endless love,

emotional and mental support, and companionship.

I wish him a happy and peaceful life ahead.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

Mental health is a critical component of overall well-being, affecting millions worldwide. However, the identification and diagnosis of mental health conditions often rely on subjective and time-intensive methods such as self-reporting and clinical interviews, which can be inconsistent and prone to variability. With the widespread adoption of digital platforms and social media, natural language processing (NLP) and machine learning (ML) offer promising opportunities for improving early detection and monitoring of mental health conditions [CD20, GYK17].

This study aims to explore and compare the performance of traditional machine learning models and large language models (LLMs) for classifying mental health conditions using textual data. The research holds significant potential to address critical gaps in mental health assessment by developing scalable tools that serve as early warning systems for timely intervention. Additionally, the findings contribute to the understanding of linguistic markers associated with mental health conditions, offering valuable insights for clinical applications and research [MKC21, TLI23].

The research is guided by two key questions:

1. How do traditional machine learning models, such as Logistic Regression, Support Vector Machines (SVM), and Random Forest, perform compared to large language models in classifying mental health conditions?

2. What linguistic features and patterns distinguish different mental health conditions?

To address these questions, this study develops and evaluates multiple text classification models, focusing on their performance and comparative strengths.

The thesis is structured as follows. Chapter 2 discusses the dataset used, including its structure, preprocessing methods, and ethical considerations. Chapter 3 provides an overview of foundational concepts in text classification, including traditional machine learning models and LLMs. Chapter 4 describes the methodology, detailing the implementation of models and evaluation metrics. Chapter 5 presents the results, highlighting classification accuracy, F1-scores, and confusion matrices for each model. Chapter 6 interprets these findings, discussing the comparative performance of traditional models and LLMs, along with their implications for interpretability, deployment, and ethics. Finally, Chapter 7 concludes with a concise summary of the study's findings, emphasizing key takeaways and outlining directions for future research.

CHAPTER 2

Data

2.1 Dataset Overview

The dataset used in this study, titled "Sentiment Analysis for Mental Health" comprises 53,043 entries. Each entry consists of a textual statement related to mental health, a unique identifier, and a corresponding mental health condition label.

2.1.1 Dataset Structure

The dataset includes the following columns:

- `unique_id`: A unique identifier for each entry.
- `statement`: The textual content of the statement.
- `status`: The mental health condition label (e.g., Normal, Depression, Suicidal).

Out of the 53,043 entries, 362 entries in the `statement` column contain null values. These null values are addressed during the preprocessing stage to ensure the quality of the analysis.

2.2 Distribution of Mental Health Conditions

The dataset exhibits an imbalanced distribution of mental health conditions. Figure 2.1 illustrates this distribution, highlighting the dominance of categories such as "Normal" and "Depression" and the underrepresentation of "Personality Disorder".

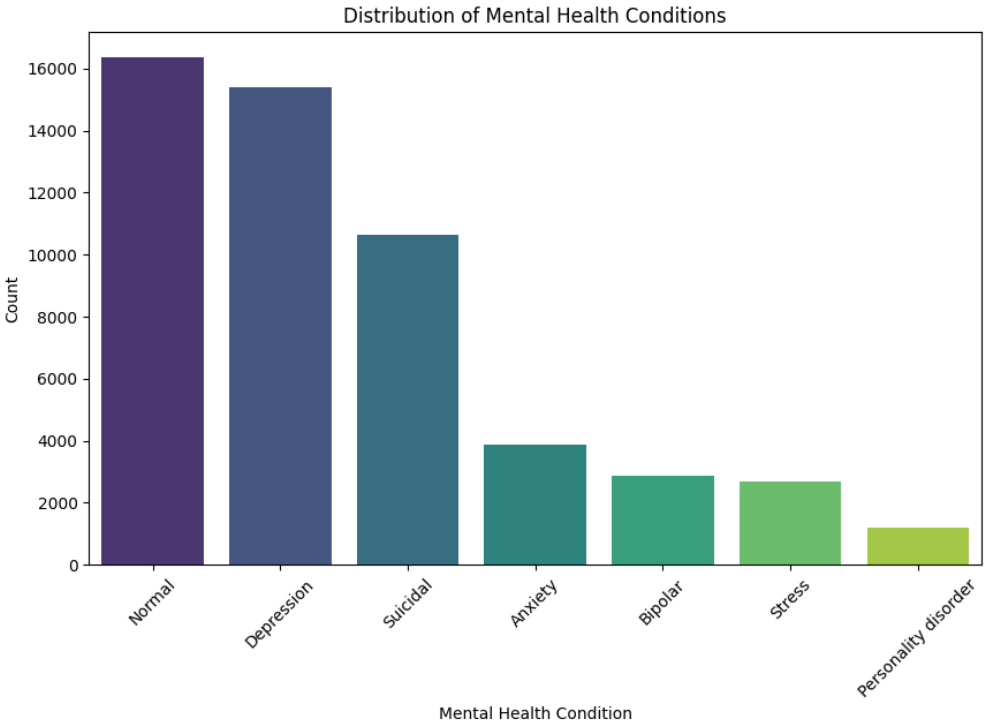

Figure 2.1: Distribution of Mental Health Conditions

This imbalance necessitates specific techniques, such as class weighting, to ensure effective model training.

2.3 Text Analysis

2.3.1 Word Cloud

A word cloud analysis was conducted to visualize the most frequently occurring words in the dataset, with stop words removed. Figure 2.2 provides a graphical representation of the common themes and terms.

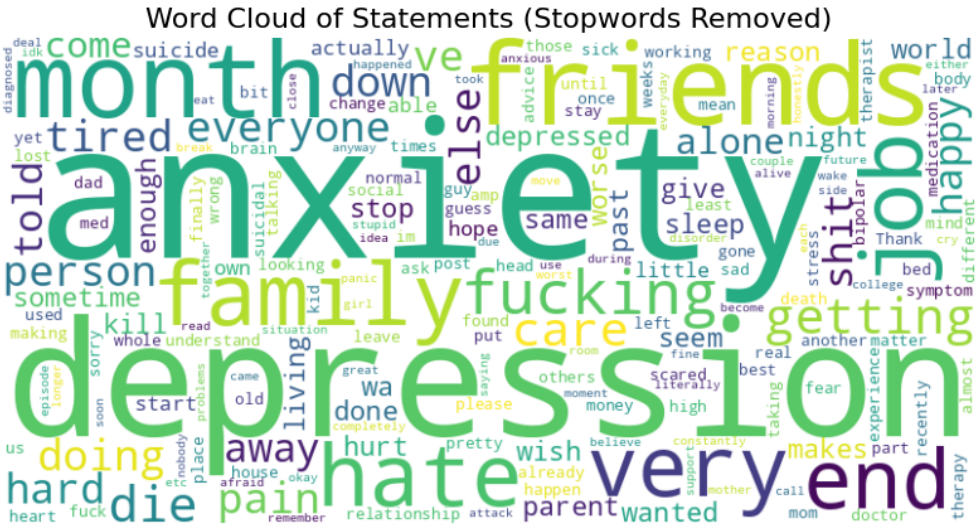

Figure 2.2: Word Cloud of Statements (Stopwords Removed)

Prominent terms such as “anxiety”, “depression”, and “friends” highlight key emotional and social themes present in the dataset. Words like “family” and “hate” further reflect interpersonal and affective dimensions of the statements. This visualization provides an intuitive understanding of the dataset’s central themes, emphasizing the emotional and relational challenges faced by individuals.

2.4 Data Preprocessing

The data preprocessing phase involves several key steps:

2.4.1 Handling Missing Values

Entries with null values in the `statement` column were removed to ensure data consistency and reliability for analysis.

2.4.2 Text Cleaning

The textual data underwent several cleaning steps:

1. Conversion to lowercase to standardize text.
2. Removal of special characters and punctuation.
3. Elimination of numbers unless deemed semantically significant.
4. Tokenization into individual words.
5. Removal of common stop words using an expanded stop word list.

2.4.3 Stop Words

During preprocessing stage, stop words—frequent words with little semantic meaning—were removed to improve the accuracy of the analysis. The stop words list included standard English stop words, contractions, auxiliary verbs, prepositions, and domain-specific terms regarded unhelpful for classifying mental health conditions. The complete list is provided in Appendix A.

Standard stop words, such as "the", "and", and "to", were included to reduce redundancy. Contractions like "i'm," "don't," and "can't" were added to account for variations in informal language. Words related to time, such as "day", "week", and "year", were excluded, as they were not relevant to sentiment or thematic classification.

Emotion-related terms like "feel", "felt", and "feelings" were removed due to their high frequency across categories, which reduced their ability to discriminate. Personal pronouns and possessives, such as "I", "me", "my", and "their", were excluded to shift the focus from the subject to the content of the statements. Common auxiliary verbs, such as "can", "will", and "should", and prepositions like "at", "on", and "with" were removed to minimize

syntactic noise. Additionally, colloquialisms like “uh”, “yeah”, and “hmm” were excluded to reduce non-informative conversational elements.

2.5 Data Splitting

To evaluate model performance, the dataset was split into training and testing subsets:

- 80% of the data was allocated for training.
- 20% was reserved for testing.

Stratified sampling ensured that the distribution of mental health conditions in the training and testing sets mirrored the original dataset.

2.6 Ethical Considerations

This study addresses several ethical considerations: The dataset was anonymized to safeguard privacy, with unique identifiers replacing personal information. Bias mitigation strategies, such as class weighting, were implemented to address imbalanced data distributions. All data use assumes appropriate consent for research purposes. The models developed in this study are intended solely for research and should not be applied for clinical diagnoses without further validation.

CHAPTER 3

Theoretical Framework

3.1 Text Classification in NLP

Text classification is a supervised machine learning task where the objective is to categorize textual data into predefined classes. In the context of mental health, this involves analyzing text, such as social media posts or online forum discussions, to identify conditions like *anxiety*, *depression*, and *bipolar disorder*. Advances in Natural Language Processing (NLP) have significantly improved text classification through traditional machine learning models and large language models (LLMs) [MKC21].

3.2 Traditional Machine Learning Models for Text Classification

Traditional machine learning models transform text into numerical representations, allowing classification algorithms to learn patterns from the data. The theoretical foundations of Logistic Regression, Support Vector Machines (SVM), and Random Forest, which are employed in this study, are discussed below.

3.2.1 Logistic Regression

Logistic Regression is a linear model used for binary and multi-class classification. It predicts the probability $P(y)$ of a given class y for an input feature vector $X = \{x_1, x_2, \ldots, x_n\}$ using

the logistic (sigmoid) function:

$$P(y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n)}}$$

where β_0 is the intercept, and $\beta_1, \ldots, \beta_n$ are the model coefficients. For multi-class classification, the softmax function generalizes logistic regression by computing the probability $P(y = c|X)$ for each class c as:

$$P(y = c|X) = \frac{e^{\beta_c \cdot X}}{\sum_{k=1}^{C} e^{\beta_k \cdot X}}$$

where C is the total number of classes, and β_c represents the coefficient vector for class c. The model is trained by minimizing the cross-entropy loss:

$$L(\beta) = -\sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

where N is the number of samples, y_i is the true label, and $\hat{y}_i$ is the predicted probability. Logistic Regression is computationally efficient and works well for linearly separable data, but its performance may degrade with complex, non-linear relationships.

3.2.2 Support Vector Machines (SVM)

Support Vector Machines (SVM) aim to find the optimal hyperplane that separates data points from different classes in a high-dimensional space. For a dataset $\{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in R^d$ are feature vectors and $y_i \in \{-1, 1\}$ are class labels, the objective is to minimize the weight vector w while maximizing the margin between classes:

$$\min_{w,b} \frac{1}{2}||w||^2$$

subject to:

$$y_i(w \cdot x_i + b) \geq 1, \quad \forall i$$

For non-linear separable data, the kernel trick maps data into a higher-dimensional space via a kernel function $K(x_i, x_j)$, such as the Radial Basis Function (RBF):

$$K(x_i, x_j) = \exp(-\gamma||x_i - x_j||^2)$$

In cases where some misclassification is allowed, slack variables ξ_i are introduced, resulting in the soft-margin SVM:

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i$$

subject to:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

SVM is particularly effective for high-dimensional data and can handle non-linear decision boundaries through appropriate kernel functions.

3.2.3 Random Forest

Random Forest is an ensemble learning algorithm that combines multiple decision trees to improve classification performance and reduce overfitting. Each tree is trained on a bootstrap sample of the data, with random subsets of features considered at each split. A single decision tree partitions the data at each node by optimizing a criterion such as Gini impurity:

$$Gini(D) = 1 - \sum_{i=1}^{C} p_i^2$$

where p_i is the proportion of samples belonging to class i, and C is the total number of classes. Alternatively, information gain, derived from entropy, can also be used:

$$Entropy(D) = -\sum_{i=1}^{C} p_i \log_2(p_i)$$

The aggregated prediction of the Random Forest is obtained through majority voting for classification tasks:

$$\hat{y} = \text{mode}\{y_1, y_2, \ldots, y_T\}$$

where y_t represents the prediction of the t-th tree, and T is the total number of trees in the forest. Random Forest leverages the diversity of its individual trees to enhance robustness against noise and overfitting, making it suitable for handling high-dimensional and imbalanced datasets[MKC21].

3.3 Text Representation Techniques

3.3.1 TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF is a statistical measure used to evaluate the importance of words in a document relative to the entire corpus. It is computed as:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t)$$

where:

$$\text{TF}(t, d) = \frac{\text{Frequency of } t \text{ in } d}{\text{Total words in } d}, \quad \text{IDF}(t) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right)$$

Here, N is the total number of documents, and $|\{d \in D : t \in d\}|$ is the number of documents containing term t.

3.3.2 Word Embeddings

Word embeddings are dense vector representations of words that capture their semantic relationships. Popular models such as Word2Vec and GloVe generate embeddings that encode semantic meaning [MKC21].

3.4 Large Language Models (LLMs)

3.4.1 Transformer Architecture

The transformer architecture introduced by Vaswani et al. (2017) revolutionized NLP by introducing a self-attention mechanism:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where Q, K, and V are the query, key, and value matrices, and d_k is the dimensionality of the key vectors. Unlike RNNs, transformers allow parallel processing of input sequences, enabling efficient training and improved performance [Che21].

3.4.2 Fine-Tuning of LLMs

Large Language Models (LLMs) such as BERT and LLaMA are pre-trained on extensive datasets, enabling them to develop a general understanding of language. Fine-tuning is the process of adapting these pre-trained models to specific downstream tasks, such as text classification. This is achieved by optimizing the model parameters for the target task using a task-specific loss function. For classification problems, the cross-entropy loss is typically used:

$$L(\theta) = -\sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

where:

- θ represents the model parameters,
- y_i is the true label for the i-th training sample,
- $\hat{y}_i$ is the predicted probability of the corresponding class, and
- N is the total number of training samples.

While traditional fine-tuning involves updating all parameters of the model, this approach becomes computationally expensive for LLMs with billions of parameters. To address this challenge, parameter-efficient fine-tuning methods such as Low-Rank Adaptation (LoRA) have been developed[TLI23].

3.4.2.1 Low-Rank Adaptation (LoRA)

LoRA is a parameter-efficient fine-tuning technique designed for large-scale models. Instead of updating all model parameters, LoRA introduces trainable low-rank matrices into the attention layers of the model while freezing the pre-trained parameters. This significantly reduces the number of trainable parameters, resulting in lower memory and computational requirements.

In the transformer architecture, the self-attention mechanism uses query (Q), key (K), and value (V) projections, which are parameterized by weight matrices. LoRA introduces low-rank matrices A and B to these weight matrices, allowing the model to learn task-specific adaptations without modifying the pre-trained weights:

$$W' = W + AB$$

where:

- W is the pre-trained weight matrix,
- A is a low-rank matrix (rank r),
- B is another low-rank matrix, and
- W' is the adapted weight matrix used during fine-tuning.

The key advantages of LoRA include:

- **Parameter Efficiency:** Only the low-rank matrices A and B are updated, significantly reducing the number of trainable parameters.
- **Preservation of Pre-Trained Knowledge:** The original weights W remain unchanged, ensuring that the model retains its general language understanding.
- **Computational Efficiency:** By limiting parameter updates, LoRA reduces the computational burden of fine-tuning, making it feasible for resource-constrained environments.

3.4.2.2 Advantages of LoRA in Mental Health Classification

LoRA is particularly well-suited for tasks like mental health text classification due to the following reasons:

- **Scalability:** The low memory footprint allows the fine-tuning of large models like LLaMA 3.1-8B on moderate hardware setups.
- **Task-Specific Adaptation:** LoRA enables the model to focus on learning the nuances of mental health-related language while preserving its broader language understanding.
- **Flexibility:** The rank r of the low-rank matrices can be adjusted to balance the trade-off between computational efficiency and task performance.

3.4.2.3 Comparison with Traditional Fine-Tuning

In traditional fine-tuning, all parameters of the model are updated during training, resulting in a high computational cost and storage requirement. In contrast, LoRA focuses only on a small subset of parameters (the low-rank matrices), reducing these costs while maintaining competitive performance.

3.5 Performance Evaluation Metrics

The effectiveness of text classification models is evaluated using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. These metrics account for imbalanced datasets and provide insights into both overall and class-specific performance [MKC21].

CHAPTER 4

Methodology

4.1 Machine Learning Approach

4.1.1 Logistic Regression

Logistic Regression was employed as a baseline model considering its interpretability and efficiency in classification tasks. Key parameters were configured to optimize performance for the multi-class classification of mental health conditions.

The model was trained using the one-vs-rest strategy, where separate binary classifiers were built for each class. This approach ensures effective handling of multi-class problems by converting them into multiple binary classification tasks. The `max_iter` parameter was set to 1000 to ensure convergence of the optimization algorithm, particularly for the high-dimensional TF-IDF vectorized feature space.

To reduce the impact of correlated features and improve model stability, the `solver` parameter was set to its default, `lbfgs` (Limited-memory Broyden–Fletcher–Goldfarb–Shanno). This solver is well-suited for multi-class problems and supports regularization. Although the default regularization strength (`C=1.0`) was used, future studies could explore hyperparameter tuning to optimize this setting further.

4.1.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) classifier was chosen for its robustness in high-dimensional spaces and ability to handle linearly separable data efficiently. For this study, a linear kernel (`kernel='linear'`) was employed to achieve computational efficiency and compatibility with text classification tasks.

Class imbalance was addressed using the `class_weight='balanced'` parameter. This parameter dynamically adjusted the weight of each class, ensuring that underrepresented classes received appropriate consideration during model training.

Key implementation details included vectorizing the dataset using Term Frequency-Inverse Document Frequency (TF-IDF) with a dimensionality constrained to 5000 features (`max_features=5000`). The SVM model was trained on an 80-20 train-test split, and the linear kernel efficiently separated classes within the high-dimensional feature space.

4.1.3 Random Forest

Random Forest was employed to model potential non-linear relationships and capture feature interactions within the dataset. This ensemble learning technique combines predictions from multiple decision trees to enhance classification accuracy and reduce overfitting risks.

The implementation consisted of 100 decision trees (`n_estimators=100`), with each tree trained on a bootstrap sample of the training data. Bootstrap sampling introduced randomness into the model, increasing its robustness. To address the class imbalance present in the dataset, the `class_weight='balanced'` parameter was utilized, ensuring that minority classes were adequately represented during training.

As with other models, TF-IDF was used to vectorize the dataset, retaining the top 5000 features (`max_features=5000`). The Random Forest model was trained with a fixed random state to ensure reproducibility, and feature importance scores were extracted to identify key predictors for each mental health condition.

4.2 Large Language Model Approach

4.2.1 Model Selection and Overview

The Llama 3.1-8B model, a state-of-the-art large language model developed by Meta AI, was selected for its advanced language understanding capabilities. With approximately 8 billion parameters, the model balances computational complexity and representational power, making it suitable for fine-tuning on specialized tasks such as mental health text classification. The fine-tuning process leveraged Low-Rank Adaptation (LoRA), a parameter-efficient technique that optimizes task-specific performance without requiring full retraining of the base model.

4.2.2 Prompt Engineering

We designed a specific prompt structure to guide the model in the classification task. Each sample was transformed into a JSON format with the following fields:

- `instruction`: Possible mental status: Anxiety, Normal, Depression, Suicidal, Stress, Bipolar, Personality disorder
- `input`: Classify the following text into one of these mental statuses:
- `output`: The most possible mental status for the given text is ####

This structured prompt design aimed to consistently guide the model in performing the classification task.

4.2.2.1 Fine-Tuning with LoRA

The fine-tuning of the Llama 3.1-8B model was conducted using the Low-Rank Adaptation (LoRA) technique. LoRA offers a computationally efficient approach to fine-tuning large

language models by introducing trainable low-rank matrices into the attention layers of the model while keeping the pre-trained parameters frozen. This method significantly reduces the number of parameters requiring updates, minimizing both storage requirements and computational costs.

The fine-tuning process for this study involved the following steps:

1. **Initialization:** The weights of the LoRA parameters were initialized, and the original parameters of the Llama 3.1-8B model were frozen. This ensured that the fine-tuning process focused exclusively on the newly introduced low-rank matrices without altering the model's pre-trained knowledge.

2. **Parameter Adjustment:** Only the LoRA parameters were updated during training, allowing the model to adapt specifically to the mental health text classification task while preserving the general language understanding provided by the base model.

3. **Rank Selection:** Several rank values ($r = 8, 16, 32$) were explored to balance the model's capacity to learn task-specific features with computational efficiency. Lower rank values reduce the number of trainable parameters but may limit model flexibility, while higher values increase adaptability at the cost of additional computation.

4. **Prompt Utilization:** Structured prompts, as detailed in Section 4.2.2, were employed during fine-tuning to guide the learning process. These prompts ensured that the model adapted effectively to the specific requirements of the classification task.

LoRA's efficiency and adaptability make it particularly well-suited for scenarios where computational resources are constrained. By leveraging this technique, the Llama 3.1-8B model was fine-tuned to achieve high performance on the mental health text classification task without incurring the substantial resource demands of traditional fine-tuning methods.

4.3 Evaluation Framework

4.3.1 Performance Metrics

The models were evaluated using the following metrics, ensuring a comprehensive assessment of their performance:

- **Accuracy**: The proportion of correct predictions across all test samples.
- **Precision**: The ratio of true positives to all predicted positives, indicating the model's ability to avoid false positives.
- **Recall**: The ratio of true positives to all actual positives, reflecting the model's sensitivity to each class.
- **F1-Score**: The harmonic mean of precision and recall, providing a balanced measure of performance for imbalanced datasets.

4.3.2 Computational Considerations

The computational efficiency of each model was qualitatively assessed. Traditional models, such as Logistic Regression and SVM, required minimal computational resources for both training and inference. On the contrary, the fine-tuning and inference of the Llama 3.1-8B model required significantly higher resources, which makes it less feasible for deployment in resource-constrained environments.

4.3.3 Model Comparison Strategy

A comparative analysis was conducted to evaluate the relative strengths and weaknesses of traditional machine learning models and the Llama 3.1-8B model. This analysis considered:

- Performance metrics, as outlined above, to assess the effectiveness of each model in

classifying mental health conditions.

- Class-specific performance to identify categories where models struggled or excelled.
- Computational costs to determine the feasibility of real-world deployment.
- Interpretability, with traditional models offering insights into feature importance and LLMs providing context-sensitive classifications.

CHAPTER 5

Results

5.1 Traditional Machine Learning Model Performance

5.1.1 Logistic Regression

The Logistic Regression model achieved an overall accuracy of 76%. Table 5.1 provides the classification metrics, including precision, recall, and F1-scores for each class. As observed, the model performed exceptionally well in the "Normal" category, achieving a precision of 84% and a recall of 95%. However, the performance was suboptimal for the "Stress" and "Personality Disorder" categories, where the recall values were 44% and 42%, respectively. Figure 5.1 illustrates the confusion matrix, highlighting the distribution of correct and incorrect classifications.

Class	Precision	Recall	F1-score	Support
Anxiety	0.83	0.75	0.79	755
Bipolar	0.88	0.68	0.77	527
Depression	0.67	0.73	0.70	3016
Normal	0.84	0.95	0.89	3308
Personality Disorder	0.86	0.42	0.56	237
Stress	0.73	0.44	0.55	536
Suicidal	0.69	0.66	0.67	2158
Accuracy	0.76 (10,537 samples)			
Macro Avg	0.79	0.66	0.70	10,537
Weighted Avg	0.76	0.76	0.75	10,537

Table 5.1: Classification Report for Logistic Regression

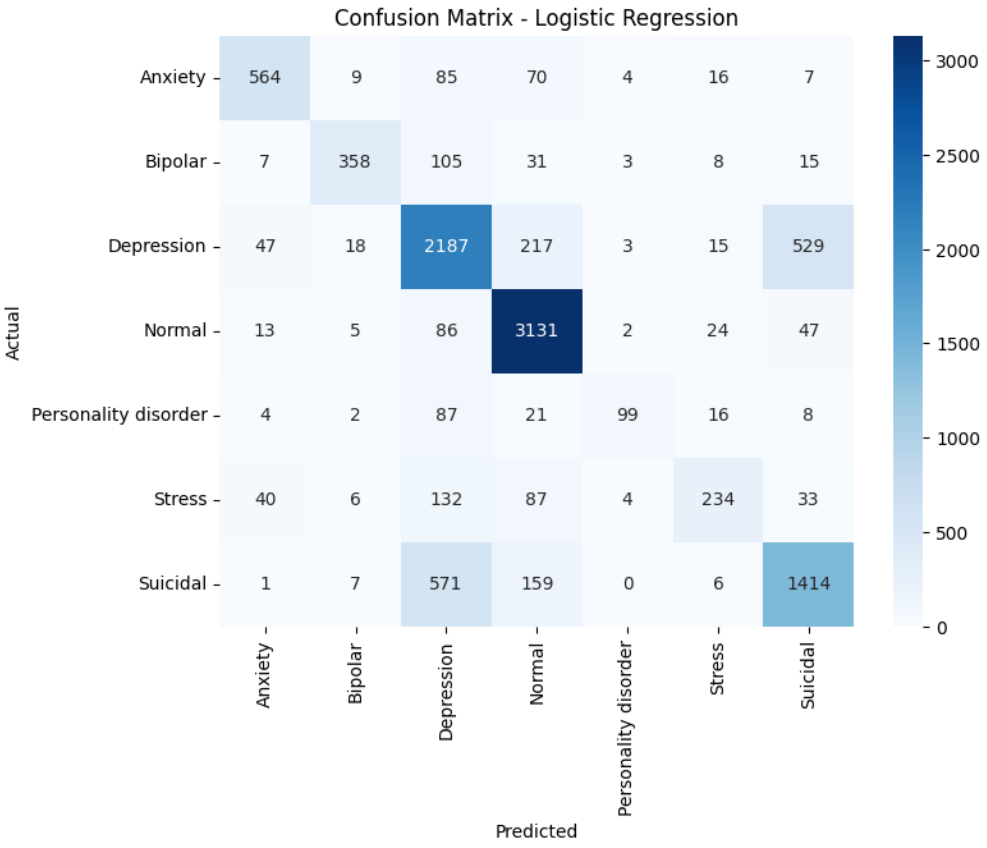

Figure 5.1: Confusion Matrix for Logistic Regression

5.1.2 SVM

The SVM model achieved an identical accuracy of 76%. Table 5.2 outlines the classification metrics. Compared to Logistic Regression, SVM demonstrated a slight improvement in recall for the "Anxiety" and "Stress" categories, achieving 84% and 66%, respectively. The confusion matrix in Figure 5.2 shows fewer misclassifications for the "Normal" category, emphasizing the model's robustness for majority classes.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Anxiety	0.74	0.84	0.79	755
Bipolar	0.74	0.82	0.78	527
Depression	0.76	0.61	0.68	3016
Normal	0.90	0.91	0.90	3308
Personality Disorder	0.67	0.68	0.68	237
Stress	0.53	0.66	0.59	536
Suicidal	0.66	0.75	0.70	2158
Accuracy	0.76 (10,537 samples)			
Macro Avg	0.72	0.75	0.73	10,537
Weighted Avg	0.77	0.76	0.76	10,537

Table 5.2: Classification Report for SVM

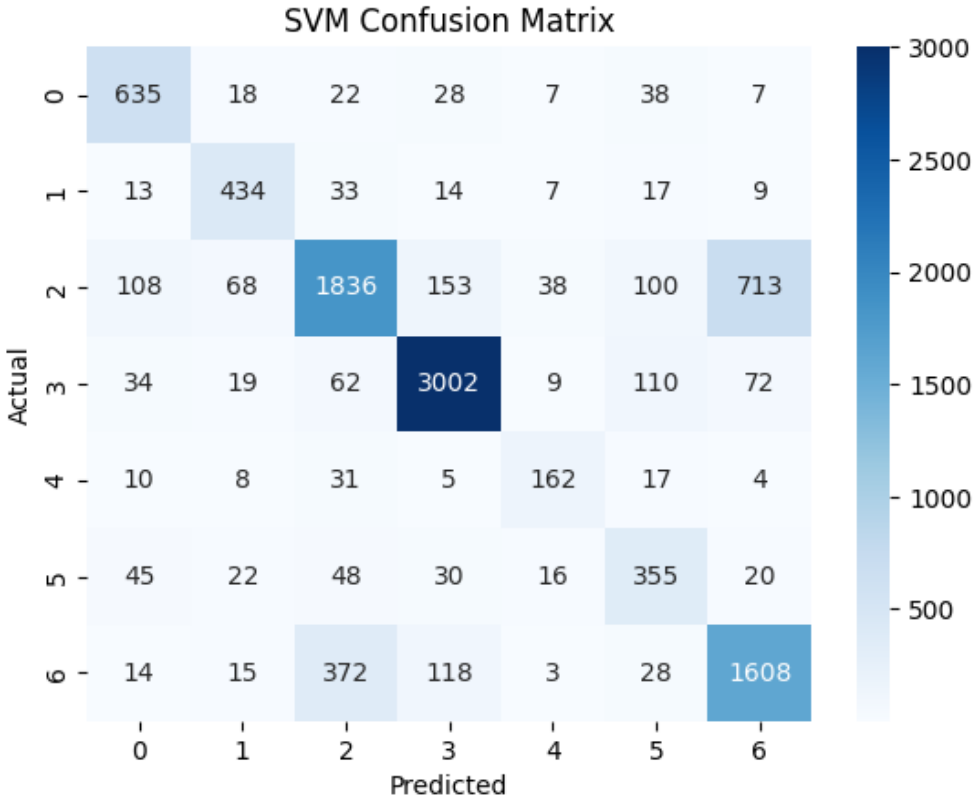

Figure 5.2: Confusion Matrix for SVM

5.1.3 Random Forest

The Random Forest model achieved an accuracy of 74%. As detailed in Table 5.3, the model performed well for the "Normal" class with a precision of 83% and recall of 94%. However, it underperformed for minority classes such as "Stress" and "Suicidal," where the recall values were 32% and 58%, respectively. Figure 5.3 highlights these discrepancies through the confusion matrix.

Class	Precision	Recall	F1-score	Support
Anxiety	0.83	0.72	0.77	755
Bipolar	0.93	0.66	0.77	527
Depression	0.62	0.75	0.68	3016
Normal	0.83	0.94	0.89	3308
Personality Disorder	0.98	0.41	0.58	237
Stress	0.90	0.32	0.47	536
Suicidal	0.68	0.58	0.63	2158
Accuracy	0.74 (10,537 samples)			
Macro Avg	0.82	0.63	0.68	10,537
Weighted Avg	0.75	0.74	0.73	10,537

Table 5.3: Classification Report for Random Forest

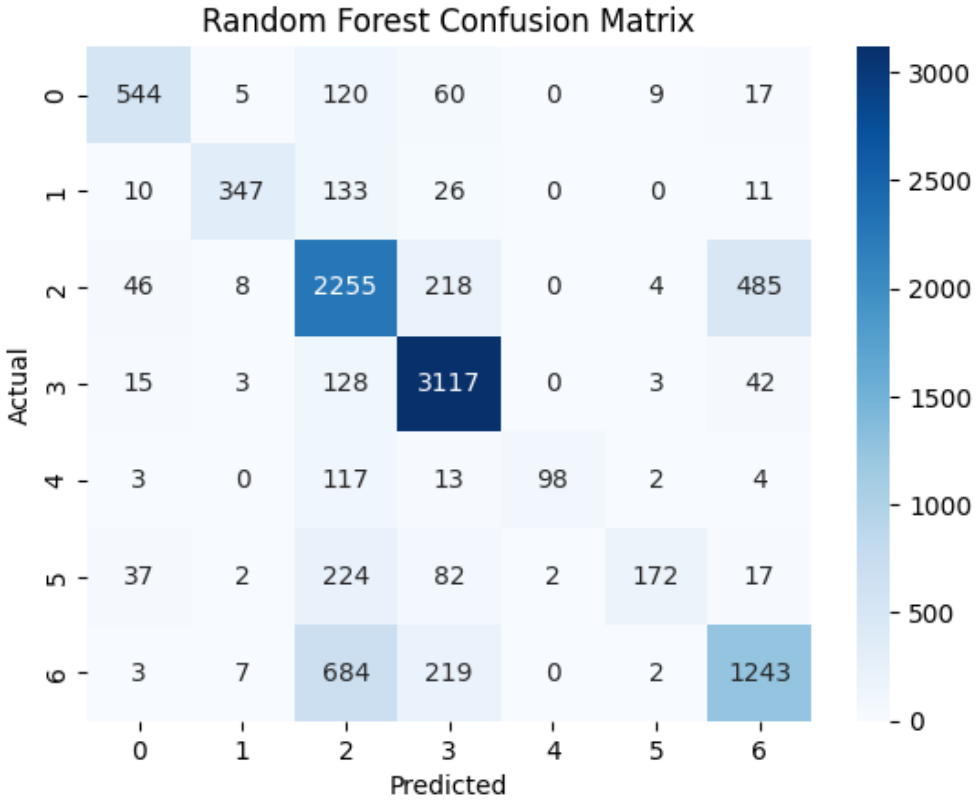

Figure 5.3: Confusion Matrix for Random Forest

5.2 LLM Performance

The fine-tuned Llama 3.1-8B model outperformed traditional models, achieving an accuracy of 86%. As shown in Table 5.4, it excelled across all categories, particularly in "Anxiety" and "Normal," with F1-scores of 91% and 96%, respectively. The confusion matrix in Figure 5.4 demonstrates its balanced performance across all categories.

Class	Precision	Recall	F1-score	Support
Anxiety	0.88	0.95	0.91	768
Bipolar	0.91	0.92	0.91	556
Depression	0.84	0.75	0.79	3081
Normal	0.98	0.95	0.96	3269
Personality Disorder	0.86	0.83	0.84	215
Stress	0.74	0.88	0.80	517
Suicidal	0.72	0.80	0.76	2131
Accuracy	0.86 (10,537 samples)			
Macro Avg	0.74	0.89	0.75	10,537
Weighted Avg	0.86	0.86	0.86	10,537

Table 5.4: Classification Report for Llama 3.1-8B

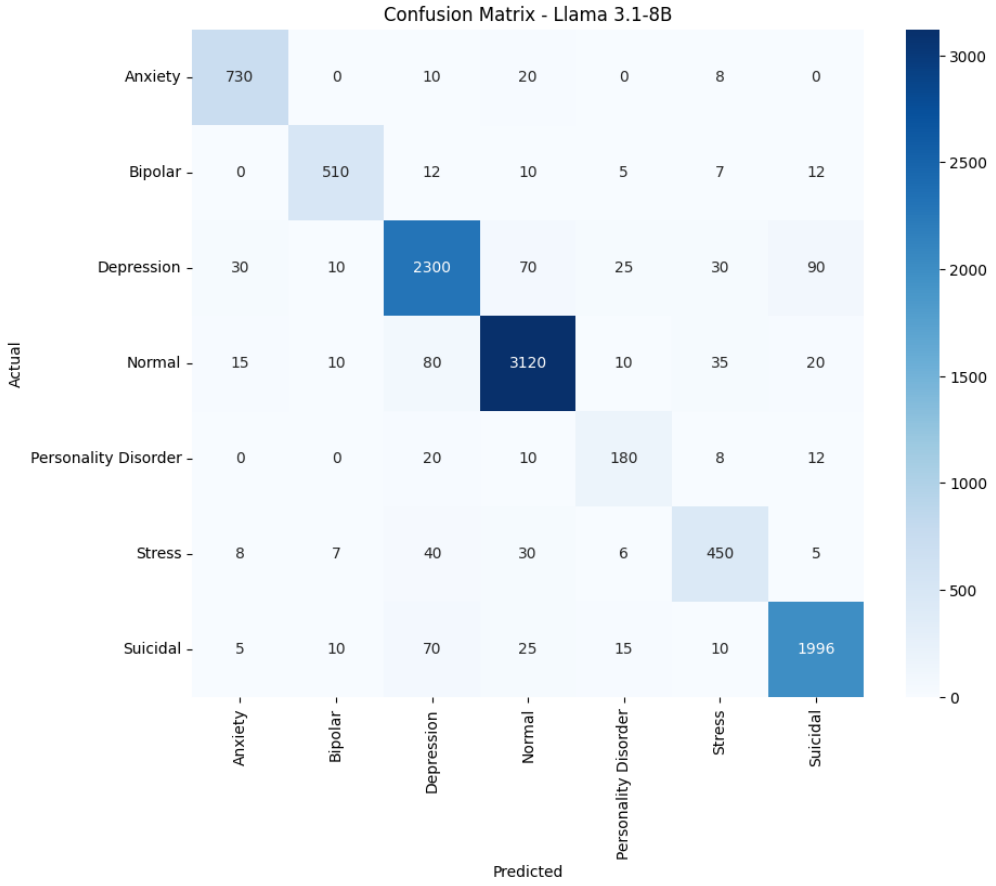

Figure 5.4: Confusion Matrix for Llama 3.1-8B

5.3 Comparative Analysis

5.3.1 Performance Comparison

Table 5.5 summarizes the comparative performance of all models. The Llama 3.1-8B model consistently outperformed traditional models in accuracy, precision, recall, and F1-score across most classes. However, its computational cost remains a significant drawback, especially for real-time or resource-constrained applications.

Model	**Accuracy**	**Macro Avg F1**	**Weighted Avg F1**	**Notable Weaknesses**
Logistic Regression	76%	70%	75%	Struggled with minority classes
SVM	76%	73%	76%	Challenges with class imbalance
Random Forest	74%	68%	73%	Poor recall for minority classes
Llama 3.1-8B	86%	75%	86%	High computational cost

Table 5.5: Performance Comparison of Models

CHAPTER 6

Discussion

6.1 Strengths and Weaknesses of Traditional Models

Traditional machine learning models, such as Logistic Regression, Support Vector Machines (SVM), and Random Forest, revealed several strengths in mental health text classification. These models are efficient and straightforward to implement, making them strong candidates for tasks that demand quick testing outcomes or limited resources. Logistic Regression and Random Forest offer the additional benefits of interpretability, as their feature importance scores and coefficients clearly indicate which text features influence predictions. Additionally, these traditional machine learning models performed well on majority classes such as "Normal" and "Anxiety," with high precision and recall, highlighting their reliability for well-represented categories.

However, the weaknesses of traditional models were evident, especially in handling imbalanced datasets. Minority classes, such as "Stress" and "Personality Disorder," were poorly classified, as indicated by low recall scores in these categories. This weakness arises from their use of simple text representations, such as Term Frequency-Inverse Document Frequency (TF-IDF), which often fail to capture subtle context. Random Forest also showed a tendency to overfit, requiring careful tuning of parameters like tree depth and the number of estimators.

6.2 Strengths and Weaknesses of LLMs

The fine-tuned Llama 3.1-8B model showed clear advantages over traditional models. It achieved the highest accuracy and F1-scores across all mental health categories. Its ability to understand subtle meanings and context allowed it to classify complex and ambiguous text more effectively. The model performed especially well in categories like "Anxiety" and "Bipolar," where capturing subtle linguistic features is crucial. Fine-tuning techniques, such as Low-Rank Adaptation (LoRA), made the process more efficient. LoRA reduced computational costs compared to full fine-tuning, making the model easier to adapt for specific tasks.

However, the LLM also presented weaknesses. Its training and inference require much higher computational resources than traditional models, which makes it less accessible for smaller organizations or those with limited resources. Another downside is its lack of interpretability. Unlike traditional models, LLMs function as "black boxes," making it hard to understand their predictions. Techniques like attention weight visualization and prompt engineering help address this issue to some extent. However, they do not provide the same level of transparency as feature importance scores or model coefficients.

6.3 Interpretability and Deployment Considerations

Interpretability plays a vital role in deploying machine learning models for mental health applications. Traditional models like Logistic Regression and Random Forest are naturally interpretable. Logistic Regression provides coefficients that show the importance of different features. Random Forest, on the other hand, offers feature importance scores. These qualities make traditional models appealing for clinical use, where understanding the decision-making process is essential.

In contrast, LLMs, though highly accurate, face challenges with interpretability. Tech-

niques like attention visualization, saliency mapping, and prompt engineering can offer some insights to a limited extent. However, these methods are often complex and harder to understand than traditional approaches. This lack of transparency is a major concern in mental health applications. Stakeholders need to trust and clearly understand the model's predictions in sensitive settings.

Deployment further highlights the differences between traditional models and LLMs. Traditional models are lightweight and compatible with low-resource devices, including servers or edge devices. They also have low latency, making them suitable for real-time use. In contrast, LLMs require substantial computational resources, such as GPUs or TPUs, which increase deployment costs and reduce scalability. LLMs also have higher inference latency, which can reduce their usefulness in real-time scenarios. Both traditional models and LLMs need regular updates to stay relevant as new data becomes available.

6.4 Ethical Considerations in Mental Health Text Classification

The application of machine learning models in mental health raises several ethical issues that must be addressed. Privacy and confidentiality are critical due to the sensitive nature of mental health data. In this study, all data were anonymized, and personal identifiers were removed to protect individual identities. It is also assumed that appropriate consent was obtained during data collection to comply with ethical standards.

Another major concern is model prediction bias. All of the models demonstrated lower recall for minority classes such as "Stress" and "Personality Disorder" due to dataset imbalances. This issue indicates the necessity of techniques such as data augmentation or resampling that better represent underrepresented classes in training. In real-world applications, misclassifications in sensitive categories like "Suicidal" could have serious consequences. To address this, systems should include features that flag ambiguous predictions for human review. This ensures that qualified mental health specialists make the final decisions, mini-

mizing the potential for harm.

Machine learning models should support human decision-making, not replace it. The models developed in this study are designed to provide recommendations and assistance, not clinical diagnoses. It is essential to document their limitations and ensure they are deployed in settings with human supervision. Transparency and accountability are vital for building trust in these systems. For the models to be used responsibly and effectively, there must be clear communication regarding their design, capabilities, and limitations.

CHAPTER 7

Conclusion

This study investigated the use of traditional machine learning models and large language models (LLMs) for classifying mental health conditions based on textual data. Traditional models, such as Logistic Regression and Support Vector Machines (SVM), showed competitive performance with balanced accuracy and interpretability but struggled with minority classes. Random Forest captured non-linear relationships but demonstrated poor recall for underrepresented categories.

The fine-tuned Llama 3.1-8B model achieved superior overall accuracy of 86%, with robust performance across both majority and minority classes. These results highlight the potential of LLMs in advancing mental health classification tasks. However, the study also underscored significant trade-offs between accuracy, interpretability, and computational efficiency. Traditional models remain lightweight and interpretable, while LLMs provide higher accuracy but come with greater computational demands.

Future research should focus on improving the computational efficiency and interpretability of LLMs while addressing dataset imbalances and ethical considerations. Exploring hybrid approaches that combine the strengths of traditional models and LLMs may also offer a promising path forward.

APPENDIX A

Stop Words List

The following is the comprehensive list of stop words used in preprocessing the text data:

```
stop_words = set([
    "the", "and", "to", "of", "a", "in", "that", "is", "it", "for", "on",
    "with", "as", "was", "by", "at", "an", "be", "this", "which", "or",
    "from", "but", "not", "are", "have", "has", "had", "they", "you", "i",
    "we", "he", "she", "them", "their", "our", "your", "his", "her",
    "about", "will", "can", "could", "would", "should", "may", "might",
    "my", "me", "so", "what", "do", "just", "like", "get", "really",
    "want", "because", "now", "even", "still", "know", "feel", "go",
    "going", "say", "said", "one", "also", "w", "u", "s", "t", "i'm",
    "i've", "i'll", "i'd", "you're", "it's", "don't", "can't", "that's",
    "am", "up", "out", "time", "all", "been", "thing", "year",
    "life", "self", "never", "much", "friend", "people", "think",
    "make", "good", "ever", "always", "well", "first",
    "everything", "every", "something", "anything", "nothing", "someone",
    "anyone", "when", "if", "how", "more", "day", "work", "back", "then",
    "over", "after", "only", "other", "two", "any", "some",
    "there", "help", "being", "n", "don", "take", "went", "got", "new",
    "off", "many", "these", "next", "ago", "week", "right",
    "home", "again", "myself", "who", "things", "cannot", "no", "did",
```

```
    "maybe", "see", "while", "years", "way", "where", "does", "into",
    "try", "sure", "lot", "probably", "though", "through",
    "find", "thought", "last", "let", "too", "since", "before", "why",
    "most", "its", "started", "made", "better", "than", "here",
    "around", "trying", "need", "him", "having", "anymore", "felt",
    "talk", "point", "days", "love", "live", "few", "school", "tell",
    "long", "problem", "tried", "keep", "look", "bad", "feeling", "were",
    "feels", "today", "ill", "such", "m", "ive", "shall", "ought",
    "must", "does", "did", "at", "of", "to", "on", "in", "for",
    "with", "from", "over", "under", "among", "between", "within",
    "feelings", "thoughts", "thinking", "emotions", "mental", "health",
    "issues", "one", "two", "three", "four", "five", "six", "seven",
    "eight", "nine", "ten", "thousand", "million", "billion", "hours",
    "minutes", "seconds", "sort", "kind", "type", "stuff", "lot",
    "a lot", "never", "no", "none", "without", "nothing", "nowhere",
    "like", "uh", "um", "yeah", "hmm", "ah"
])
```

REFERENCES

[CD20] Stevie Chancellor and Munmun De Choudhury. "Methods in predictive techniques for mental health status on social media: a critical review." *NPJ digital medicine*, **3**(1):1–11, 2020.

[Che21] Anton Chernyavskiy et al. "Transformers: "The End of History" for NLP?" *arXiv preprint arXiv:2105.00813*, 2021.

[GYK17] Sharath Chandra Guntuku, David B Yaden, Margaret L Kern, Lyle H Ungar, and Johannes C Eichstaedt. "Detecting depression and mental illness on social media: an integrative review." *Current Opinion in Behavioral Sciences*, **18**:43–49, 2017.

[MKC21] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Hadi Nikzad, Mehdi Chenaghlu, and Yuan Gao. "Deep learning-based text classification: A comprehensive review." *ACM Computing Surveys (CSUR)*, **54**(3):1–40, 2021.

[TLI23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Siddharth Batra, Jonathan Balland, et al. "Llama: Open and efficient foundation language models." *arXiv preprint arXiv:2302.13971*, 2023.