Fine-tuning BERT for Sentiment Analysis on Mental Health Texts

Amrita Moturi
amoturi@ucsd.edu

1 Abstract

In this paper, we explore the application of a fine-tuned BERT-based classifier to predict mental health conditions from user statements. The classifier was trained on an aggregated Kaggle mental health dataset, achieving an overall accuracy of 83% and a weighted F1-score of 0.83, outperforming traditional baseline models like Linear SVM (accuracy: 77.89%, F1-score: 0.78). BERT demonstrated superior handling of minority classes, with significant improvements in F1-scores for "Bipolar" (0.89 vs. 0.61) and "Personality Disorder" (0.77 vs. 0.68) compared to SVM. However, challenges arose in accurately classifying overlapping categories such as "Stress" and "Depression" and in identifying "Suicidal" text, where BERT underperformed relative to simpler models. These findings emphasize the importance of addressing class imbalances and refining loss functions for complex datasets. This paper highlights the potential of pre-trained language models for use in mental health detection and support, offering insights into their strengths and limitations.

2 Introduction

With the rise of large language models (LLMs) and foundation models, machine learning has made it more accessible and accurate to analyze highly sensitive and complex subjects like mental health. Social media platforms have made it easier for people to share their innermost thoughts, emotions, and feelings, often finding comfort in the anonymity of a username. As a result, there is more publicly available data. This large corpus of data presents an opportunity to use machine learning to identify and understand mental health conditions from online communication, offering the potential to reach a wider group of people and provide initial support to those without access to a specialist.

In this project, we aimed to:

- Collect and pre-process dataset
- Perform exploratory data analysis to understand relevant features and visualize data
- Build, train, and perform hyper parameter tuning on baseline models such as Logistic Regression, SVM, and a Random Forest Classifier with TF-IDF vectorization on the dataset and examine its performance
- Fine-tune pretrained BERT model on the dataset and tune to achieve better performance than baselines
- Address issues related to unbalanced data to improve performance on under performing class labels

3 Related work

Similar work has been done in the paper 'Their Post Tells the Truth: Detecting Social Media Users' Mental Health Issues with Sentiment Analysis' (Herdiansyah et al., 2023), which performs sentiment analysis on 10,000 tweets from Twitter, categorizing them into Positive (having a mental health condition), Negative (no mental health disorder), or Neutral (the presence or absence of a mental health disorder cannot be identified), based on the presence of certain keywords associated with the Negative and Positive categories.

Another paper used Support Vector Machines (SVMs) to classify 7,321 blog posts from 271 college students into depressive or non-depressive states (Zhang et al., 2022). On a corpus of 2.5M tweets, researchers employed linear SVMs and a Naïve Bayes Classifier to compare the accuracy

of the models, ultimately achieving an 81% accuracy rate in classifying depression using the SVM model (Nadeem, 2016).

Reddit is another common platform studied; in one study, researchers aggregated 10,000 posts to investigate suicidal vs. non-suicidal posts. They employed Support Vector Machine, Logistic Regression, and Multinomial Naïve Bayes classifiers to separate the posts into the two categories. Their results showed that the Logistic Regression model outperformed the others in terms of accuracy and precision, while Multinomial Naïve Bayes yielded the best recall (Kaushik et al., 2023).

An analysis of patient self-narratives to determine post-traumatic stress disorder (PTSD) used decision trees, Naïve Bayes, Support Vector Machine, and the product score model in combination with n-gram representation models to identify patterns between verbal features in self-narratives and psychiatric diagnoses (He et al., 2017).

4 Sentiment Analysis for Mental Health Kaggle dataset

This data contains over 53K posts and 51K unique text statements sourced from social media posts, Reddit posts, Twitter posts, and other platforms, all consolidated in the Sentiment Analysis for Mental Health Kaggle dataset. It contains the features unique_id, statement (the textual data or post), and status (the tagged mental health status of the statement). The statuses fall under the categories Normal, Depression, Suicidal, Anxiety, Stress, Bipolar, and Personality Disorder. The labels are split 31%, 29%, 20%, 7%, 5%, 5%, and 2% respectively, as seen in Figure 1.

Some examples from the dataset include *"You don't have to complicate things anymore, people,"* given a label of "Normal" and another example is *"Have you ever been so nervous, scared, and anxious that you want to throw up?"* given the label of "Anxiety." This data was split into a train, validation, and test split, and given a text post or statement, we aim to create an accurate model that can classify it into one of the seven categories, as seen in Table 1. The challenge of this task using this dataset comes with the training data imbalance, which poses the risk of the model over predicting the labels with the largest group. This issue was addressed during model training.

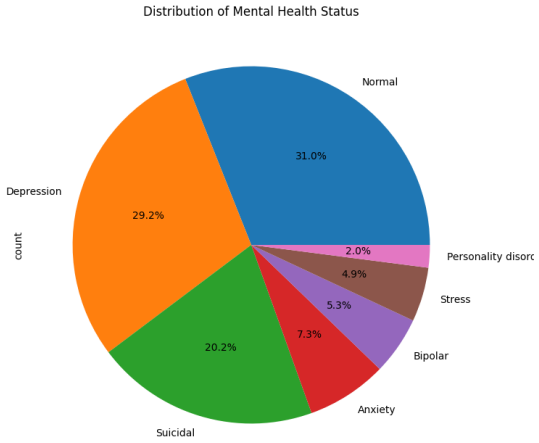

Figure 1: Distribution of mental health status

Statement	Predicted Label
I do not think I can do this anymore. I just literally do not care at all. I have heard so much advice. I have received so much help. I just do not care. I am tired. Waking up in the morning is pure dread.	Suicidal
Hey all I've been getting really bad brain zaps recently. Like constant zap zap zap. I take Wellbutrin and my dose just went up, so that could be why. Does anyone else experience this? Should I be concerned?	Anxiety

Table 1: Example Statements and Predicted Labels

4.1 Data preprocessing

The dataset consisted of 362 instances of NaN values across the statements, which we dropped before performing any data analysis or modeling. We added additional features, such as the length of the statement (by characters) and the number of words in each statement, to see if there was a correlation between these features and the labels. A statement labeled as "Normal" had the shortest length and fewest number of words, with an average of 90 characters and 17 words, while the personality disorder category had the longest statements and the greatest number of words, with an average of 957 characters and 179 words, as seen

in Figures 2 and 3.

Since the statement lengths and number of words were also heavily right-skewed, we performed a log transformation of the data to normalize it.

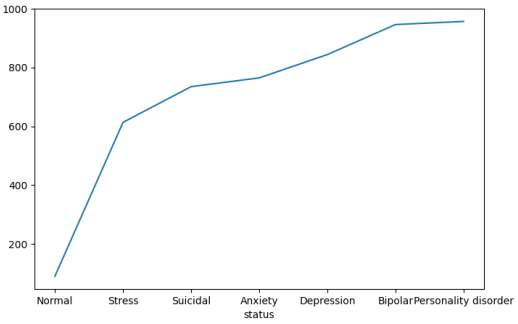

Figure 2: Distribution of statement lengths by mental health status

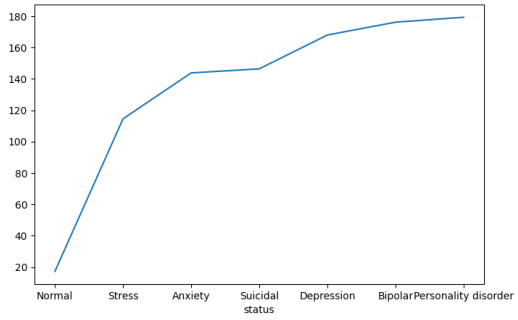

Figure 3: Distribution of number of words by mental health status

5 Baselines

We implemented several baselines, including a Logistic Regression model, Support Vector Machine (SVM), and Random Forest model. The associated code can be referenced in the Google Colab under the section "Baseline Models: Logistic Regression, SVM, and Random Forest Classifier using TF-IDF metrics." We used a 60/20/20 train-test-validation split. Each of the data samples was broken down using TF-IDF vectorization to extract the most relevant features. The Logistic Regression model was implemented using the sklearn module and tuned based on the maximum number of features (we tried 1000, 2000, and 5000) for the TF-IDF vectorization and the coefficient C for the regularization strength (we tried 0.01, 0.1, 1, and 10). These results can be seen in Table 2. The best Logistic Regression model, using 5000 features for the vectorizer and a C value of 1, achieved an overall accuracy of 77.3% on the test set. This model classified "Suicidal" samples the best, with a precision of 0.86, recall of 0.95, and F1-score of 0.91. The model performed the worst on "Bipolar" samples, with a precision of 0.71, recall of 0.50, and F1-score of 0.58.

Similarly, a linear SVM model was implemented from the sklearn.svm module and optimized using the same hyperparameter search as for the Logistic Regression model, as seen in Table 3. The linear SVM model classified the "Suicidal" samples most effectively, with a precision of 0.88, recall of 0.95, and F1-score of 0.92. However, it classified "Bipolar" samples with a precision of 0.70, recall of 0.54, and F1-score of 0.61. The best SVM model performance used a C value of 1 and 5000 features for the vectorizer, achieving an accuracy of 77.89% on the test set.

We implemented a Random Forest classifier from the sklearn.ensemble module and tuned the hyperparameters for the number of features for the TF-IDF vectorizer, the number of trees, and the maximum depth to consider for each split, as seen in Table 4. The best model used 5000 features for the TF-IDF vectorizer, a value of 200 for the number of estimators, and a maximum depth of 30, achieving an accuracy of 65.95% on the test set. The Random Forest model classified the "Suicidal" samples most effectively, with a precision of 0.76, recall of 0.95, and F1-score of 0.84. However, it classified "Bipolar" samples with a precision of 1.00, recall of 0.15, and F1-score of 0.26. The hyperparameters for each model were tuned by searching through every combination to see which produced the highest accuracy on the validation set.

max_features	**C**	**Accuracy**
1000	0.01	0.5749
1000	0.1	0.7073
1000	1	0.7531
1000	10	0.7534
2000	0.01	0.5701
2000	0.1	0.7090
2000	1	0.7659
2000	10	0.7635
5000	0.01	0.5632
5000	0.1	0.7046
5000	1	0.7691
5000	10	0.7686

Table 2: Logistic Regression Model Results

max_features	C	Accuracy
1000	0.01	0.6952
1000	0.1	0.7498
1000	1	0.7561
1000	10	0.7538
2000	0.01	0.6999
2000	0.1	0.7648
2000	1	0.7682
2000	10	0.7546
5000	0.01	0.6914
5000	0.1	0.7683
5000	1	0.7700
5000	10	0.7423

Table 3: Linear SVM Model Results

n_estimators	max_depth	Accuracy
50	10	0.5506
50	20	0.6313
50	30	0.6594
100	10	0.5499
100	20	0.6282
100	30	0.6589
200	10	0.5491
200	20	0.6274
200	30	0.6612

Table 4: Random Forest Classifier Results

6 Methods and Results

We sought to develop a model that outperforms the baselines in terms of accuracy across the models. With the introduction of foundation models, we now have access to open-source large language models like Bidirectional Representation for Transformers (BERT). The power of BERT lies in its ability to exploit bidirectional context, enabling it to acquire complex and insightful word and phrase representations. By considering both the left and right context of each word, BERT captures the full meaning of a word in its context, unlike earlier models that were limited to unidirectional context. This makes BERT especially useful for handling ambiguous and complex text. Additionally, the pre-trained BERT model can be fine-tuned with just one extra output layer, making it well-suited for classification tasks (Devlin et al., 2019).

We implement a pretrained BERT language model that was accessed on HuggingFace under the name 'bert-base-uncased' and utilized its AutoTokenizer. For our sentiment classification task, we used the AutoModelForSequenceClassification() model based on Google's 'bert-base-uncased'. We fine-tuned this model on our mental health training dataset for sentiment analysis. The associated code can be referenced in the Google Colab notebook under the section "My Approach: Utilizing pretrained language models (BERT) for sentiment analysis." Initially, we trained the model using the AdamW optimizer with a standard Cross Entropy loss function and a learning rate of 5e-5. After 3 epochs of training, the model achieved a training loss of 0.1935 with an accuracy of 92.35%, and a validation loss of 0.4594 with an accuracy of 83.07%. The "Normal" class achieved the best metrics, with a precision, recall, and F1-score of 0.96 across all categories, while the "Suicidal" class performed the worst, with precision, recall, and F1-scores of 0.70, 0.72, and 0.71, respectively.

Since validation accuracy was steadily increasing after each epoch, and the categories "Suicidal," "Stress," and "Bipolar" underperformed compared to the others, we explored alternative approaches. To address the unbalanced dataset and improve performance, we modified the training loop to use a weighted Cross Entropy loss function based on class frequencies and extended the training to 5 epochs. This approach achieved an accuracy of 81.93%, with the "Normal" category again performing the best (precision, recall, and F1-score of 0.96, 0.95, and 0.95) and the "Suicidal" category performing the worst (precision, recall, and F1-scores of 0.70, 0.68, and 0.69). While the "Suicidal" category performed better in the baseline models, it consistently performed the worst with BERT. Notably, training for more epochs improved the performance of the "Stress" and "Bipolar" categories, achieving F1-scores of 0.74 and 0.88, respectively, highlighting the benefits of extended training.

When comparing BERT with the best-performing baseline, the Linear SVM, BERT demonstrated superior performance in overall text classification. It achieved an accuracy of 83% and a weighted F1-score of 0.83, outperforming the SVM's accuracy of 77.89% and weighted F1-score of 0.78. BERT also handled imbalanced classes better, with significant improvements in minority classes such as "Bipolar" (F1: 0.89 vs. 0.61) and "Personality Disorder" (F1: 0.77 vs. 0.68). Additionally, BERT excelled in capturing contextual nuances, achieving a higher macro F1-score (0.82 vs. 0.74) and better overall precision and recall. While the SVM performed well on the "Suicidal" class, achieving a higher recall

and F1-score, it struggled with class imbalance and generalization. These results showcase the advantages of pre-trained language models like BERT, which consistently deliver better performance across complex and imbalanced datasets compared to traditional methods like SVM with TF-IDF.

We implemented a working solution for classifying the mental health dataset, as detailed in the submitted Google Colab notebook. We utilized a MacBook Air with an M1 chip and accessed a T4 GPU via Google Colab. Model checkpoints were saved to allow retrieval of the trained model without requiring retraining. While we attempted to use GridSearchCV for parameter optimization of the baseline models, the runtime was prohibitively long and caused frequent disconnections in Google Colab. As an alternative, we switched to using for-loops for hyperparameter tuning. Training the fine-tuned BERT model for 3 epochs took approximately 52 minutes, while training for 5 epochs took around 90 minutes.

7 Error analysis

For the baseline models such as logistic regression, errors were most pronounced in the minority classes, such as "Stress" (F1: 0.59) and "Bipolar" (F1: 0.58). Linear SVM similarly struggled with these classes, achieving F1 scores of 0.68 and 0.61, respectively. These models often failed on rare categories, likely due to their reliance on linear boundaries, which are insufficient for capturing nuanced features in imbalanced datasets. Inputs with overlapping features, such as those between "Stress" and "Anxiety" or "Bipolar" and "Depression," were common points of failure. For instance, the sample "It was worse than that, I wanted to throw up, not because I found the scars disgusting but because the person I loved so so much, had done that to herself" was predicted as "Depression" by the model, while the true label was "Stress." These classes share semantic similarities (e.g., mentions of mood instability or emotional distress), making linear classification particularly challenging. Random Forest also performed poorly on minority classes, with F1 scores of 0.34 for "Stress" and 0.26 for "Bipolar," likely due to data sparsity in these categories and difficulty distinguishing semantics in overlapping symptoms.

The BERT-based approach performed the worst on the "Suicidal" and "Stress" classes, with F1 scores of 0.71 and 0.75, respectively. Instances with vague expressions or overlapping symptoms posed significant challenges for this model. For example, in the statement "I have suffered from depression for over 10 years now...I do not want to die but I do not know what else to do," the model predicted the label as "Suicidal," while the true label was "Depression." BERT's attention mechanism may have focused on dominant terms without sufficiently disambiguating the speaker's intent. The presence of terms associated with both the true and predicted labels likely contributed to this misclassification.

An analysis of the confusion matrices in Figures 4, 5, and 6 reveals that the baseline models performed similarly across classes, with the best performance on the "Suicidal" class and the worst on "Stress" and "Bipolar." In contrast, the BERT model performed the best on the "Normal" class and the poorest on "Suicidal," as shown in Figure 7. This discrepancy is likely due to the model leveraging the large proportion of "Normal" samples in the training set.

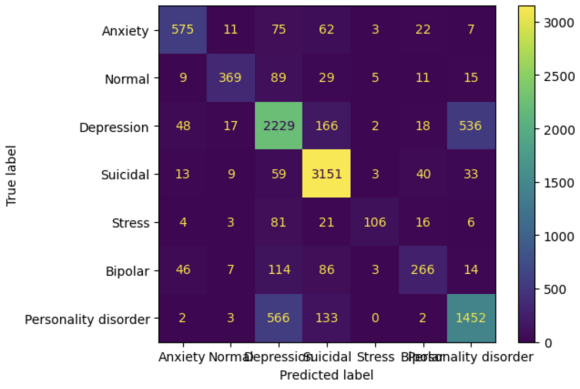

Figure 4: Confusion Matrix for Logistic Regression model

8 Conclusion

This paper highlights the potential of leveraging pre-trained language models like BERT, to classify mental health-related text effectively. The results emphasize the importance of context-aware models, as BERT consistently outperformed traditional baselines like Linear SVM and logistic regression, especially for imbalanced datasets. One major takeaway is how well BERT captured nuanced semantic patterns in minority classes like "Bipolar" and "Personality Disorder," which the

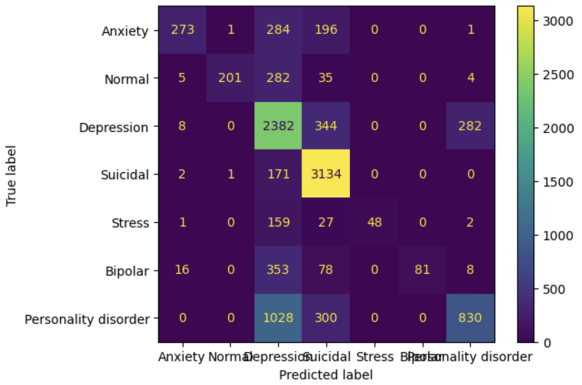

Figure 5: Confusion Matrix for Linear SVM model

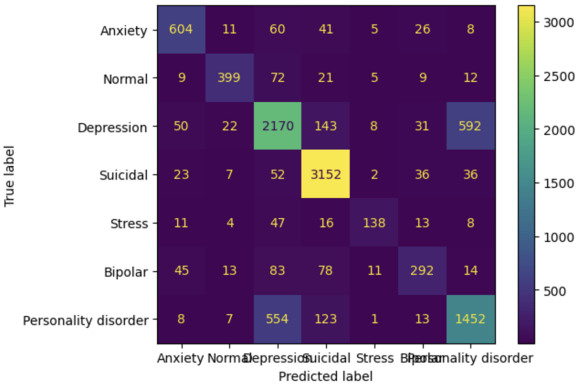

Figure 6: Confusion Matrix for Random Forest Classifier

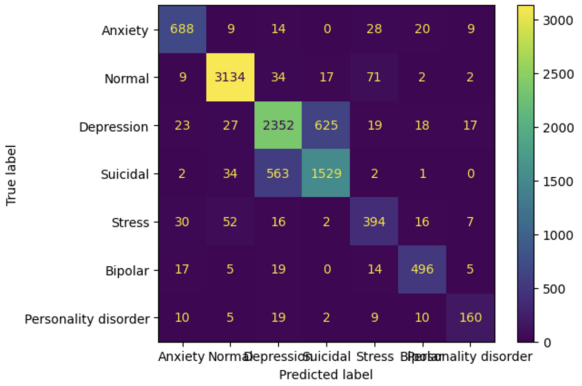

Figure 7: Confusion Matrix for BERT-based model

baselines struggled with. However, the project also underscored the inherent challenges of handling complex and overlapping categories, such as distinguishing between "Stress," "Anxiety," and "Depression," or accurately identifying "Suicidal" instances.

What proved surprisingly difficult was addressing the imbalanced class distribution and its impact on model performance. While weighted loss functions and increased epochs improved results for some minority classes, overfitting became a significant issue. Additionally, parameter tuning for baseline models was unexpectedly time-intensive, especially when leveraging methods like GridSearchCV, which led to runtime limitations. Another surprising finding was that BERT, despite its advanced architecture, performed worst on the "Suicidal" class, suggesting that even state-of-the-art models may struggle with vague or ambiguous text, and that simpler models might still hold value in specific contexts like this.

Future considerations for this paper include experimenting with other transformer-based models like RoBERTa or DistilBERT to understand how architectural differences affect performance on mental health datasets. We also seek to improve model performance by addressing class imbalance through advanced techniques like SMOTE or data augmentation.

References

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

He, Q., Veldkamp, B. P., Glas, C. A. W., and de Vries, T. (2017). Automated assessment of patients' self-narratives for posttraumatic stress disorder screening using natural language processing and text mining. In *Assessment*, volume 24, pages 157–172. SAGE Publications.

Herdiansyah, H., Roestam, R., Kuhon, R., and Santoso, A. S. (2023). Their post tells the truth: Detecting social media users' mental health issues with sentiment analysis. In *Procedia Computer Science*, volume 216, pages 691–697. Elsevier.

Kaushik, B., Sharma, A., Chadha, A., and Sharma, R. (2023). Machine learning model for sentiment analysis on mental health issues. In *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, pages 21–25.

Nadeem, M. (2016). Identifying depression on twitter. In *arXiv preprint*, volume arXiv:1607.07384.

Zhang, T., Schoene, A. M., Ji, S., et al. (2022). Natural language processing applied to mental illness detection: a narrative review. *npj Digital Medicine*, 5:46.