Analyzing Mental Health Posts on Social Media: A Deep Learning Approach to Reddit Post-Classification

Georgios Ioannou gi2100@nyu.edu
Center for Data Science
New York University
New York, NY 10011-1185, USA

Zechen Yang zy3398@nyu.edu
Grossman School of Medicine
New York University
New York, NY 10016-1185, USA

Abstract

Early identification of mental health conditions through social media analysis presents a promising avenue for scalable digital intervention. In this study, we develop and evaluate deep learning models to classify Reddit posts into six mental health categories: depression, anxiety, bipolar disorder, borderline personality disorder, schizophrenia, and autism. We compiled a large dataset from condition-specific subreddits and applied rigorous preprocessing (Bird et al., 2009; Miller, 1995), including cleaning, augmentation, and tokenization. Four models were built from scratch (BiLSTM, CNN + BiLSTM, CNN, and BiGRU), and two transformer-based models (BERT and Mistral-7B) were fine-tuned for comparison. Among these, a class-balanced BERT model achieved the highest performance, with 87.3% accuracy and F1-score, outperforming even larger models like Mistral-7B. While r/mentalhealth posts were excluded from training, they were collected to simulate real-world scenarios and test potential model biases. We also discuss the limitations of class imbalance, lack of co-morbidity modeling, and the need for broader validation. Future directions include the use of large language models for synthetic data generation, zero-shot classification, and enhancing fairness across demographics. Our findings affirm the feasibility of using deep learning on social media content to support early mental health detection and intervention.

Keywords: BiLSTM, CNN, BiGRU, BERT, Mistral 7B, NLP, LORA, PEFT

1 Introduction

Mental health discussions are increasingly prevalent on social media, where users openly share psychological experiences and seek support, often leveraging the anonymity of these platforms to disclose issues they might withhold offline. These digital communities not only foster emotional expression but also facilitate peer connection and informal self-diagnosis.

Recent advances in deep learning (Kim et al., 2020; Goldberg, 2016; Wolf et al., 2020), particularly in text classification, have enabled the identification of linguistic patterns linked to various mental health conditions (Kim, 2020). As millions turn to platforms like Reddit for support and self-expression, such data presents opportunities for computational mental health research.

Building on prior work, this study develops a deep learning model to classify Reddit posts into specific mental health categories—depression, anxiety, bipolar disorder, borderline personality disorder (BPD), schizophrenia, and autism—based solely on textual content. Reddit was chosen for its many mental health-focused subreddits, where users discuss specific conditions in detail.

We hypothesize that transformer-based models can accurately classify user posts into relevant subreddits, enabling targeted support. This paper outlines our data collection and preprocessing methods, model architecture, evaluation results, and ethical considerations. Our findings demonstrate the potential of machine learning to identify mental health concerns through social media content, contributing to the field of computational mental health.

Research question: Can we determine whether a user's social media post indicates a mental illness?

2 Data

To train and evaluate our classification models and fine-tune transformers (Howard & Ruder, 2018), we compiled 488,472 posts from seven mental health-related subreddits: r/anxiety (86,243), r/autism (7,142),r/bipolar (41,493), r/bpd (38,216), r/depression (58,496), r/schizophrenia (17,506), and r/mentalhealth (39,373). The first six represent specific diagnoses, while r/mentalhealth, a general mental health forum, was used only for exploratory analysis and excluded from model training to preserve diagnostic clarity. Each post includes a title, body text, and subreddit label. Due to class imbalance, we applied class weighting during transformer fine-tuning but did not use resampling techniques

In terms of textual characteristics, the combined length of post titles and body text varied widely, with an average length of 1,016 characters (standard deviation: 1,197), a median length of 674 characters, and a maximum length of 41,573 characters. These statistics highlight the high variability in user expression across posts, which poses both challenges and opportunities for deep learning models tasked with capturing relevant features for classification. Basic preprocessing steps were applied to clean and normalize the text before inputting it into the models.

2.1 Version 1 of Data

Before training, we performed preprocessing to ensure data consistency and suitability for text classification. Posts from r/mentalhealth were removed, as they lacked specific diagnoses. Column and subreddit names were standardized to lowercase, and the title and text fields were merged into a single input field. We dropped rows with null subreddit labels and removed duplicates based on the combined title_text field to ensure each post had a unique label.

We then applied a custom text cleaning pipeline using NLTK, implemented through a Cleaner class. The pipeline converted text to lowercase, removed stopwords, numbers, punctuation, URLs, and applied lemmatization and stemming (via Porter Stemmer). These steps normalized the text and reduced noise, resulting in a cleaner dataset optimized for deep learning.

2.2 Version 2 of Data

To create a more diverse dataset for BERT training, we transformed version 1 into version 2 using text augmentation (Miller, 1995; Bird et al., 2009; Kudo and Richardson, 2018). Two techniques were applied. Synonym replacement was used for words over 3 characters, there was a 10% chance of replacement with WordNet-derived synonyms—only if valid, underscore-free alternatives existed—to introduce semantic variation while preserving meaning. Random typos with a 10% chance, words over 3 characters were altered using realistic keyboard-based typos, affecting common letters (e.g., 'a', 'e', 'i') via an adjacency dictionary.

These two augmentations increased training diversity, improved model robustness to typos and noise, and helped prevent overfitting by enhancing generalization to real-world text.

3 Materials & Methods

We developed and evaluated six models: four built from scratch (BiLSTM, CNN + BiLSTM, CNN, BiGRU) and two fine-tuned transformers (BERT and Mistral 7B).

For all models developed from scratch, the dataset was split 80/10/10 into training (358,314), validation (44,789), and test (44,790) sets. Subreddit labels were encoded using scikit-learn's LabelEncoder, and text was tokenized with Keras's Tokenizer, converting combined title and body text into integer sequences. These were pre-padded to a fixed length based on the 95th percentile of post lengths. Models were built in TensorFlow2 and trained on an H100-SXM5-80GB GPU using a batch size of 64, the Adam optimizer (learning rate 0.001), and categorical cross-entropy loss. Training used EarlyStopping to restore the best weights if validation accuracy stalled for 3 epochs, and ReduceLROnPlateau to halve the learning rate after 2 stagnant epochs, down to a minimum of 0.00001. (Chollet et al., 2015; Abadi et al., 2016; Kingma and Ba, 2015)

The fine-tuned transformer models were implemented in PyTorch (Paszke et al., 2019) and trained on a V100 GPU.

3.1 BiLSTM

Our Bidirectional Long Short-Term Memory (BiLSTM) architecture begins with an embedding layer that transforms input tokens into 200-dimensional dense vectors, followed by a dropout layer with a rate of 0.6 to mitigate overfitting. The network then stacks three BiLSTM layers with increasing output sizes of 80, 160, and 320 units, respectively. The final dense layer applies a softmax activation function to output class probabilities over the target mental health categories. This architecture was designed to progressively capture higher-level sequential patterns in Reddit posts and is trained end-to-end for multi-class classification. (Goldberg, 2016)

Each epoch took about 29 minutes on an H100-SXM5-80GB GPU.

3.2 CNN + BiLSTM

To leverage both local feature extraction and sequential modeling, we implemented a hybrid architecture combining Convolutional Neural Networks (CNN) with a BiLSTM layer. The

model begins with an embedding layer that maps input tokens to 200-dimensional dense vectors, followed by a dropout layer with a rate of 0.25 to reduce overfitting. A 1D convolutional layer with 128 filters and a kernel size of 5 is then applied to extract local n-gram features from the embedded sequences. This is followed by a max-pooling layer with a pool size of 4 to reduce dimensionality and highlight the most salient features. The output is then passed to a BiLSTM layer with 128 units, enabling the model to capture both forward and backward dependencies in the sequence. Finally, a dense layer with softmax activation outputs the probability distribution over the six target mental health classes. This architecture is designed to effectively integrate spatial and temporal representations for improved classification performance.

Each epoch took about 7.5 minutes on an H100-SXM5-80GB GPU.

3.3 CNN

We also implemented a CNN architecture designed to capture multi-scale textual features through parallel convolutional branches. The model begins with two parallel input pipelines, each consisting of an embedding layer that maps input tokens to 200-dimensional dense vectors. The first branch applies a 1D convolution with a kernel size of 3 to extract trigram-level features, while the second branch uses a kernel size of 5 to capture broader 5-gram patterns. Both branches include batch normalization, ReLU activation, dropout with a rate of 0.5, and global max pooling to produce fixed-length feature vectors. The outputs of the two branches are concatenated and passed through a dense layer with 128 ReLU-activated units, followed by another dropout layer (0.5) for regularization. The final output layer uses softmax activation to generate probability distributions over the six mental health classes. This dual-branch CNN architecture enables the model to simultaneously learn local patterns at different lengths of text.

Each epoch took about 7.25 minutes on an H100-SXM5-80GB GPU.

3.4 BiGRU

Our Bidirectional GRU (BiGRU) model begins with an embedding layer that projects tokens into a 200-dimensional vector space, followed by a dropout layer with a rate of 0.4 to reduce overfitting. A BiGRU layer with 128 units is then applied to extract contextual representations from both directions of the sequence. The resulting output is passed through a dense hidden layer with 128 units and ReLU activation, followed by another dropout layer (rate 0.5) to further enhance generalization. Finally, a softmax output layer predicts the probability distribution across the six mental health categories.

Each epoch took about 13 minutes on an H100-SXM5-80GB GPU.

3.5 BERT

The BERT (Devlin et al., 2019; Vaswani et al., 2017; Wolf et al., 2020) model strategy employs four training approaches on Reddit mental health posts: (1) training on all seven subreddits using cleaned text to create a comprehensive classifier, (2) training on six specific subreddits (excluding "mentalhealth") to focus on distinct conditions, (3) implementing class balancing with WeightedRandomSampler to address dataset imbalance and improve

performance on minority classes, and (4) using text transformations (synonym replacement and simulated typos) to enhance model robustness against real-world text variations. All models use the bert-base-uncased architecture with consistent hyperparameters (max length 256, batch size 16, learning rate 2e-5, 4 epochs), while differing in their dataset composition and preprocessing strategies.

3.6 Mistral 7b

The Mistral-7B fine-tuning strategy leverages a much larger foundation model (7B parameters) with advanced quantization techniques to enable training on limited hardware. It implements 4-bit quantization via BitsAndBytes, Parameter-Efficient Fine-Tuning (PEFT) with LoRA (targeting specific attention layers with r=8, alpha=32) (Hu et al., 2022; Dettmers et al., 2022), and aggressive memory optimizations (gradient checkpointing, micro-batch size of 2 with gradient accumulation of 64 steps). The approach includes advanced training configurations like learning rate of 2e-4, warmup ratio of 0.03, and max gradient norm of 0.3, while maintaining similar dataset processing to the BERT models but requiring substantially more computational resources.

4 Results

Four evaluation metrics were employed to validate the performance of the models: accuracy, precision, recall, and F1-score.

The results show that BERT consistently outperforms traditional deep learning models such as BiLSTM, CNN, and BiGRU across different versions of the dataset. However, its performance initially drops in Version 2 of the data, suggesting sensitivity to class imbalance. Once classes are balanced, BERT significantly outperforms even larger models like Mistral7B, emphasizing that model quality and data balance matter more than size alone in this context.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
BiLSTM	85.3	85.3	85.3	85.3
CNN + BiLSTM	84.7	84.7	84.7	84.7
CNN	84.1	84.1	84.1	84.1
BiGRU	85.2	85.2	85.2	85.2
BERT	**86.4**	**86.4**	**86.4**	**86.1**

Table 1: Performance Metrics on Version 1 of Data

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
BiLSTM	84.5	84.5	84.5	84.5
CNN + BiLSTM	83.8	83.8	83.8	83.8
CNN	83.3	83.3	83.3	83.3
BiGRU	**84.6**	**84.6**	**84.6**	**84.6**
BERT	83	81	79	80

Table 2: Performance Metrics on Version 2 of Data

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
BERT	**87.3**	**87.5**	**87.1**	**87.3**
Mistral7b	69.9	76.6	69.5	70.8

Table 3: Performance Metrics on Version 2 of Data With Balanced Classes

5 Discussion

By analyzing posts from six mental-health-related subreddits, our models were able to classify posts associated with specific disorders such as depression, anxiety, and schizophrenia. Although we focused exclusively on these six classes for model training and evaluation, we also collected data from r/mentalhealth as a negative class to test how the model might behave in real-world applications and assess possible biases. While this paper assumes a closed-world scenario with only six diagnostic categories, future work will aim to expand this scope. Specifically, we plan to explore zero-shot classification using large language models such as facebook/bart-large-mnli to evaluate generalizability, and we are interested in using LLMs to generate synthetic data to balance the dataset more effectively—an issue SMOTE (Chawla et al., 2002) failed to address due to incompatibilities with embedding-based inputs. Our findings reinforce the value of using social media as a complementary tool for early detection and intervention in mental health.

6 Conclusion

The Balanced Six BERT model achieved the highest performance (87.3% accuracy and F1), outperforming all other models, including the larger Mistral-7B. This shows that addressing class imbalance was more effective than using text transformations or larger models. Despite its size and advanced techniques, Mistral-7B lagged behind (69.9% accuracy), highlighting that data balance matters more than model size in mental health text classification.

7 Contributions

For this paper, Georgios was responsible for exploring, pre-processing, and cleaning the data, developing BiLSTM, CNN + BiLSTM, CNN, and BiGRU. Zechen was responsible for collecting the data, transforming the data, and fine-tunig BERT and Mistral-7B.

8 Code GitHub Link

You can find the code and diagrams of this paper in the following GitHub repository:
https://github.com/GeorgiosIoannouCoder/mindscanner

References

J. Kim, J. Lee, E. Park, et al. A deep learning model for detecting mental illness from user content on social media. Scientific Reports, 10:11846, 2020. doi:10.1038/s41598-020-68764-7.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proc. of NAACL-HLT, pages 4171–4186, 2019.

S. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. 8-bit Optimizers via Block-wise Quantization. In Proc. of ICLR, 2022.

E. Hu, Y. Shen, P. Wallis, et al. LoRA: Low-Rank Adaptation of Large Language Models. In Proc. of ICLR, 2022.

G. A. Miller. WordNet: A Lexical Database for English. Communications of the ACM, 38(11):39–41, 1995.

N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16:321–357, 2002.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In Proc. of ICLR, 2015.

F. Chollet et al. Keras. https://keras.io, 2015.

M. Abadi, P. Barham, J. Chen, et al. TensorFlow: A System for Large-Scale Machine Learning. In Proc. of OSDI, pages 265–283, 2016.

A. Paszke, S. Gross, F. Massa, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems, 32, 2019.

J. Howard and S. Ruder. Universal Language Model Fine-tuning for Text Classification. In Proc. of ACL, 2018.

T. Wolf, L. Debut, V. Sanh, et al. Transformers: State-of-the-Art Natural Language Processing. In Proc. of EMNLP: System Demonstrations, pages 38–45, 2020.

T. Kudo and J. Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Proc. of EMNLP: System Demonstrations, pages 66–71, 2018.

S. Bird, E. Klein, and E. Loper. Natural Language Processing with Python. O'Reilly Media Inc., 2009.

Y. Goldberg. A Primer on Neural Network Models for Natural Language Processing. Journal of Artificial Intelligence Research, 57:345–420, 2016.

A. Vaswani, N. Shazeer, N. Parmar, et al. Attention is All You Need. In Proc. of NeurIPS, 30, 2017.