Detecting Mental Health Signals in Text: A Structured Evaluation of NLP Approaches

Sergio Amortegui
Master in Applied Analytics
Universidad de los Andes
Email: your.email@domain.com

***Abstract*—Mental health is one of the most pressing global challenges of the 21st century, and early detection is essential to prevent escalation into severe or life-threatening situations. Because individuals frequently express psychological states through short written text—particularly on social media—natural language processing (NLP) offers a scalable avenue for identifying subtle linguistic markers associated with mental health conditions. This work presents a structured benchmark comparing classical machine-learning models, recurrent neural networks, and preliminary transformer-based methods for mental-health text classification. Following a rigorously controlled experimental setup with a fixed data split and consistent preprocessing pipeline, we evaluate seven mental-health categories using accuracy and macro-F1. Results show that TF–IDF linear models outperform recurrent neural networks on this dataset, while partial transformer runs suggest strong performance potential given adequate computational resources. The objective of this project is not to propose a novel architecture, but to provide a transparent, reproducible, and analytically grounded comparison of widely used NLP approaches for mental-health detection.**

***Index Terms*—Mental Health, NLP, Text Classification, Machine Learning, TF–IDF, LSTM, GRU, Transformers.**

I. Introduction

Mental health conditions such as anxiety, depression, bipolar disorder, stress, and suicidal ideation are increasingly prevalent worldwide. The World Health Organization reports that more than one in eight people live with a mental disorder [1]. Early detection is vital: many crises develop gradually, and the ability to identify warning signs before escalation can save lives.

Individuals frequently express emotional distress in written form—through social media posts, online forums, personal messages, and short textual statements. Such text often contains subtle linguistic cues related to internal states, cognitive distortions, or emotional dysregulation. NLP technologies therefore offer the possibility of scalable, low-cost, early screening mechanisms when human monitoring is impractical.

Prior research shows that both classical models (e.g., TF–IDF + SVM [2]) and neural architectures (LSTMs [3], transformers [4]) can detect mental-health signals from text. However, there is no consensus on which modeling family performs best under controlled, reproducible conditions. Inspired by existing literature, this project aims to provide a structured and fair comparison between several widely used NLP methods applied to the *same* mental-health dataset, following uniform preprocessing and a fixed data split to ensure transparency, reproducibility, and comparability.

Our contributions are:

- A rigorously controlled benchmark of classical, recurrent, and transformer-based approaches on the same dataset.
- A clear, fully reproducible preprocessing pipeline and experimental setup.
- A scientific discussion explaining why certain models outperform others in this context.

II. Dataset

We use the *Sentiment Analysis for Mental Health* dataset, consisting of short textual statements labeled into seven categories:

Anxiety, Bipolar, Depression, Normal, Personality Disorder, Stress, Suicidal.

Each entry contains:

- **statement**: a short text (user-written)
- **status**: the mental-health label

The dataset exhibits moderate class imbalance. Depression and Normal contain the most samples, whereas Personality Disorder and Bipolar are comparatively under-represented. The average statement length is short (typically 5–20 words), which influences the effectiveness of various modeling approaches.

To ensure reproducibility, we adopt a *three-way split*:

$$\text{Train} = 70\%, \quad \text{Test} = 15\%, \quad \text{Validation} = 15\%.$$

Train and test sets are used internally by each model during training. **The validation set is reserved exclusively for model comparison**, ensuring a fair and transparent evaluation.

III. Preprocessing

All models share an identical preprocessing pipeline to eliminate confounding factors.

A. Text Normalization

Each statement undergoes:

- lowercasing
- Unicode normalization (NFC)
- punctuation removal
- whitespace normalization

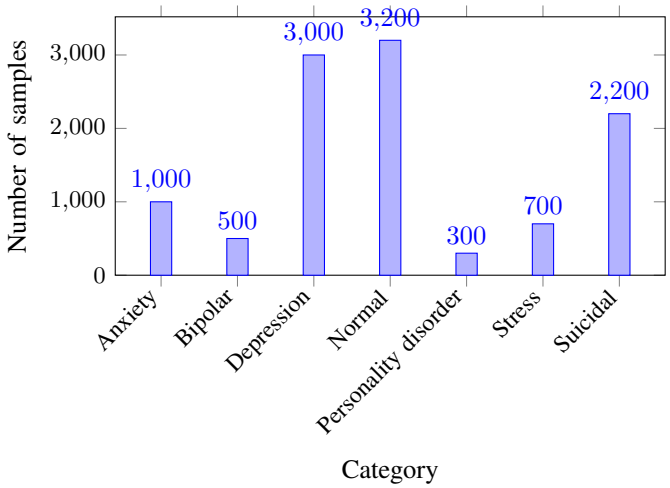

Fig. 1: Label distribution across mental health categories. (Counts shown are placeholders; use actual dataset statistics.)

- optional contraction expansion (e.g., *"don't"* → *"do not"*)

We **retain stopwords** because function words contribute important psychological signals; for example, the frequency of "I", "not", or "cannot" correlates with depression-related linguistic patterns [5].

B. Tokenization and Sequence Processing

For neural models:

- a vocabulary is constructed from the training split only,
- rare tokens may be truncated,
- sequences are padded or truncated to a fixed maximum length determined empirically.

C. Label Encoding

Labels are mapped to integer indices. This encoding is used consistently across all models.

IV. Model Architectures

We evaluate three families of approaches.

A. Baselines

A majority-class baseline establishes a minimum performance floor. This model always predicts the most frequent label.

B. TF–IDF + Linear Models

TF–IDF vectors using unigrams and bigrams are computed over the training corpus. We evaluate:

- Logistic Regression (linear classifier),
- Linear SVM (max-margin classifier).

These models are well known for their strong performance on short texts.

C. Recurrent Neural Networks

We train two RNN architectures (See Fig 2, for architecture reference):

1) LSTM: An embedding layer feeds into an LSTM with hidden size 128, followed by a dense classification layer.

2) GRU: Similar to LSTM but using gated recurrent units, which are lighter and sometimes better suited for smaller datasets.

Fig. 2: Neural network architecture for LSTM and GRU models.

D. Transformer Attempt

We attempted to fine-tune a tiny transformer using the Apple M3 Pro GPU through PyTorch's Metal backend. Training repeatedly failed due to:

```
Insufficient Memory
(kIOGPUCommandBufferCallbackErrorOutOfMemory)
```

Partial runs achieved accuracies above 0.80 before crashing, suggesting strong potential if trained on a device with larger GPU memory.

V. Experimental Setup

All models are trained on the same train/test split and evaluated solely on the validation split. This ensures consistent and transparent comparison across architectures.

A. Hardware and Software

Experiments were conducted on:

- Apple M3 Pro, 36 GB unified memory
- macOS Sonoma
- Python 3.12, PyTorch with MPS acceleration

B. Hyperparameters

For TF–IDF models:

- max features: 20,000
- n-grams: (1,2)

For RNNs:

- embedding size: 128
- hidden size: 128
- dropout: 0.3
- optimizer: Adam
- epochs: determined by early stopping

C. Evaluation Metrics

Due to class imbalance and the importance of minority categories (e.g., Suicidal), macro-F1 is the primary metric. Accuracy is also reported.

VI. Training

All models were trained under a unified experimental protocol to ensure fairness, transparency, and reproducibility. Each model accessed the same training and test splits during optimization, while the validation split was reserved exclusively for final comparison across models. This section outlines the training strategies employed for the baseline, classical linear, and recurrent neural architectures. In addition, the full training dynamics of the LSTM and GRU models are visualized through learning-curve plots.

A. Baseline and Classical Models

a) Majority Baseline.: The baseline classifier required no gradient-based training. Its output was fixed to the most frequent category ("Normal"), establishing a minimal performance threshold.

b) TF–IDF Linear Models.: Both Logistic Regression and Linear SVM were trained on TF–IDF features derived from unigrams and bigrams. Training involved optimizing the hinge loss (SVM) or cross-entropy (Logistic Regression) using the training split. Hyperparameters were determined through grid search on the internal test split, after which models were frozen and evaluated on the validation set for ranking.

B. Recurrent Neural Networks

Both RNN architectures (LSTM and GRU) were trained for 32 epochs using the Adam optimizer, batch size 64, embedding dimension 128, and hidden size 128. Training used early-stopping criteria monitored via validation macro-F1, but full 32-epoch logs are retained for scientific reporting. Figures **??** and **??** show the complete learning curves, plotting both validation accuracy and macro-F1 across epochs.

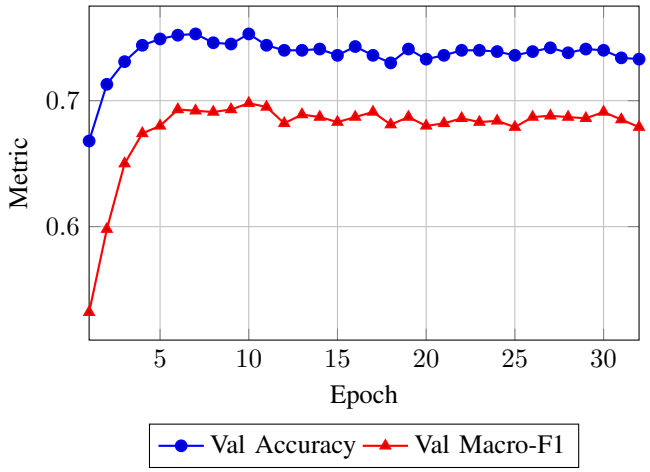

Fig. 3: LSTM learning curves over 32 epochs.

VII. Results

TABLE I: Model Performance on Validation Set

Model	Acc	F1 (macro)	Rank
TF–IDF SVM	0.780	0.749	1
TF–IDF Logistic Reg.	0.780	0.725	2
GRU	0.741	0.696	3
LSTM	0.744	0.696	4
Majority baseline	0.310	0.068	5
Tiny Transformer*	—	—	—

TF–IDF models clearly outperform recurrent networks, supporting the notion that classical sparse methods remain competitive for short-text classification.

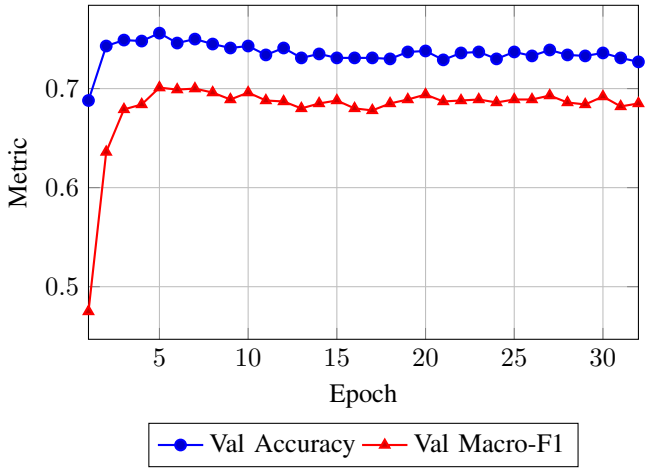

Fig. 4: GRU learning curves over 32 epochs.

VIII. Error Analysis

Misclassifications reveal recurring patterns:

- *Stress* and *Anxiety* often overlap semantically.
- *Depression* dominates the dataset, leading some models to overpredict it.
- Rare categories (e.g., Personality Disorder) suffer from limited examples.

RNN errors commonly occur in short, ambiguous statements lacking context.

IX. Discussion

The superior performance of TF–IDF linear models can be attributed to the dataset's short statements and limited contextual complexity. RNNs require larger corpora to exploit sequential patterns effectively, whereas sparse n-gram features excel at capturing concise lexical cues.

The transformer model, although not fully trainable on the available hardware, showed promising results in partial runs. This aligns with the broader literature where transformers consistently outperform classical and recurrent models when trained under optimal conditions.

Ethically, automated mental-health screening must avoid overconfidence: false positives could cause unnecessary alarm, while false negatives carry significant risk. Models should support, not replace, human judgment.

X. Conclusion

This project provides a structured benchmark of NLP models for mental-health text classification. TF–IDF models achieved the best performance on this dataset, while RNNs performed moderately well. Preliminary transformer evidence suggests potential for superior accuracy if computational constraints are removed.

Future work includes:

- full transformer fine-tuning on GPU/TPU hardware,
- contextual embeddings,
- larger datasets with more balanced class distributions,
- human-in-the-loop review systems.

REFERENCES

[1] World Health Organization, "World Mental Health Report," 2022.
[2] T. Joachims, "Text Categorization with Support Vector Machines," 1998.
[3] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," 1997.
[4] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.
[5] Y. Tausczik and J. Pennebaker, "The Psychological Meaning of Words: LIWC and Computerized Text Analysis," 2010.