Mental Health Micro Trend Detection at Community Level

Naveen Sai Tamanampudi
College of Information
University of North Texas
Denton, TX, USA
ORCID: 0009-0009-4521-2975

Rajiv Adusumalli
College of Information
University of North Texas
Denton, TX, USA
RajivAdusumalli@my.unt.edu

Thanmmay Yanamadala
College of Information
University of North Texas
Denton, TX, USA
ThanmmayYanamadala@my.unt.edu

Shyamala Reddy Dharmala
College of Information
University of North Texas
Denton, TX, USA
ShyamalaReddyDharmala@my.unt.edu

Dr. Clifford Whitworth
College of Information
University of North Texas
Denton, TX, USA
Cliff.Whitworth@unt.edu

Ravi Varma Kumar Bevara
College of Information
University of North Texas
Denton, TX, USA
RaviVarmaKumarBevara@my.unt.edu

Laxmigayathri Challa
College of Information
University of North Texas
Denton, TX, USA
LaxmigayathriChalla@my.unt.edu

Jiyuan Li
College of Information
University of North Texas
Denton, TX, USA
JiyuanLi@my.unt.edu

Abstract*—Mental health concerns such as anxiety, stress, depression, etc. are increasingly shared through online platforms like reddit, twitter, togetherall, etc. The current research in this domain focuses on individual classification (user-level or post-level) rather than community level. As a result, researchers and policymakers lack tools to detect short-term spikes or shifts in mental health signals across groups. To address this issue, our project focuses on building an application to track and analyze levels of distress in 7 classes of mental health in subreddits. Since the application need to work in real-time, we focused on fine-tuning light weight transformer-based classification models i.e. DistilBERT uncased, DistilBERT cased, Mental-RoBERTa and DeBERTa-v3 using a pre-classified reddit posts dataset from HuggingFace. We achieved an accuracy of 85.22% using Mental-RoBERTa and 84.91% using DeBERTa-v3. Mental-RoBERTa is performing better than DeBERTa in every category except for "Suicidal". Since "Suicidal" is the most severe of the 7 categories, we prioritized DeBERTa-v3. We utilized the Reddit API to extract posts from the subreddit of users choice, preprocessed, and classified using Fine-Tuned DeBERTa-v3. We performed time series analysis on the sentiment scores extracted using VADER. The application aggregates score to daily-level to detect spike using z-scores on rolling average of sentiment scores over a 7-day window. Once the spike is detected, the user can perform Root Cause Analysis to identify possible underlying causes such as seasonal events or economic factors. We clustered posts using K-Means Clustering, extracted top-15 keywords from each category, and used Phi-3-mini-4k-instruct to identify the root cause. We sampled 100 posts for each category, clustered them into 2 groups, and manually evaluated them. We used a binary agreement score to quantify the congruence between manual qualitative analysis and the rationale generated by the model and achieved a 78.5% score. We included SHAP plot to improve the explainability of the classification model. Our project classifies, detects anomalies in sentiment and identify root causes at a community level, shown in an interactive website that uncovers the patterns for specific subreddits.

***Index Terms*—Mental Health, Transformers, SHAP, VADER, Sentiment Analysis, Time Series Analysis, Anomaly Detection**

I. Introduction

Mental Health is often not considered as important as physical health. But mental health determines how a human functions in a day. According to [1], in 2024, 23.4% of US adults faced some degree of mental illness. Thats almost 1 person in every 4 individuals. People experience these mental health conditions due to their personal and professional life. Anxiery, Stress and Depression are the most common out of these. But many don't want to share these with family and friends with a fear of being judged. Depression alone increased from 8.2% in 2014 to 13.1% in 2023 [2]. With the rise of social media, more people were able to share their experiences freely. Unlike Facebook or LinkedIn where everyone knows who you are, Reddit allows users to stay anonymous. This anonymity makes people feel safer to open up about their honest feelings without the fear of being judged by friends or family.

Sometimes these experiences are not only confined to an individual. This might determine the overall health of the community. New government policies, lack of jobs, exams seasons etc can be the driving forces. Studies [3] showed that content shared by people on Facebook can actually be used to detect future occurrence of Depression. Also, technology has come a long way in helping us understand these conversations.

In the past, computers mostly just counted negative words to find sad posts. But now, with transformer based models like BERT and RoBERTa, we can actually understand the context and the deeper meaning behind what someone is writing, rather than just looking at keywords.

However, previous studies have focused on individual classification determining whether a single user or a single post exhibits signs of mental illness. While valuable, this misses the broader picture on how the collective emotional state of a community and how it fluctuates over time. Mental health doesn't happen in a bubble. If a whole group of students is stressed at the same time, it might be because of exam season. If a specific community shows a spike in anxiety, it could be due to a real-world event. By looking at the timeline of a whole community instead of just one person, we can spot these trends and understand the root causes better.

II. Problem Statement

The existing methods solely focuses on classifying individual posts into different mental health categories [4] [5] [6]. We do not have any tools to identify these trends at community level. The current methods cannot identify the reason behind the distress caused in the community. We would like this space in the research. A tool which can identify the above mentioned things like this can enable people, policy makers, subreddit moderators, researchers to understand what's causing the distress in the community. Authorities can understand if any of their policies is causing tension in the community,

III. Literature Review

These authors [4] classified depression posts on Reddit using Logistic Regression combined with TF-IDF. It just counts important words. They obtained an F1 score of about 70%, This shows that mental health can be identified by just looking at the pattern of the words. The issue is that they missed the advanced transformer models. Also, this is a binary classification. So, it cannot differentiate between different mental health issues.

The authors [5] created a model called "MM-EMOG" which combines BERT embeddings with emotion lexicons to classify mental health issues. While they claimed it was a huge improvement, there was no actual mention of the accuracy. For MentalBERT and BERT, it is varying from 71% to 79% .

The authors [6] created "Contextual Emotional Transformer Model" (CETM). They used the base RoBERTa and BERT and added an extra emotional attention layer at the end. But this work also focus on individual level posts. This cannot identify when the distress is happening and whats causing the distress.

The paper [7] used a graph-based model called TextGCN and used emotion features to identify depression. They worked on 5 datasets and got accuracy ranging from 78% to 91%. Again this is just a binary classification like [4] and [8]. [8] achieved better accuracy.

This paper [8] was actually the closest to what we are trying to do. They built a "Multi-Class Multi-Level" model that could classify posts into different categories like ADHD, PTSD, and Bipolar using RoBERTa. They compared machine learning, deep learning, and transformer-based transfer learning models. They achieved strong—98% accuracy for Level-1 which is binary clasification and 85% for Level-2 which is a multi class classification.

In the work [9], the authors utilized a curated dataset of 17000 posts from 13 subreddits to classify DSM recognized disorders using LSTM, BERT and RoBERTa. They achieved 89% accuracy using RoBERTa by combining the post titles and text. We extend this work by including DSM recognized symptoms as well. A single post might fall into multiple categories since these symptoms also cause disorders. So, we expect a slight drop in performance.

In the work [10], the authors focused on detecting 8 psychiatric conditions like Schizophrenia, BPD, and GAD using BERT based classification models. They performed 2 levels of analysis. One is user level and the other is post level. They achieved accuracy and f1 scores between 56% and 70% on post level predictions. The dataset we collected focuses on different set of disorders barring Bipolar. Moreover, we are sticking with the post level work to not violate the privacy of the users.

The paper [11] explores how social media can effectively detect Major Depressive Disorder (MDD). The authord crowdsourced a group of Twitter users with diagnosed depression and tracked their posting habits for the year which drove their diagnosis. They found clear behavioral red flags like social withdrawal, negative emotions, insomnia, and a spike in using words like "I" and "me." By feeding these signals into a model, they successfully predicted depression risk with 70% accuracy. Even though the study uses older methods and focuses on twitter, this provides a good base for this project.

The authors propose a framework [12] that combines pretrained language models with a multi-perspective prompt ensemble. They asked the model to assume the role of sociologist, psychologist, and educator and analyzed data. They addressed the challenge of unstructured social media data by introducing a self-supervised task to enhance robustness where the model learns to identify nuances of the internet data. This method achieved an F1-score of 79.18% on the MultiWD dataset outperforming standard BERT models and Llama-2.

The paper [13] focused specifically on the detection of stress, a symptom to many severe conditions. They used a curated dataset of over 190,000 Reddit posts gathered from five different domains, including interpersonal conflict, financial anxiety, and PTSD. They modeled stress detection as a supervised learning task. Their work demonstrated that BERT-based models outperformed traditional feature-based classifiers like LIWC. In our work, we combined both disorders and symptoms since categorizing symptoms like stress into anxiety disorder can be harsh. The stress is caused by some external pressure and temporary where as anxiety occurs over a prolonged period. So, making this distinction is clearly necessary.

One common pattern in all the existing studies is that they are limited to predicting whether a post contain any text related to a mental health category. It is always confined to user level

and never knew the condition of the community. Our project advances these works by performing time series analysis on sentiment scores to detect anomalies in the distress. After identifying that, we will also show the reasons for the distress. This will help the policy makers, subreddit moderators etc.

IV. Objectives

The objectives of this project are as follows:

- Identify the best light weigh transformer based classification model to classify posts into different categories in real time.
- Assign distress related scores to the live data after preprocessing and classification.
- Detect micro-trends in a subreddit for a particular category using time series analysis on aggregated sentiment scores to identify how mental health is progressing over time.
- Identify the common patterns causing mental health issues in the subreddit.
- Build a web application to integrate all these modules to view results.

A. What is Micro-Trend?

We define a micro-trend in the mental health domain as a short-term surge in the sentiment scores of a particular mental health category within a subreddit. Operationally, we will calculate the sentiment scores on a daily basis. We use an anomaly detection metric, i.e. z-score to identify these micro-trends. If any z-score exceeds 2 standard deviations within a rolling 7-day time-window, it will be marked as an anomaly. The user can run Root Cause Analysis (RCA) to extract the reasons behind it. The user can also change the anomaly sensitivity to make the system more rigid.

B. Why Reddit?

We have many social media platforms like reddit, quora, togetherall etc. But we chose reddit because of 3 reasons. First, different platforms have different forms of expressions. So, we want to work on only 1 platform. Secondly, there is abundant data on reddit (either classified or raw). Finally, reddit already follows a community structure. Each subreddit represents a community. So, we believe reddit is the ideal choice for this project. We will try to transition to other platforms or integrate them in the same application in the future works.

V. Models Used

A. Transformers

Before the introduction of transformers, NLP was dominated by Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks. These models processed data sequentially, reading text one word at a time. Because of this, the training used to be computationally inefficient because the multiple pieces of text cannot be processed at one. Moreover, the models struggled with long-range dependencies, often forgetting the context of the beginning of a sentence by the time they reached the end.

In 2017, Transformer architecture was introduced for the first time. This new architecture replaced recursion with a mechanism called Self-Attention. This allowed the model to process an entire sequence of words parallelly rather than sequentially. This reduced the training times drastically. Models were also able to capture complex relationships between distant words in a sentence. In traditional models, all nearby words are treated with fixed importance. But in self-attention mechanism, the model dynamically weighs the relevance of every word in a sentence to every other word in the sentence, This works even when the words are far apart. Through self-attention, the Transformer assigns an attention score to the connect the words of relevance/similarity to solve the issue of ambiguity and understanding the context. This scoring occurs for every token parallelly resulting in a dense matrix of contextual understanding.

Since the Transformer processes all words in a sentence at a time, they don't have any order. It just sees the text like a bag of words rather than a structured sentence. To solve this, the transformer architecture uses Positional Encodings and appends them to the input embeddings. The positional encodings are mathematical vectors that give the model specific information about the relative or absolute position of each word. This allows the model to understand sentences and distinguish between different sentences having same words.

The original Transformer architecture consists of two types of networks. They are Encoder and the Decoder. The Encoder uses self-attention to weigh the importance of different tokens with respect to other tokens in the sentence. It is responsible for reading and understanding the input text, compressing it into numerical context vectors. The Decoder then uses this representation to generate output one word at a time. This architecture is useful for tasks like translation or text summarization. But in classification tasks, text generation is not needed. That is why models like BERT, DistilBERT, Mental-RoBERTa etc. only have the Encoder network. This allows the model to understand the text and classify them effectively. We used 3 types of BERT models in this project.

1) DistilBERT: DistilBERT (Distilled BERT) is a simpler version of the original BERT model. It uses a technique called knowledge distillation to reduce its size. In this, a smaller model (DistilBERT) is trained to replicate the behavior of a larger model (BERT-Base). DistilBERT reduces the parameter count to 66 million by reducing the number of transformer layers by half. This makes the model 40% smaller than original BERT model while still retaining around 97% of its language understanding capabilities. It also executes 60% faster. This architecture is useful in projects where speed is prioritized over capturing subtle details.

2) Mental-RoBERTa: Mental-RoBERTa is a specially trained variant of RoBERTa (Robustly Optimized BERT) that went through Domain-Adaptive Pre-Training (DAPT). It has 86 million parameters. The standard models like BERT, DistilBERT, RoBERTa are trained on general corpora like Wikipedia and BookCorpus dataset whereas Mental-RoBERTa was further pre-trained on a massive dataset of Reddit posts

specifically from mental health communities. This allows the model to understand the unique lexicon, slang, and emotional patterns of online distress. With the help of this domain knowledge, Mental-RoBERTa is expected to achieve better performance than general models like DistilBERT

3) DeBERTa: DeBERTa(Decoding-enhanced BERT with disentangled attention) uses a different architecture unlike BERT. Traditional BERT based models combine word content and position into a single vector. But DeBERTa uses Disentangled Attention mechanism which represents each word using two separate vectors. One for its content and the other for its relative position. This allows the model to better understand the dependency of words based on their distance and ordering. This is especially useful for detecting negation words like "not" and conflicting sentiments often found in longer posts. Since we are using the base model, it has 100 million parameters. DeBERTa is expected to perform better than DistilBERT. But we will see if it performs any better than Mental-RoBERTa.

B. VADER

VADER (Valence Aware Dictionary for sEntiment Reasoning) is a rule-based sentiment analysis tool. It uses a lexicon and a set of grammatical rules to identify the sentiment of text between -1 and 1 with -1 being the most negative and +1 being the most positive. Many standard sentiment classifiers score the text based only on word polarity. But VADER is specifically trained on social media data. So, it is capable of understanding contexts and nuances common in online platforms such as capitalization (e.g., "BAD" vs. "bad"), punctuation intensity (e.g., "!!!"), and emojis. VADER returns 4 scores for each piece of text. It says how much percentage of text is negative, neutral and positive. Along with that, it returns a compound score which states the overall sentiment of the text. It is computed by summing the valence scores of each word in the lexicon, adjusted according to VADER's rules (like capitalization and punctuation), and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive).

C. Phi-3-Mini-4k-Instruct

Microsoft's Phi-3-Mini-4K-Instruct is a Small Language Model which only contains a dense decoder-only Transformer architecture. It has 3.8 billion parameters. Unlike traditional Large Language Models (LLMs) that rely on massive parameter counts and huge volumes of internet, Phi-3 reasoning abilities are because of its training on highly curated dataset of 3.3 trillion tokens of textbook-quality data. This allows it to achieve reasoning benchmarks comparable to models like GPT-3.5 which is 10 times bigger. Major advantage with this model is its size which is computationally efficient to run on laptop/computer hardware. We used the Instruct variant because of its ability to comply with the formatting conditions. We want the output to be formatted as title and explanation which is later integrated into the web application consistently.

D. SHapley Additive exPlanations (SHAP)

All the transformer models used in the prior phases are kind of "black box" in nature. It is really hard to explain why a certain model arrived at a particular conclusion. To make our conclusions transparent and clinically valid, we are using SHapley Additive exPlanations (SHAP) in this project. It is a game theory method that quantifies the contribution of every word to a final prediction. By calculating Shapley values, the model assigns a positive or negative score to each token. This lets us view in a visual way to see exactly which terms pushed the model toward a specific classification and which models pushed against that class.

VI. Methodology

Fig. 1 shows the architecture of the system.

A. Data Collection

We used 2 sources to collect data. We used the pre-classified dataset [15] from HuggingFace to fine tune the classification models. It has 53042 posts and belongs to 7 classes. They are Bipolar Disorder, Depression, Anxiety, Stress, Normal, Personality Disorder, and Suicidal. The distribution of classes in the dataset can be seen in Table- I

TABLE I: Dataset Distribution

ID	Class	Number of Posts
0	Anxiety	3888
1	Normal	16351
2	Depression	15404
3	Suicidal	10652
4	Stress	2669
5	Bipolar	2877
6	Personality Disorder	1201

We extracted posts using reddit API. For each subreddit, we collected post id, post title, text, author and created time. Here author is not used anywhere but collected for the future extension of the project. We used PRAW library to connect to the reddit API and extract posts. The reddit API limits us to extract 1000 recent posts. If the subreddit is very active, its just 2-3 days worth of data and not very useful. To solve this issue, we created a data extraction script and used run it every day. We created a Directed Acyclic Graph (DAG) to run the script at 12:00 AM every day.

We identified 14 subreddits which we believe might contain mental health activity. We are just taking 14 subreddits since we are transitioning to community level for the first time. We took mental-health-focused communities (r/askatherapist, r/Anxiety, r/mentalhealth, r/SuicideWatch, r/bipolar, r/depression, r/selfhelp, r/TalkTherapy) and general subreddits (r/careerguidance, r/GradSchool, r/cscareerquestions, r/recruitinghell, r/jobs)

B. Data Preprocessing

Data preprocessing is a very important step when dealing with text data collected from online sources. The data include

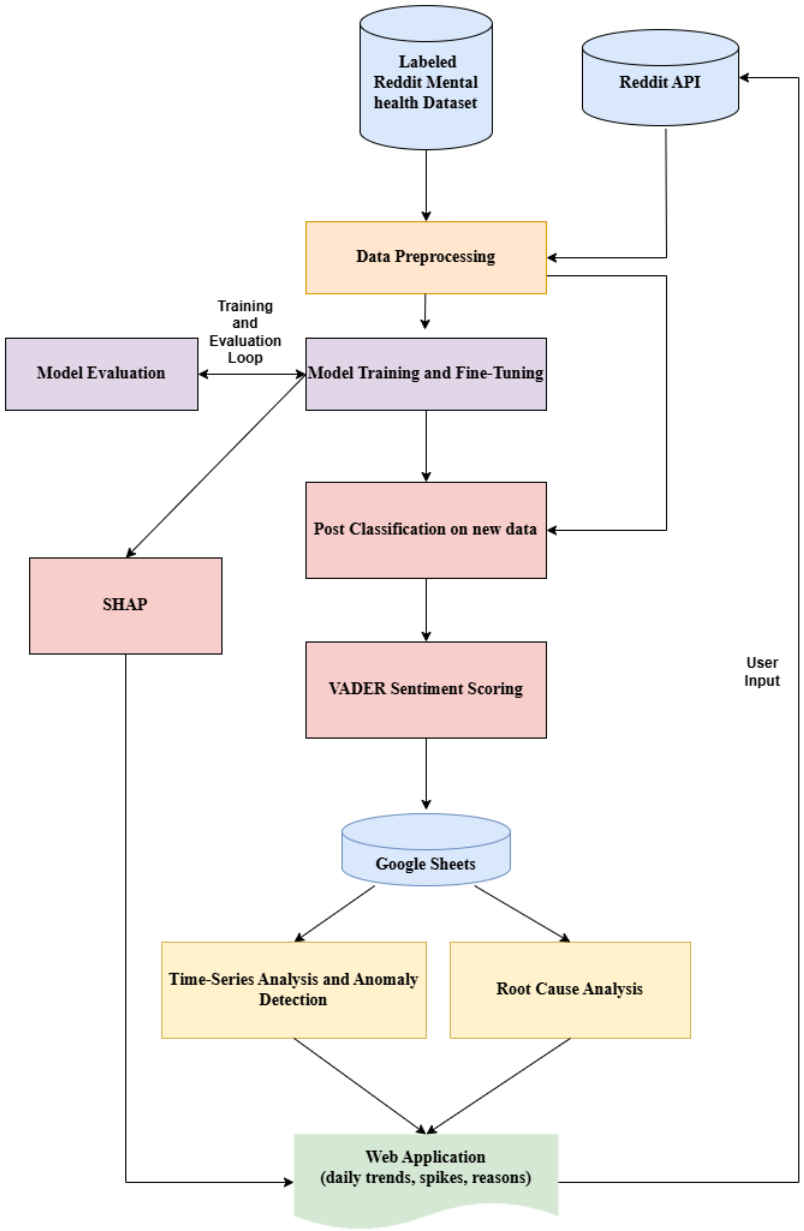

Fig. 1: System Architecture

a lot of inconsistencies that might hinder the performance of the models. We perform the following tasks to clean the data.

1) Converted all the data to string to ensure that the pipeline doesn't break if it encounters any non-string data (like just numbers).
2) Removed non-English posts since the analysis is limited to English language. This was done using the langid library.
3) Removed any posts with less than 3 characters since they are most likely acronyms and wouldn't impact on our analysis.
4) Removed any gender specific information by replacing them with neutral words. Examples: He/She → They, His/Her → Them, Father/Mother → Parent etc.
5) Removed nation specific information with the word "citizen" to make sure we don't pass any ethnicity bias.
6) Removed Links/URLs from the text since these get broken into many tokens and increase the vector size.
7) Removed user and subreddit references from the text to make the text anonymous.
8) Stripped extra white spaces from the text.

C. Model Fine-Tuning

Once we cleaned the data using the above mentioned the steps, we encoded the labels to numbers since these models can handle only numbers. We split the dataset into train and test sets in the ratio 80:20. Next, we loaded the tokenizer

from the pretrained models DistilBERT, Mental-RoBERTa, DeBERTa respectively, and used that tokenizer to tokenize our posts. DistilBERT and Mental-RoBERTa models has a maximum token length of 512. That roughly equates to 380 words. During preprocessing, we performed Exploratory Data Analysis (EDA) to view the distribution of lengths of posts in terms of tokens. We identified only 1% of the text longer than 512 tokens. So, we just truncated longer posts. Finally, we loaded the pretrained classification model and trained it on our data. The fine-tuning was performed using A100 GPU provided by Google Colab.

D. Model Evaluation

We evaluated the classification model using 5 metrics.

- *Accuracy:* What percent of posts were correctly classified.
- *Precision:* Out of all the posts classified into a particular class by the model, how many of them actually belong to that class.
- *Recall:* Out of all posts available in a particular class, how many of them were correctly identified by the model.
- *F1 Macro:* This is the Harmonic Mean between precision and recall. This will tell us how well a model performs across all categories.
- *Weighted-F1:* This is similar to F1 Macro but works well for imbalanced datasets. This will weigh the class based on its distribution in the dataset. This will teach the model what distribution to expect most of the times in the real world. Since the dataset in our project is imbalanced, we will use this metric to find the best model.

We fine-tuned the 2 variants of DistilBERT model. We tested if preserving original case improves the accuracy. We identified a slight improvement. So we used original case when fine-tuning Mental-RoBERTa and DeBERTa. We didn't try different combinations with these 2 models because they are heavier compared to DistilBERT and take 2-3 times more computation time.

E. Classifying Posts and Sentiment Scores

The preprocessing steps mentioned in the step VI-B are applied to the live reddit data as well. Once we classify the posts, we assigned the sentiment scores. To quantify the distress level in the posts, we used Valence Aware Dictionary and sEntiment Reasoner (VADER) sentiment analysis tool. By using this, we can convert the text into something quantitative which expresses the level of distress in the text. Later, we append the data to Google Sheets having dedicated worksheets for different subreddits. A python script was made for these tasks and the same DAG mentioned in VI-A was utilized for processing new posts.

F. Time Series Analysis

Time series analysis is very important, as individual sentiment scores of posts might fail to capture the temporal dynamics and shifts in community mental health. We computed 3 scores to analyze these signals

- *Distress Score:* Negated the sentiment scores and aggregated them to daily level by finding mean.
- *Volume:* Number of posts received
- *Distress Load:* Multiplication of distress score and volume.

These distress score and volume doesn't say much on their own. If a day has a single post yet have a distress score or 0.9, it will be marked as an anomaly. But in reality, it is truly not an anomaly and just an individual user experiencing it. To tackle this, we calculated another score called distress load which is a multiple of "distress score" and "volume". This will weigh out both values and flag anomalies if either of the scores sky rockets or both of them are moderately high.

We calculated average and standard deviation over a 7-day rolling window on these score. These help us calculate the day-level z-scores to quantify deviations from the norm. We set a threshold of 2 standard deviations to identify anomalies and classify them as statistically significant spikes or "micro-trends" that deviate from the community's recent baseline. This can prompt users to perform root cause analysis on a particular day.

G. Root Cause Analysis

We performed root cause analysis for each category independently to identify common reasons/patterns causing that category of mental health. It was done in the following steps:

- Encoded the text into numbers using all-MiniLM-L6-v2 sentence transformer.
- Clustered the posts into 2 groups using the K-Means Clustering algorithm.
- Converted the posts into tokens and represented each post based on the frequency of tokens using count vectorizer. We limited the maximum number of tokens to 1000 to avoid sparse matrices.
- For each cluster, we extracted top-15 tokens based on their frequency and passed to a Small Language Model (Microsoft Phi-3-mini-4k-instruct) to extract valid reasons from these keywords. We used an SLM instead of LLM since the model needs to run in real-time to support the web application.

H. SHAP

To deal with the black box nature of the Transformer model, we are using SHAP to improve the visibility and interpretability. We provided a text box in the web application which was pre-populated with an example text. Clicking on "Explain with SHAP" will run the model and produce a visualization which shows what text is pushing the text to be particular class. Now, we are not showing SHAP for each and every post since these can be searched online and traced back to the user. Instead, we made the text box editable so that user can input information of their liking and still see how our model approaches the classification.

TABLE II: Model Performance Comparison (%)

Model	Accuracy	Precision	Recall	F1-Macro	Weighted F1
DistilBERT-uncased	83.52	82.96	82.53	82.74	83.50
DistilBERT-cased	83.83	84.40	81.96	82.88	83.81
Mental-RoBERTa	85.22	85.52	85.23	85.34	85.16
DeBERTa-v3	84.91	84.25	85.85	84.92	85.00

VII. Results

A. Fine-Tuning Results

Table-II shows various metrics of different models. Mental-RoBERTa-v3 is give the best performance with 85.22% accuracy and 85.16% weighted f1 score, bettering DistilBERT by 1.7%. Higher performance can be attributed to more parameters and pre-training on mental health data. The fine-tuned Mental-RoBERTa is slightly better than DeBERTa-v3 at 84.91% accuracy. Since the dataset is imbalanced, we need to see the class-wise accuracy of different models. The Fig. 2, 3, 4, 5 shows the confusion matrices of 4 fine-tuned models on the test dataset.

The labels to the left indicates true labels and the ones to the bottom indicate predicted labels. From the 4 confusion matrices, we can see that all models performed exceptionally well in identifying "Normal" class. But there is a huge leap in performance for Mental-RoBERTa and DeBERTa-v3 when other classes are considered. In between Mental-RoBERTa and DeBERTa-v3, Mental-RoBERTa is performing slightly for all other classes except for "Suicidal". Mental-RoBERTa is giving 72% on Suicidal whereas DeBERTa-v3 is giving 79%. Since suicidal is more severe compared to other classes and DeBERTa-v3 is giving better performance, we used that model for classifying live reddit posts.

B. Root Cause Analysis

To evaluate the validity of reasons extracted by the SLM using the keywords extracted using K-Means Clustering, we performed manual analysis on a sample of 100 posts from each category (Simple Random Sampling Without Replacement). We divided them into 2 groups and analyzed them manually and identified the reasons. Then, we passed the same data through the application we created. We scored them a 1 if the reasons aligned, else 0. We achieved 11 points out of 14 i.e. 78.5%.

C. Web Application

The first page of the web application Fig. 6, 7, 8 shows the trend dashboard. To the left, there is a control bar which lets user filter the posts. The first dropdown menu contains the names of all 14 subreddits as seen in Fig. 11. User can select the subreddit of his/her choice and the data will be loaded from the google sheets. This data is cached to avoid further load operations if the user decide to come back to the same subreddit.

Next, there is a category selection drop down (Fig. 12 which contains the names of 6 classes in the dataset like Anxiery, Stress etc. We removed the option to select "Normal" class since that is not the focus of this study. Next, there is an anomaly selection slider where the user can select a number from 1.0 to 3.0. This number will determine the anomaly detection range for a particular subreddit and category.

The main page contains 3 charts. The first 2 charts can be seen in Fig. 6. The first one is a distress analysis chart which is the negated sentiment scores aggregated to daily level. This was filtered according to the subreddit selected and category selected. The anomaly sensitivity will determine the range of the anomaly detection. We used rolling z-scores instead of static z-scores to make the system adapt to the recent trends. By default, the anomaly slider is at 2 meaning if the z-score of any day crosses 2 standard deviations over a rolling window of 7 days, it will be marked as an anomaly. The range of anomaly detection can be changed by the user and the bands (grey shadings on the chart) will update accordingly. The second chart shows the volume of posts in a day. The distress load over the time can be seen in Fig. 7.

After identifying the anomalies, the user can select that particular date from the data widget in the control pane and click on "Find Root Cause". This will also work for a range of dates. All the posts in that range will be clustered into 2 groups and root causes will be identified using steps mentioned in VI-G. Next, the identified causes are laid out on the web application in an expander in the form of title and explanation. We are running root cause analysis on all available posts whenever the user changes the subreddit and category. The user can change the range of dates and filter the posts as needed. The bottom of the page contains SHAP where the use can enter text and identify what class it belongs to and which words are pushing a post to be a particular class. There is a disclaimer at the bottom of the screen stating "This project is not to be taken as a substitute for a psychologist. We recommend seeking the advice of a psychologist." as shown in Fig. 9

The second page of the web application Fig. 10 is for the "Live Reddit Data Analysis". If a user want to analyze any subreddit other than the 14 mentioned in the dropdown, the user can enter the subreddit name in the textbox provided in the control pane and click on the tick mark to start analyzing the subreddit. The application will connect to the API, extract posts from the subreddit, classify them, assign sentiment scores and perform the same operations mentioned in page-1. Apart from the textbox in place of the subreddit dropdown, the layout of page-2 is similar to page-1. One drawback with this is that reddit API allows the extraction of 1000 recent posts.

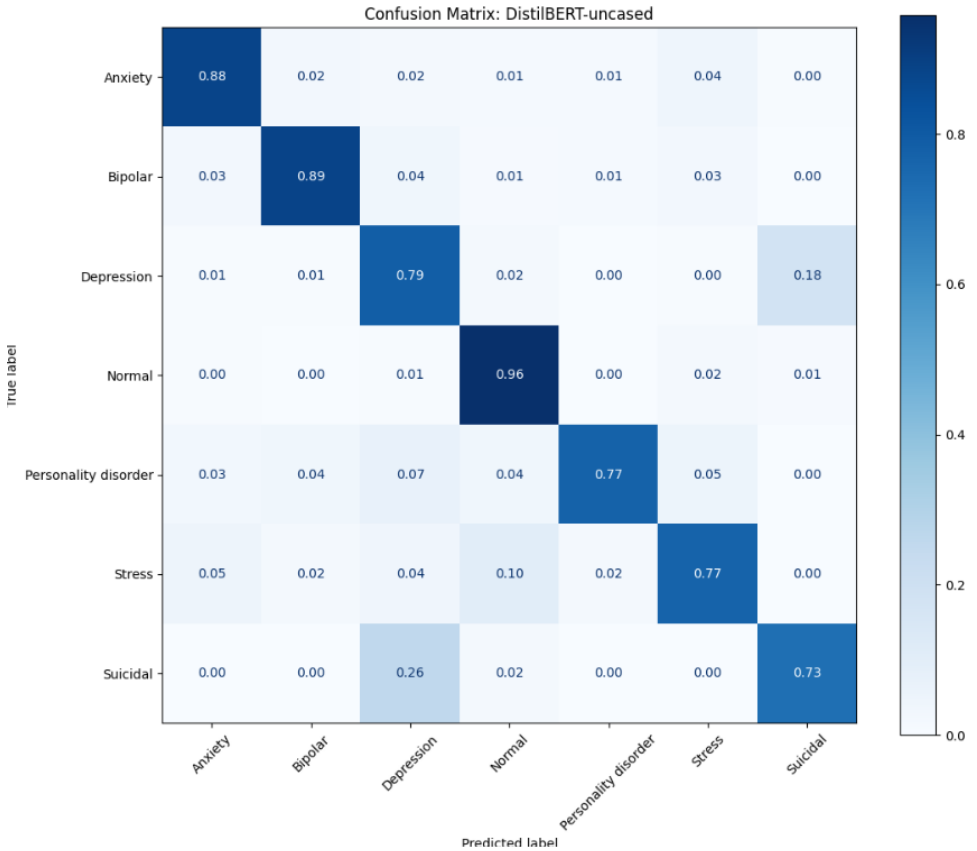

Fig. 2: DistilBERT Uncased Confusion Matrix

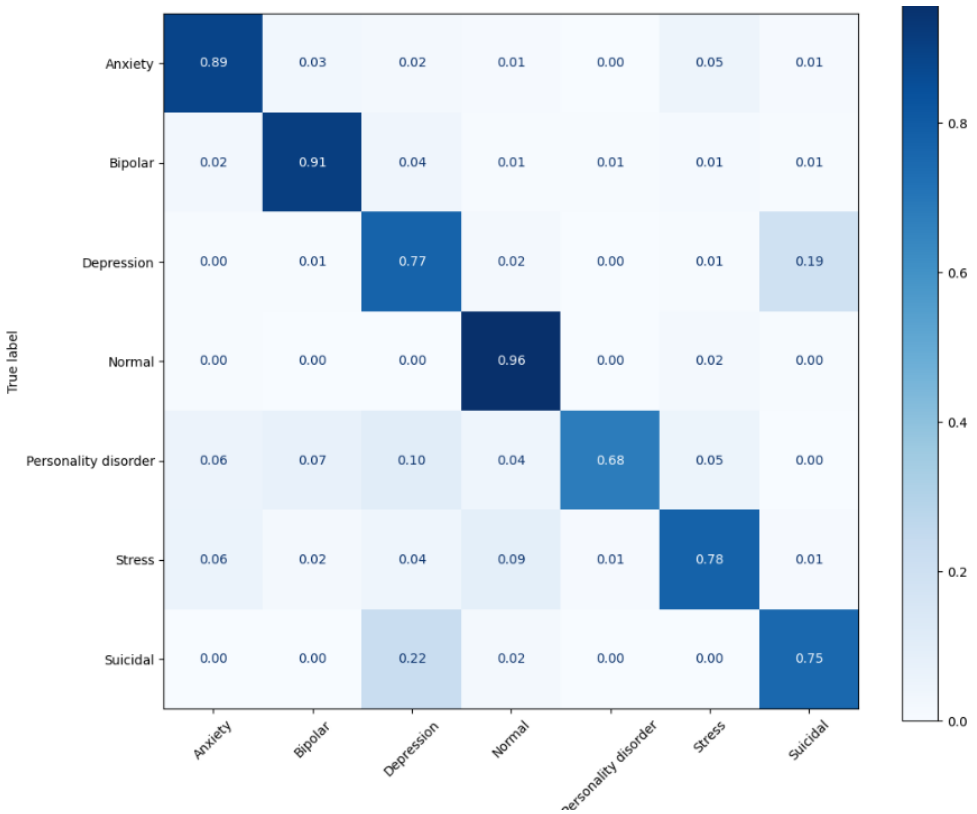

Fig. 3: DistilBERT Cased Confusion Matrix

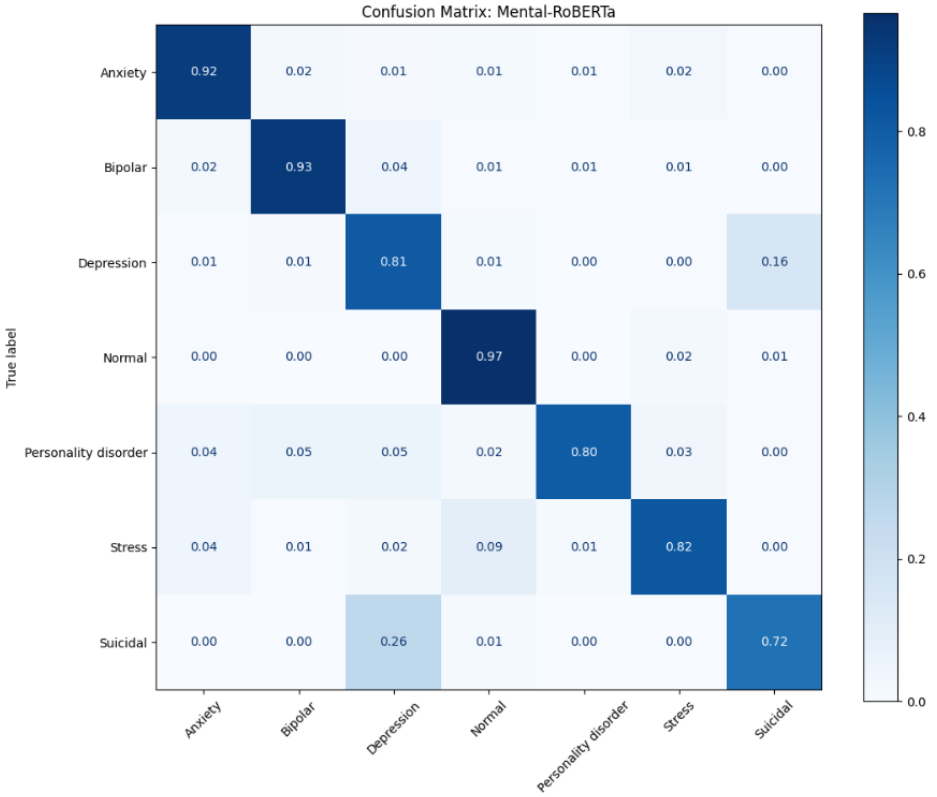

Fig. 4: Mental RoBERTa Confusion Matrix

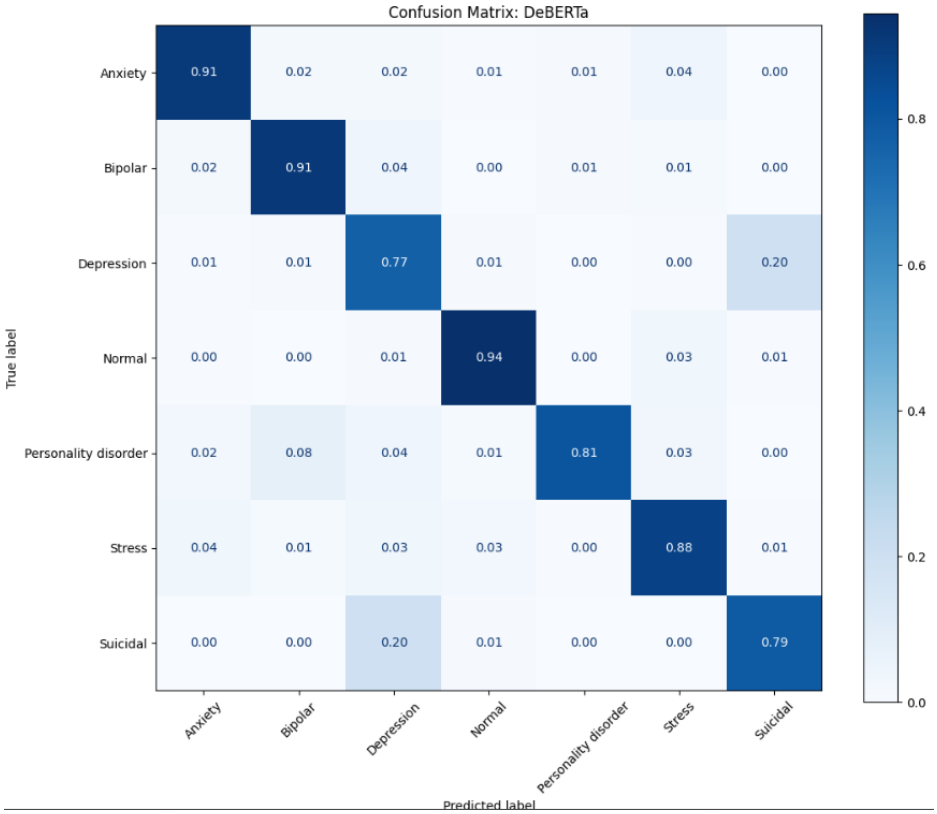

Fig. 5: DeBERTA Confusion Matrix

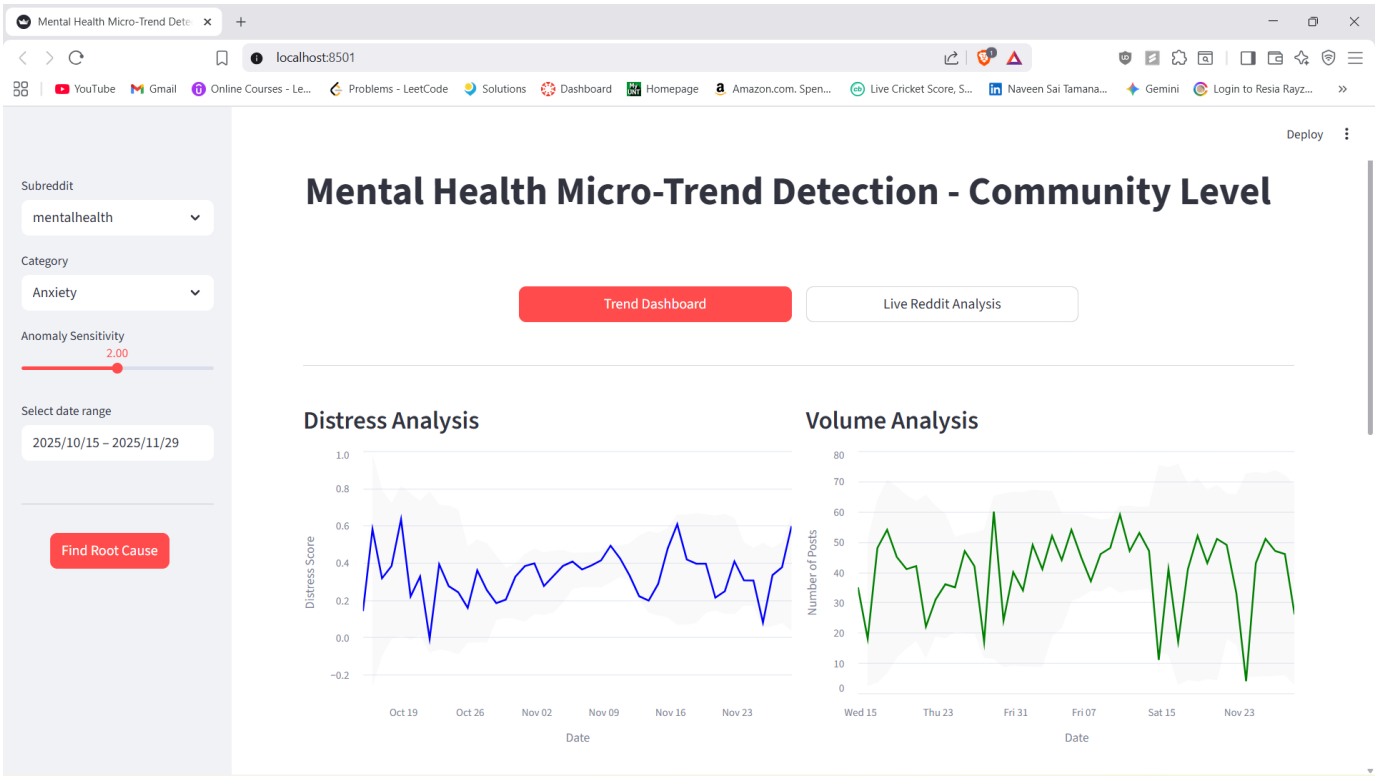

Fig. 6: Trend Dashboard showing time series analysis

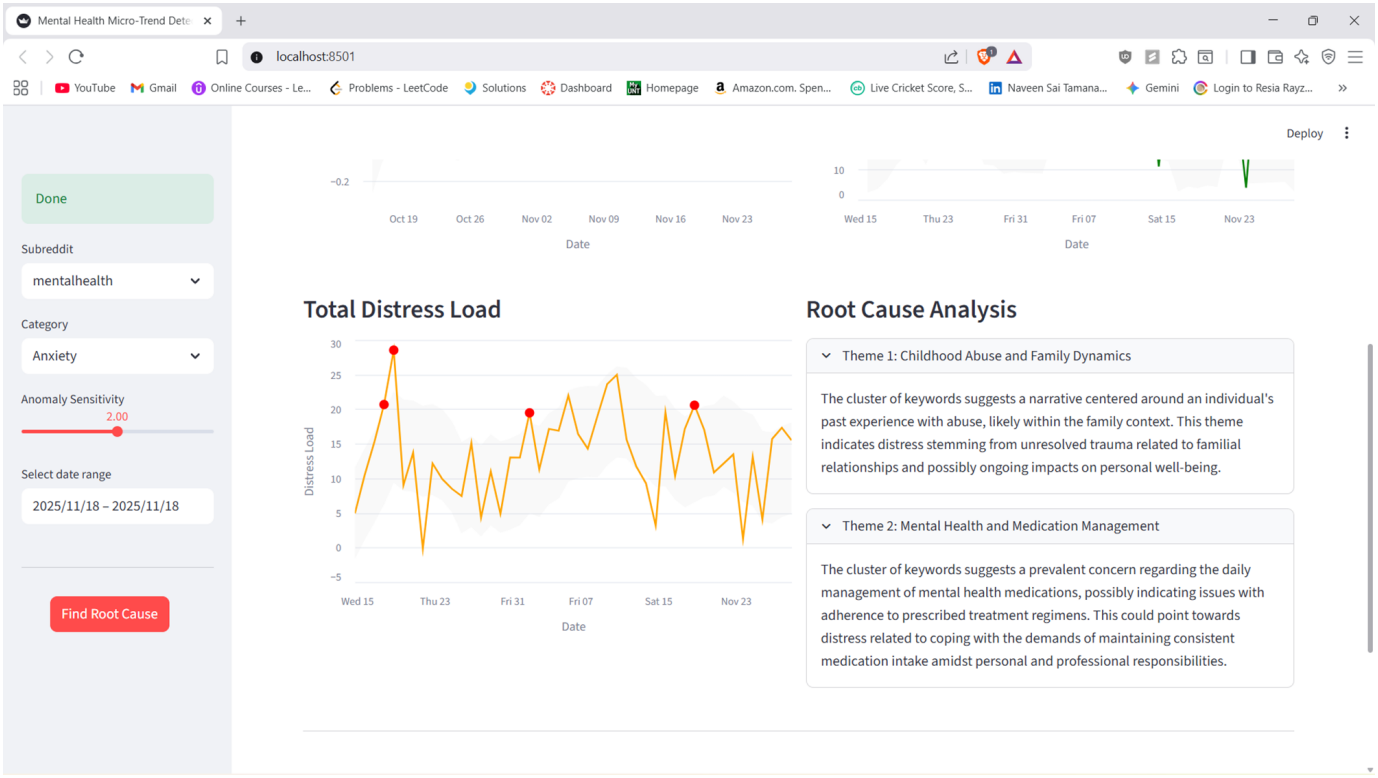

Fig. 7: Trend Dashboard showing Root Cause Analysis

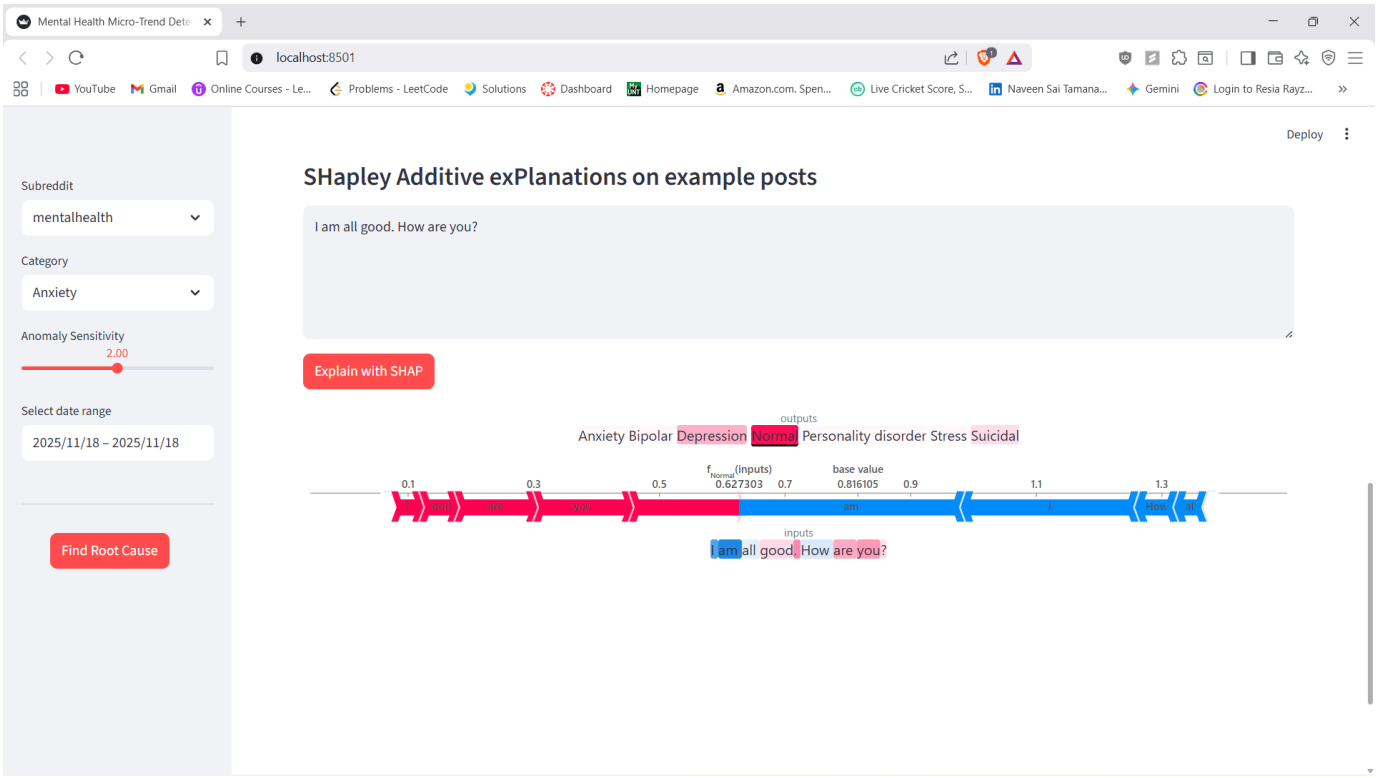

Fig. 8: Trend Dashboard showing SHAP

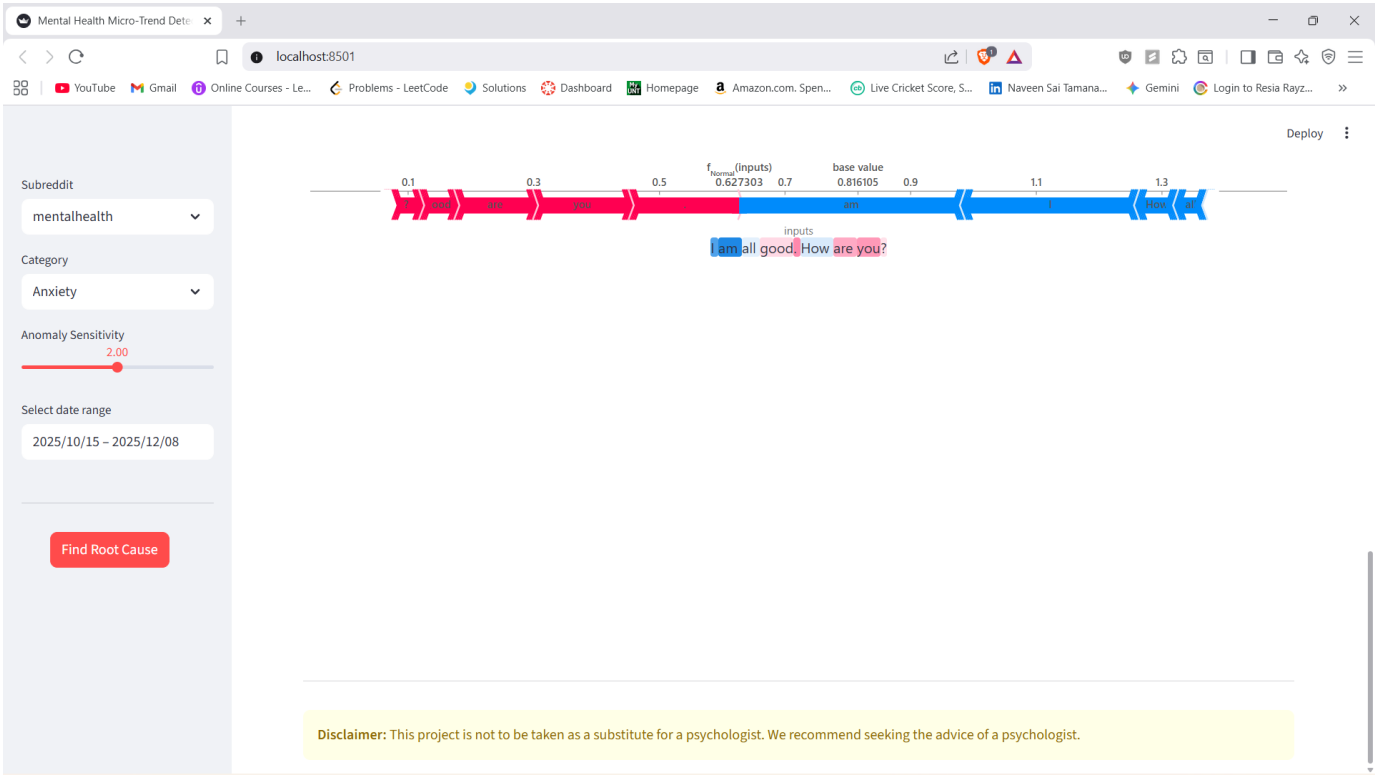

Fig. 9: Disclaimer at the end of the screen

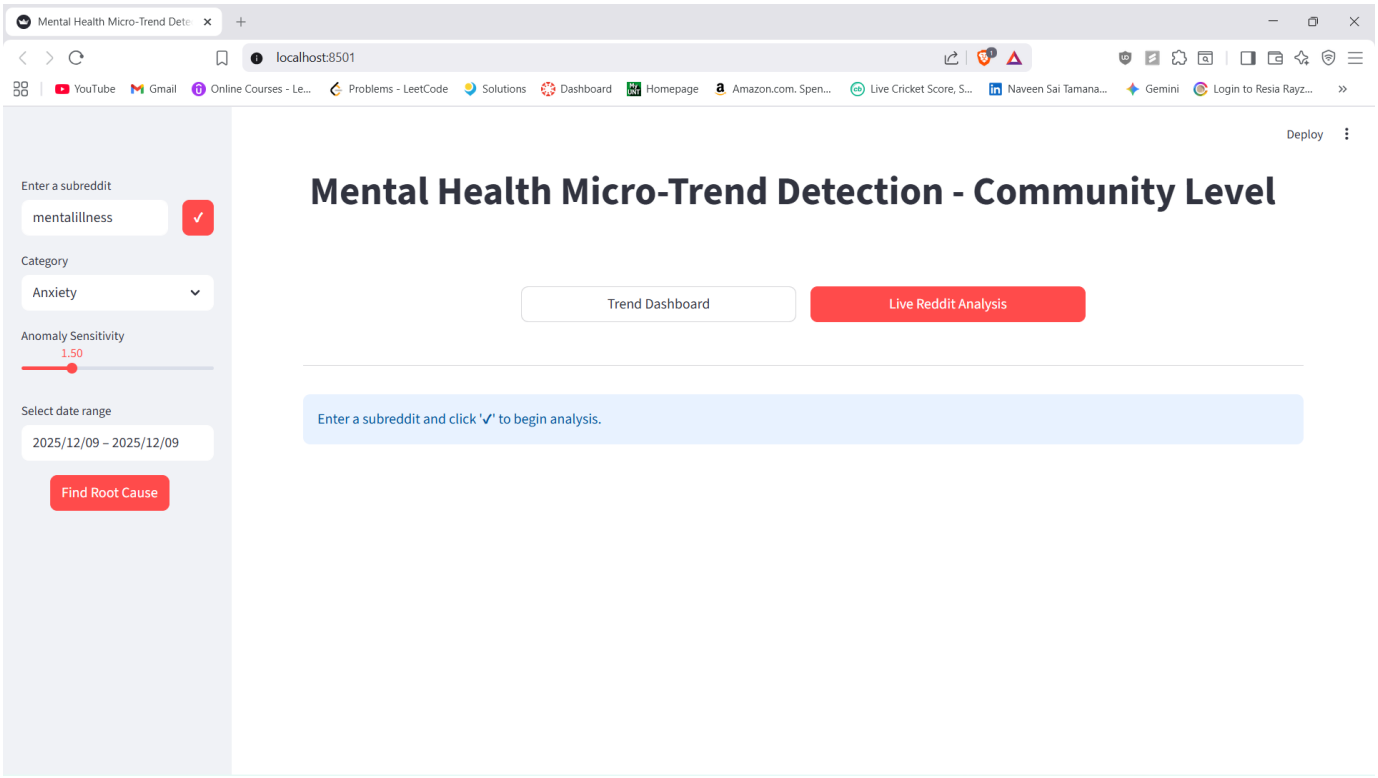

Fig. 10: Live Reddit Data Analysis Page

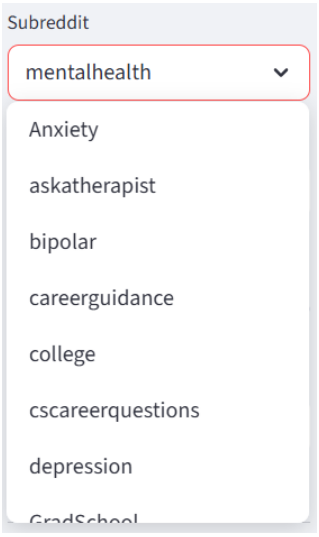

Fig. 11: Subreddit dropdown

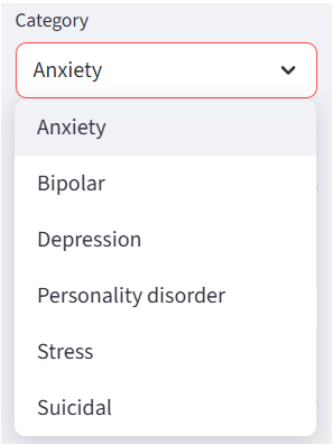

Fig. 12: Category dropdown

So, for an active subreddit, this might be 2-3 days worth of data. This might not produce a meaningful timeseries chart. Robust ways of extracting posts must be explored in the future.

VIII. Ethics, Bias and Safety

We took the following measures to make sure that the users privacy will not get violated.

- We followed IRB-style precautions to protect the identity of the users. We removed any user and subreddit references from the posts.
- We didn't show any post on our web application since anonymized post can still be traced back to the user just by searching it in the internet.
- We did not pass any bias explicitly to the models by replacing gender and regional terms with neutral words.
- We displayed a disclaimer stating "This project is not to be taken as a substitute for a psychologist. We recommend seeking the advice of a psychologist."

IX. Conclusion

To conclude, this project transitioned into a different world of mental health classification from post level prediction to community based detection to detect micro trend and root causes in a community. We got a close performance between Mental-RoBERTa and DeBERTa-v3. But because of its superior performance in "Suicidal" class, we selected "DeBERTa-v3" with 84.91% accuracy. We successfully detected micro-trends using the sentiment scores assigned by VADER. We identified the micro-trends using the z-scores. Finally, we made the complete transition by performing Root Cause Analysis using an SLM to find out the reasons behind the community level stress. We also included SHAP to explain how the model classify the text. All these components were brought together in the web application which enables easy anomaly detection and root cause identification for the policy makers, moderators and researchers in real-time.

Acknowledgment

We would like to thank Dr. Clifford Whitworth, Ravi Varma Kumar Bevara, Laxmigayathri Challa and Jiyuan Li for their continuous support and guidance. They provided valuable inputs at different stages of the project enabling us to refine our methodology and derive meaningful insights from our data

References

[1] National Alliance on Mental Illness, "Mental health by the numbers," National Alliance on Mental Illness, Apr. 2023. https://www.nami.org/about-mental-illness/mental-health-by-the-numbers/

[2] D. Brody and J. Hughes, "Prevalence of depression in adolescents and adults: United states, august 2021–august 2023," National Center for Health Statistics, vol. 527, no. 527, Mar. 2025, doi: https://doi.org/10.15620/cdc/174579.

[3] J. C. Eichstaedt et al., "Facebook language predicts depression in medical records," Proceedings of the National Academy of Sciences, vol. 115, no. 44, pp. 11203–11208, Oct. 2018, doi: https://doi.org/10.1073/pnas.1802331115.

[4] M. Madan, A. Agarwal, and S. Khan, "Reddit social media text analysis for depression prediction: using logistic regression with enhanced term frequency-inverse document frequency features," International Journal of Power Electronics and Drive Systems/International Journal of Electrical and Computer Engineering, vol. 14, no. 5, pp. 5998–5998, Aug. 2024, doi: https://doi.org/10.11591/ijece.v14i5.pp5998-6005.

[5] Rina Carines Cabral, Soyeon Caren Han, J. Poon, and Goran Nenadic, "MM-EMOG: Multi-Label Emotion Graph Representation for Mental Health Classification on Social Media," Robotics, vol. 13, no. 3, pp. 53–53, Mar. 2024, doi: https://doi.org/10.3390/robotics13030053.

[6] Ayodeji O.J. Ibitoye, O. O. Oladosu, and Olufade F.W. Onifade, "Contextual Emotional Transformer-based Model for Comment Analysis in Mental Health Case Prediction," Vietnam Journal of Computer Science, Oct. 2024, doi: https://doi.org/10.1142/s2196888824500192.

[7] H. Mao and Q. Han, "Enhancing TextGCN for depression detection on social media with emotion representation," Frontiers in Psychology, vol. 16, Aug. 2025, doi: https://doi.org/10.3389/fpsyg.2025.1612769.

[8] A. N. Sutranggono, R. Sarno, and I. Ghozali, "Multi-Class Multi-Level Classification of Mental Health Disorders Based on Textual Data from Social Media," Journal of ICT, vol. 23, no. 1, pp. 77–104, Jan. 2024, doi: https://doi.org/10.32890/jict2024.23.1.4.

[9] A. Murarka, B. Radhakrishnan, and S. Ravichandran, "Classification of mental illnesses on social media using RoBERTa," ACLWeb, Apr. 01, 2021. https://aclanthology.org/2021.louhi-1.7/

[10] M. D. Choudhury, M. Gamon, S. Counts, and E. Horvitz, "Predicting Depression via Social Media," Proceedings of the International AAAI Conference on Web and Social Media, vol. 7, no. 1, Jun. 2013, Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14432/14281

[11] Z. Jiang, S. I. Levitan, J. Zomick, and J. Hirschberg, "Detection of Mental Health from Reddit via Deep Contextualized Representations," Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis, 2020, doi: https://doi.org/10.18653/v1/2020.louhi-1.16.

[12] C.-Y. Hsieh, Q.-Y. Ye, F.-C. Liu, X. Wang, C.-H. Lee, and C.-S. Lin, "Multi-label mental health classification in social media posts with multi-perspective prompt ensemble and auxiliary self-supervision," Scientific reports, pp. 10.1038/s41598-02530873–x, May 2025, doi: https://doi.org/10.1038/s41598-025-30873-x.

[13] E. Turcan and K. McKeown, "Dreaddit: A Reddit Dataset for Stress Analysis in Social Media," arXiv:1911.00133 [cs], Oct. 2019, Available: https://arxiv.org/abs/1911.00133

[14] American Psychiatric Association, "Diagnostic and statistical manual of mental disorders," Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition, Text Revision (DSM-5-TR), vol. 5, no. 5, 2022, doi: https://doi.org/10.1176/appi.books.9780890425787.

[15] D. R. Suram, "dsuram/mistral-mental-health-lora at main," Huggingface.co, Jul. 2025. https://huggingface.co/dsuram/mistral-mental-health-lora/tree/main/workspace

[16] "Reddit ," Reddit.com, 2024. https://www.reddit.com/prefs/apps