Mental Illness Identification Using Transformer Language Models

Chirag Uday Kamath cukamath
Divya Maiya dmaiya
Neha Prakash nehaprakash
Akhil Arza aarza

1 Problem statement

Mental illness affects millions of people worldwide, yet many individuals suffer in silence due to social stigma and lack of access to mental health care. According to the World Health Organization (WHO), approximately 1 in 4 people in the world experience mental illness at some point in their lives (15). Despite the prevalence of mental illness, only 44.8% of US adults with mental illness received treatment in the past year (nim).

Internet-based social relationships are a crucial aspect of many people's social lives. As of April 2023, there are 4.80 billion social media users worldwide, which is about 60% of the population (4). Thus, social media platforms, such as Reddit, provide an outlet for individuals to express their thoughts, making it a valuable resource for mental health research.

In this paper, we aim to compare the performance of BERT, RoBERTa, and XLNet-based models that utilize posts from several mental health-related subreddits to detect the presence of specific mental illnesses, including Bipolar Disorder, Anxiety, Depression, Suicidal ideation, ADHD, BPD, Psychosis, Schizophrenia, and Autism. The goal of this project is to explore the potential of social media as a tool for detecting mental illness and facilitating early intervention and treatment.

2 What you proposed vs. what you accomplished

In this section, we give a high-level overview of what we initially proposed to do and what we finally accomplished.

In summary, we were able to complete all the tasks we proposed and also accomplished some additional milestones. The tasks we accomplished are crossed out and the new additions have been marked with an asterisk.

* We designed and built a dataset that contains more classes than the paper we are reproducing. Instead of classifying over levels of depression (severe, moderate, none), we attempt to classify different mental illnesses, to ensure the right kind of treatment.

* In addition to the original base versions of BERT, RoBERTa, and XLNet, we also trained the large model versions of these, namely BERT$_{large}$, RoBERTa$_{large}$, and XLNet$_{large}$.

* Since feedback on the proposal was very positive, we incorporated the graders' suggestions to evaluate results for majority voting and weighted averaging ensembles.

- ~~Collect a large corpus of Reddit posts and preprocess the dataset~~
- ~~Build and setup all the models which we will use in our finetuning task~~
- ~~Perform extensive hyperparameter tuning experiments to achieve the best predictive ability from the models.~~
- ~~Test multiple ensembling strategies to improve performance on the test set~~
- ~~Evaluate model performance by using multiple metrics and comparing results with the paper we are reproducing~~
- ~~Perform in-depth error analysis to figure out what kinds of examples our approach struggles with~~

3 Related work

While research in psychiatry, medicine, and linguistics provides us with resources to support individuals suffering from mental illnesses, it has become increasingly important to detect these early

on. Early research demonstrated that linguistic analysis of speech can be used to categorize people who have depression and paranoia (11). In a survey of undergraduate students, Christofides et al. (3) found that most participants disclosed more information about themselves on Facebook than in real life proving that social media has become the primary source of outlet for many individuals. One of the first studies that established the use of computers to predict mental health issues was conducted in 2004 by Rude et al. (14). The study revealed that computerized text analysis based on cognitive bias can give predictive indications of neurotic tendencies and mental health issues. The fact that people with mental disorders use social media as an outlet along with the significant research that suggests the use of computerized linguistic analysis for detection paved the way for more research in the area.

In recent years, a lot of research has been conducted on the use of NLP for detecting the presence of mental illness from user-generated content on social media platforms. Yadav (16) suggest that detecting the presence of mental illness can be based on the tone of the post, i.e. if the text has a lot of negative connotations, the user is considered a candidate for "early depression". The approach uses various machine learning classifiers such as Naive Bayes, Random Forest, etc to perform detection. Extracting psycho-linguistic features which capture psychological properties of the posts, such as sentiment, affect, and emotional valence is also a popular way to detect mental illness per (18). A potential drawback of this approach is that it might require significant manual effort to hand-engineer these features. Our approach, on the contrary, needs no manual intervention when it comes to feature extraction. To build on this, Gamon et al. (6) broadens the scope of social media-based mental health by introducing a myriad of measures including engagement, egocentric social graph, linguistic style, etc that can be used to predict depressive disorders.

Kim et al. (8) use convolutional neural networks (CNNs) to extract features from user-generated content and classify them as either indicative of a mental illness or not. Although the model shows promising accuracy, as with Yadav (16), their data collection was lacking, in that, they had a class imbalance problem. Our approach ensures that each of the 10 classes have an equal presence in the dataset to avoid the problem of class imbalance.

Li et al. (9) also incorporates multi-modal features which not only include text but also images, and metadata. The attention mechanism is used to highlight the most informative parts of the input. While this approach solves the previous issue of class imbalance, the hierarchical architecture of the model may lead to overfitting on specific datasets and reduce its transferability to other datasets or domains. Another interesting study found that those who are depressed are more likely to share Instagram photographs that are bluer, darker, and greyer. They also shared far fewer photos with people in them (13). All these factors can be beneficial when trying to determine if someone has a mental illness based on their social media activity.

The work that we present in this report closely resembles the research done by Poświata and Perełkiewicz (12). Poświata and Perełkiewicz (12) use transformer-based language models, namely, BERT, RoBERTa, and XLNet to detect the level of depression (severe, moderate, or none) from social media text. They then average predictions over 2 models in order to obtain final results. In our work, we collect and build our own dataset by scraping posts from multiple subreddits associated with different mental health issues. We then build upon their research by constructing a system that detects nine different mental illnesses. Further, to find the best result we employ both weighted averaging and majority voting ensembles.

4 Dataset

The initial goal of this project was to reproduce the results in Poświata and Perełkiewicz (12). The dataset used in our reference paper contains 3 classes namely, moderate, severe, and no depression. We aimed to further improve this work and perform a classification task to identify specific mental illnesses, as this provides for targeted treatment of the individual. There are currently only 2 datasets online that moderately align with our goal but are not publicly available for university-level projects and are only available for formal research purposes.

Hence, to achieve our goal, we build a dataset using textual data from posts made in several mental health-related subreddits, including r/adhd, r/anxiety, r/autism, r/bipolarreddit, r/bpd, r/depression, r/schizophrenia, r/psychosis, and

r/suicidewatch. These 9 classes make up approximately 85% of the data and were collected by leveraging Reddit's publicly available Pushshift API (2). The remaining 15% of the data acts as the control group and was pulled from subreddits related to happy conversations, family, or friends. This control group is labeled as 'no mental illness' to separate it from the other 9 classes, each related to a specific illness.

Our data collection follows an approach similar to Kim et al. (8), meaning our hypothesis is that a user who has a specific mental health problem posts on the corresponding subreddit that deals with the problem. Similar to Kim et al. (8) we plan on annotating the text by using the subject of the subreddits they were pulled from as the label.

We believe collecting data from multiple mental health-related subreddits is a valid approach as there is no publicly available dataset that collates data for all nine mental health illnesses we are concerned about. Most of the existing datasets like InfamousCoder (7) are designed for use with binary models and label data with just "Depressed" or "Non-depressed". By collecting data from multiple subreddits, we aim to create a diverse dataset that includes a range of mental health issues, making it more representative of the population and also ensuring that our model is more robust and is able to capture the nuances of each mental health issue. The subreddits we have selected are all dedicated to specific mental health issues, making them a reliable source of text data for our classification model.

Reddit's Pushshift API is publicly accessible and provides an easy and efficient way to collect large amounts of data from subreddits. It allows us to collect posts from multiple subreddits at once and filter the data based on various parameters.

Overall, we believe that the text data we used is appropriate for our research purpose. We successfully compiled a relatively large corpus of 9720 Reddit posts.

Post our pre-processing steps, we were able to compile a relatively balanced dataset with each of the illness classes having approximately 775 data points and the control group with 1600 data points. The pre-processed dataset has a total of 8665 data points, with an average post length of around 149 words with the longest post being 2084 words. Further analysis revealed that most posts were within 350 words with a few being beyond the limit of 512 tokens for tokenizers such as the BERT and RoBERTa Tokenizer.

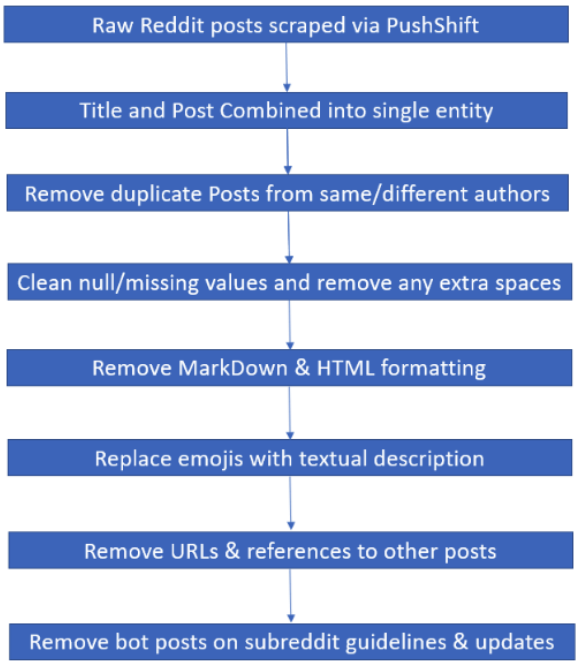

Figure 1: Data preprocessing

We used an 8:1:1 ratio for splitting our raw dataset into training, test, and validation sets. This results in the majority(6900 posts) of the dataset being used for training, with the remaining posts equally split between the testing and validation sets. The figure shows the distribution of data points across the different classes before and after the preprocessing step, thereby confirming that the dataset is balanced with no under-represented class.

4.1 Data preprocessing

Since we are using a dataset that has been curated from Reddit, it requires a fair bit of preprocessing and data cleaning before it can be used to train our model(s). Hence we subjected our dataset to the following preprocessing strategies:

Remove deleted posts

We employed the open-source PushShift API to pull Reddit posts from various subreddits we identified, where each subreddit corresponds to one of the classes we are going to classify. Since we are scraping a large number of posts from each subreddit, it is within the realm of possibility that we might encounter a deleted post or a post written by an author whose account has been deleted. These

Sample Text	Class	Datapoints Count
I've been depressed for years honestly it just seems like my normal mental state and being happy is something that just comes and goes whenever. I really don't feel like doing anything anymore, getting out of bed is a struggle.	Depression	679
I believe I have ADHD. I've been feeling this way for years but finally trying to do something about it. I recently lost my job due to me continuously zoning out in training and not being able to retain simple information. I am really disappointed in myself that I let it go this far.	ADHD	874
Why? Why do i only feel constant fear and regret ?	Anxiety	846
Have you ever recovered from a breakdown in your life? I'm really suffering academically, my social life has been ruined, and I'm on the verge of losing my job. I can't find the words to express the sadness of missing so many milestones and finally discovering that you're autistic, aged 21. I don't know what to do next.	Autism	909
I'm always contradicting myself I'm so self-conscious about this. Someone will ask my opinion one day and I'll give it honestly. Then we'll be talking about the same subject a few months later and my honest opinion will have completely changed. It feels gross, but the truth is that what is right for me in one part of my cycle is completely wrong in a different part. Anyone else relate?	Bipolar Disorder	857
Is this common with bpd? So there's a thing I struggle with since a young age and I've never been able to identify what it is or where it comes from. When I look in the mirror too long I start to feel what I describe as weird. It seems like I'm not myself anymore. The clothes I wear seem to fit strange on my body and I start to feel sort of unwell. I'm feeling a uncomfortable warm sensation and start to feel disgusted of myself, which makes me physically unwell. Does this sound familiar to anyone?	BPD	869
I wanna die My bday is coming up in 2 days and I hate my life. I wanna die. I hope I never have to be in this life again. This is torture. I want death now!	Suicidal	656
I'm not diagnosed but I'm questioning myself. I feel like I'm living in the matrix I've stopped hanging around people. I have been looked down on for not being as educated as them and displaying signs of mental illness whether they were aware of it or not. For those aware of it still mocked me. I was abused in childhood so I shut down and have been living in my own world so I guess I didn't grow like everyone else did. We have all been programed and brainwashed and now I see it that way I feel like I'm going insane.	Schizophrenia	657
im trapped pls dont be mean to me, pls. i have no access to medication or a psychiatrist. i cannot call these delusions anymore, they are real. i need someone to talk to, someone to trll this to but im so scared everyone is a spy and if thatd the case, i know it. i know that everyone is planning to hurt me. and im being forced to be here. but they are giving me all this pain so i can kms and go to them, so i can wake up in the real world that i am meany to be in. pls heşp me. i cant hear them or see them anymore but i feel them	Psychosis	743
Today I am a double dad My baby daughter was born this morning 6 lbs 13 oz and I could not be happier! She joins my 2.5 year old son who already adores her. I love this community so much and I'm happy to cement my sense of belonging with Daddit.	No mental illness	1575

Table 1: Sample Data

posts generally contain an empty body and hence are deemed unusable and were eliminated from our dataset prior to training.

Remove duplicate posts

It is a well-known fact that sharing of posts by multiple users apart from the original author is quite common on social media. While this is unlikely on mental health subreddits, it is still a possibility that must be considered. We also used the contents of the title as part of each data point, hence in order to successfully accomplish the deduplication process we do so based on both the title and contents of the posts which we have gathered.

Combine the title and post into a single textual entity

In order to retain as much information about a potential mental health disorder, it is important to include as much information which points to the class under consideration as possible. Therefore, we chose to combine the title and the post into a single entity to avoid any information lose.

Remove HTML and Reddit markdown formatting

Reddit posts are written using markdown syntax. The PushShift API (2) gives us the raw data including some HTML and markdown text formatting. Since we want a version free of any formatting characters we used the redditcleaner module in Python to remove the markdown formatting such as bullet points, strike-throughs, tables, etc.

Remove URLs, references to other posts, and extra spaces

As is common with any social media post, it is likely that the post might contain a link to another webpage. Since this URL is of no use and would possibly hamper data quality, we get rid of it as and when present. Additionally, a lot of Reddit posts reference other posts using the '>' symbol or as a quoted block. Redditcleaner doesn't handle references to other posts and hence this needed to be handled manually during the preprocessing step. Lastly, any leading or trailing spaces are removed.

Remove posts made by bots

For each subreddit, there are often updates relating to the rules and regulations for that subreddit. This is often shared via a Reddit bot. Additionally, there are a lot of automated Reddit accounts that share posts via bots. Since we are pulling data from Reddit without any means to identify and ignore posts made by bots and as we are dealing with mental illness, these posts needed to be removed. We do this by searching for certain keywords in the content returned to us by the PushShift API call that helps to identify posts associated with bots.

Replace emojis with their inferred meaning

Emojis are often used to express how a person is feeling, and hence is an important aspect to consider when evaluating a person's mental health at the time the post was written. We use the demoji module in Python to replace emojis present in the post with their inferred meaning so as to capture as much useful information as possible.

After applying all of the aforementioned preprocessing, we removed about 1055 posts from our dataset. Even though the extensive preprocessing reduces our dataset by approximately 11%, it will remove any outliers and unimportant data that could confuse our model resulting in better predictions.

4.2 Data annotation

There was no explicit manual annotation task associated with this project since each data point is labeled with the subreddit it was scraped from. While this does not ensure that each data point is correctly labeled (in the case of troll posts), it is the most feasible way considering the sheer volume of the dataset.

5 Baselines

Since we are reproducing the results of a paper we do not have a direct baseline to compare to. Based on the piazza post 549, we use one of the models we fine-tuned on as our base model and compare it with our best model. We decided on BERT$_{base}$ as our baseline and we compare it with the 5 other models we fine-tuned as well as the 2 ensembles. We opted to choose the BERT$_{base}$ as our baseline since it is the smallest model we are using and probably the least powerful one compared to RoBERTa, XLNet, and the larger variations. Hence it makes sense to compare the small-

est model with the best-performing model which is outlined in the results section.

Our goal in this project apart from finding the best model performance is to answer the following questions:

- What effect does the model size and the number of layers have on model performance? Do large models consistently perform better than the base model for our classification task?
- What effect does the amount of data and type of data the models were pre-trained on have on the accuracy?
- What effect does the pre-training strategy have on the accuracy?

We used an 8:1:1 train-test-validation split for finetuning our models and the hyperparameters associated with each of them along with the different model characteristics can be found in Table 2. Since these models are widely used today, we didn't find it pertinent to include the logic behind their working in this report but a brief overview is given in the Models and Architecture section.

6 Your approach

Our implementation was developed from scratch and was written without referencing the code developed by the authors of the paper we are reproducing. We did reference the code in HW1 of our course which assisted us in our finetuning tasks. A high level overview of the process can be seen in 2.

6.1 Models and Architecture

In recent years, the dominant trend in text-based classification has been the utilization of deep learning techniques and the adoption of large language models that are based on pre-trained transformers. With our limited data restriction, training a model from scratch would not be able to compete with fine-tuned versions of state-of-the-art models currently available to use today. Hence, we aim to replicate the work of "Poświata and Perełkiewicz (12)" and plan to incorporate state-of-the-art large pre-trained language models like BERT (5), RoBERTa (10), and XLNet (17) and subsequently fine-tune them on our dataset.

BERT is a bi-directional transformer for pre-training over a lot of unlabeled textual data to learn a language representation that can be used to fine-tune specific machine-learning tasks. The improvement in the performance of BERT in comparison to state-of-the-art techniques at the time could be attributed to the bidirectional transformer, novel pre-training tasks of the Masked Language Model, and Next Structure Prediction. RoBERTa and XLNet are variations of BERT and have already learned contextualized word representations as they are pre-trained on a large corpus of text data (RoBERTa used 1000% more training data compared to BERT) using different training techniques. RoBERTa uses an optimized training algorithm that improves the pre-training objectives by removing the NSP task and replacing BERT's static masking with dynamic masking where the masked token changes during the training epochs. In contrast, XLNet uses a permutation-based approach where tokens are predicted in random order instead of sequentially, thereby allowing the model to capture bidirectional context without relying on the masking mechanism used in BERT and hence handles dependencies and relations between words better.

We also use the large variations of these 3 base models which have double the number of layers with a lot more parameters and hence further improve upon the performance of the base versions. An overview of the model characteristics can be found in 2.

6.2 Tokenization

After preprocessing the data, we tokenize the text into subword tokens using the corresponding language model-specific tokenizer. BERT uses the WordPiece tokenizer, RoBERTa uses a byte-level BPE tokenizer and XLNet uses a SentencePiece tokenizer.

6.3 Fine Tuning

Next, we use the pre-trained large language models as a feature extractor to extract contextualized word embeddings for each sentence in the dataset. BERT can generate representations of words in the context of the sentence, which can help capture the meaning of the words in a better way. The contextualized embeddings can be obtained by running the input sentences through the pre-trained models and extracting the output from the last hidden layer of the model. We then add a classification layer on top of the last hidden layer of the model and fine-tune it on our classification task. The classification layer consists of a single dense layer with

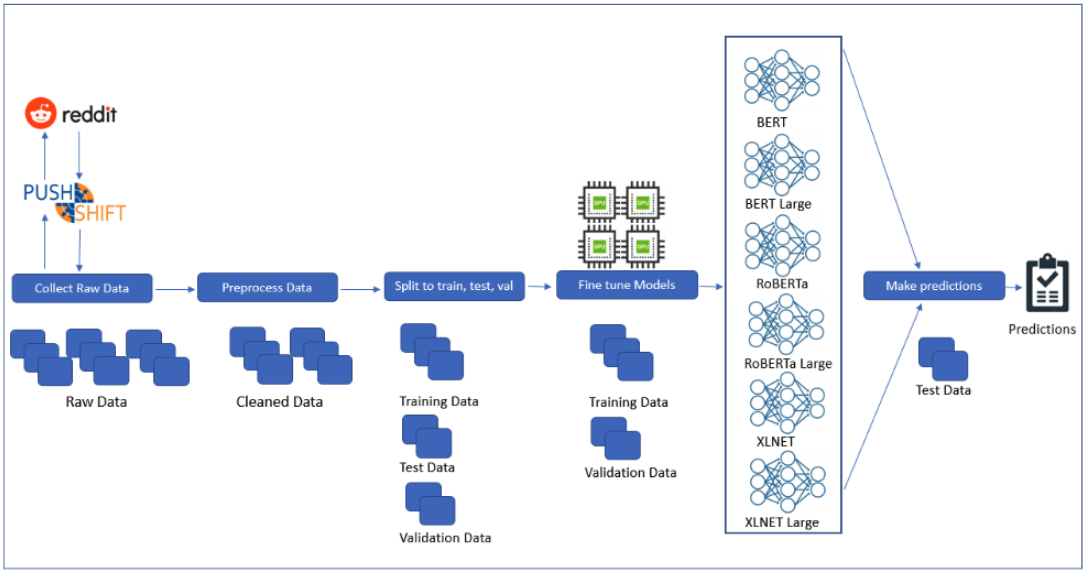

Figure 2: High level approach

a softmax function that outputs a probability score between 0 and 1 indicating the likelihood of a sentence being a particular class. During the fine-tuning process, we use a supervised learning approach to minimize the cross-entropy loss between the predicted and actual labels. The loss curves for our baseline and best models can be seen in 3 and 4.

6.4 Ensemble

After we fine-tuned our 6 models on the training set we evaluate them on the test set and store the predictions made by each model. Next, we try to improve the predictions made by individual models by ensembling models using two different strategies explained below.

6.4.1 Majority Voting

In this technique, we implement *hard majority voting* ensemble. As opposed to soft voting where we sum the probabilities for each class label, in hard majority voting we sum up the votes of each predicted class label made by each model and the final prediction is the class with the highest votes. Majority voting, in specific hard majority voting, suits our use case well since all the models we use have relatively good performance and mostly already agree in general. Additionally, the performance of this ensemble surpasses that of any other model in the ensemble, while also exhibiting lower variance than its counterparts.

$$\text{Majority Vote} = \max(|\text{class 1}|, |\text{class 2}|, |\text{class n}|)$$

6.4.2 Weighted Averaging

This ensemble technique is slightly different from the majority voting method. For each test data point, each model makes a prediction where each class is assigned a probability of it being the correct label. The class with the highest probability from the final softmax layer is considered the predicted label.

One way to average the results is to use a total averaging method where the probabilities for each class are averaged out across all the models in consideration. A caveat here is that all models have equal weightage and contribute equally to the final prediction.

A better solution would be to use an averaging technique where the contribution of each model is weighted proportionally to its prediction capability. The weight in question that we use is the accuracy that each model achieves on the validation set.

Now instead of directly taking the average probabilities for each class we first multiply the class probabilities of a model with its validation accuracy, following which we evaluate the average probabilities of each class across the models. The final prediction will be the class that has the highest weighted average probability.

$$\text{Weighted Avg} = \max\Big(\text{Avg}\big(\text{Eval Weight} * \text{Model Probability for model i for class j}\big)\Big)$$

6.5 Framework, Tools, and Libraries

During the dataset collection phase, we used PushShift to scrape the data and langdetect library to ensure that we only scraped English posts. We used redditcleaner, demoji, and re for our preprocessing tasks which was also heavily dependent on the pandas library. All 6 models along with their corresponding tokenizers were downloaded from HuggingFace. We used the PyTorch Transformer library in conjunction with other PyTorch libraries to complete our classification task. Finally, we used matplotlib and pyplot for visualization and graphical analysis.

6.6 Infrastructure

We were initially constrained by computing restrictions, so we fine-tuned our model on a single Tesla A100 GPU on Google Colab. All 6 models used were from HuggingFace and corresponding loss and optimizer functions have been used from PyTorch. The models were trained on Google Colab Notebooks which gives Jupyter notebooks virtual machines with GPI access. The issue is that Colab gives access to the machine for a limited timespan and limited memory and compute power. The best model achieved based on accuracy on the validation accuracy during training was saved and these checkpoints were used on the test set. Since we needed to fine-tune models with data points that are relatively large, we attempted to split our fine-tuning task across multiple Google accounts, but in the end, we purchased a Colab Pro subscription for increased reliability, since model sizes overwhelmed the limited RAM.

To ensure that sessions didn't time out during training, we used an interesting hack where we used a Chrome extension that randomly moves the mouse at regular intervals.

6.7 Hyperparameters

We performed extensive hyperparameter tuning for all the models. We varied the batch size between the values of 8 to 32 and found that for all models a common batch size of 16 worked best. For optimizer, based on several studies online, we decided to work with AdamW. We found that AdamW with an epsilon value of 1e-8 worked best across the board. After varying the training epochs a few times, we fixed the number of epochs each model was fine-tuned to 10. The learning rate was the one hyperparameter that varied between models. We see that BERT and BERT$_{large}$ worked best with a learning rate of 5e-5 while XLNet and XLNet$_{large}$ works best with 3e-5. RoBERTa and RoBERTa$_{large}$ performed best with a learning rate of 2e-5 and RoBERTa$_{large}$ was the best performing model of the lot. Surprisingly we saw that all base and large versions performed best with the same learning rate.

As mentioned earlier we checkpoint the best model and persist to disk and load it during evaluation on the test set.

6.8 Evaluation Metrics

We aimed to utilize metrics such as accuracy, macro-averaged precision, macro-averaged recall, and macro-averaged F1-score across all classes. The primary evaluation measure for solutions is the macro-averaged F1-score.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1-Score} = \frac{2 \times (Precision \times Recall)}{Precision + Recall}$$

7 Results

Our results are presented in 4. The original paper was related to the classification of the degree of depression(3 classes - severe, moderate, none) while our models were tuned to classify mental illnesses into one of 10 classes. Conceptually our task is more difficult than the one in our reference paper since a lot of our classes have overlapping and similar features. Even though the 2 tasks were different in nature there were certain parallels we saw in relation to model performance. BERT$_{base}$

Table 2: Hyperparameters and Model Characteristics

Model	Learning Rate	Tokenizer	Pretraining Strategy
BERT$_{base}$ and BERT$_{large}$	5e-5	WordPiece	NSP, MLM
RoBERTa$_{base}$ and RoBERTa$_{large}$	2e-5	Byte-level BPE	Dynamic Masking, No NSP
XLNet$_{large}$ and XLNet$_{large}$	3e-5	SentencePiece	Permutation Modelling

Table 3: Other Hyperparameters

Parameter	Value
Optimizer	AdamW
Batch Size	16
Epochs	10
Max Sequence Length	512

and XLNet$_{base}$ performed better than their corresponding large versions while RoBERTa$_{large}$ performed better than the base version.

Overall RoBERTa$_{large}$ had the highest accuracy(0.7359), Recall(**0.73587**) and F1 Score(**0.73281**). For Precision, similar to the reference paper, we found XLNet$_{large}$ to be the best(**0.73785**) with RoBERTa$_{large}$ a close second(**0.73768**).

Both ensembles outperformed all models when it came to all 4 evaluation metrics. We found a significant increase in model performance where the majority voting ensembles reported an accuracy of **0.74509** and an F1 Score of **0.74239** while the weighted average ensemble reported **0.75432** and **0.75096** for the same metrics. Thus overall, the weighted average ensemble was the best-performing model, achieving a **2.5%** increase in accuracy over RoBERTa$_{large}$. The other metrics from the ensemble are reported in the results table.

These results helped us answer a lot of questions about our hypothesis. At a high level, we were able to conclude the following about our models from our experiments:

- In general, we see that larger models(for RoBERTa and XLNet) tend to perform better than base models since they have more parameters and capability to learn more features.

- Amount of data that the models were pretrained on has a sizeable effect on model prediction capability. We found that RoBERTa and XLNet consistently outperformed BERT, and both these models were trained on a much larger dataset compared to BERT.

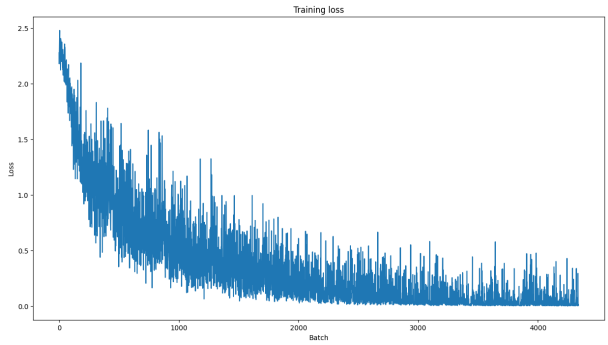

Figure 3: BERT$_{base}$ Training Loss Progression

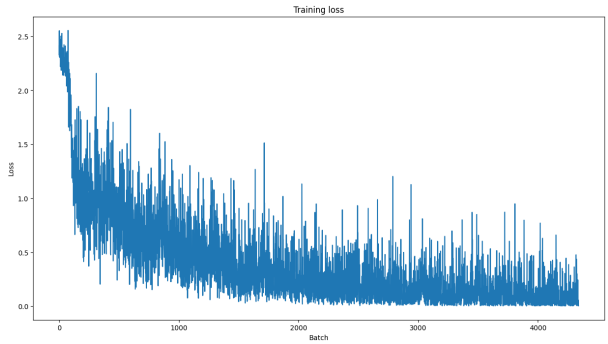

Figure 4: RoBERTa$_{large}$ Training Loss Progression

- RoBERTa's dynamic masking and XLNet's permutation modeling approach seem to be better than BERT's NSP and MLM pretraining strategies.

- Model ensembling in general seems to be a good strategy to improve model predictions and weighting the model contributions by their prediction ability seems to be a better approach than the equal weightage approach of majority voting.

Table 4: Model Evaluation Metrics

Model	Accuracy	Precision	Recall	F1 Score	Top 2 Accuracy
BERT$_{base}$	0.695	0.719	0.695	0.693	0.838
BERT$_{large}$	0.687	0.689	0.687	0.685	0.845
RoBERTa$_{base}$	0.696	0.712	0.696	0.694	0.845
RoBERTa$_{large}$	0.735	0.737	0.735	0.732	0.855
XLNet$_{base}$	0.689	0.703	0.689	0.689	0.835
XLNet$_{large}$	0.730	0.737	0.730	0.727	0.848
Majority Voting Ensemble	0.747	0.751	0.747	0.744	0.797
Weighted Average Ensemble	**0.754**	**0.763**	**0.754**	**0.751**	**0.875**

8 Error analysis

8.1 Common errors across models

We conducted an extensive error analysis to better understand the variability in model predictions across all 8 candidate models and attempted to find commonalities between them. Firstly a common characteristic that was seen across predictions made by all models was that the 2 classes with the highest misclassification rate were always 'depression' and 'suicidal ideation'. Conceptually, this makes sense as several studies show that there is a strong link between depression and suicide. In fact, the lifetime risk of suicide among people with untreated depression is nearly 20%. Additionally, after analyzing the wrongly classified texts for both these classes, we noticed that a common phrase seen for posts related to depression almost always included some reference to the person wanting to end their own life. Consequently, when a similar analysis was done for wrongly classified texts whose true labels indicated suicidal nature, we noticed that in addition to phrases that pointed to the person wanting to end their life, there was more often than not, a mention of depression being one of the reasons behind how they were feeling. Both these scenarios naturally confuse the model as it cannot decipher which of the 2 classes the text belongs to.

As was the case with 'depression' and 'suicidal nature', we found very similar model behavior when it came to classifying 2 more pairs of classes that were consistently being confused for each other. The classes are psychosis and schizophrenia. With some online research, we were able to conclude that psychosis is in fact one of the many symptoms of schizophrenia. So it was natural that a lot of posts from the schizophrenia subreddit made mentions of 'psychosis' and 'losing touch with reality'.

Similarly, with 'bpd' and 'bipolar' classes, it was found that bpd was most often misclassified as being bipolar. One reason behind this could be the fact that both these classes share a common theme of being impulsive and moody and often suffering from some kind of depression(which explains why the second most misclassified class for these two classes was depression).

Lastly, anxiety was found to be the class with the lowest error rate and hence had the best classification rate. With these observations, a few major pitfalls of the dataset comes to light. Firstly, classifying the posts based on subreddits is subject to high variability since it is common for people with similar symptoms to self-diagnose and post on different subreddits. This leads to the model learning features that are common to different classes leading to misclassification at the time of evaluation. Secondly, a lot of the classes are not mutually exclusive. As highlighted earlier many illnesses are results of one another or share common symptoms with each other. Since there is no commercially available dataset for our classification task, it is difficult to remedy this situation. Lastly, a lot of the posts mention another class in their text. This again causes the model to get confused when attempting to classify the data points of the test set.

8.2 Top k classification accuracy

The next step of our error analysis was to check how well our model did if we overlooked the overlapping nature of classes. In our opinion, the best way to do this was to find the top k classification for each model where k=2. With this, if our model truly learned and focused on features that were synonymous with each of the classes then at least one of the 2 classes with the highest probability should match with the ground truth. As can be

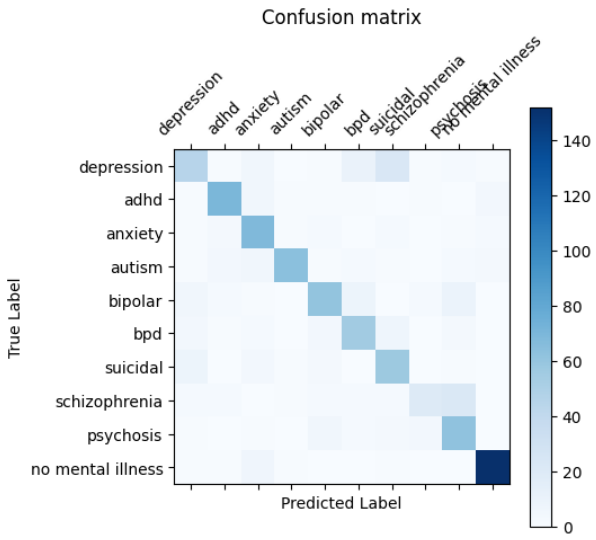

Figure 5: Confusion Matrix for Weighted Avg Ensemble

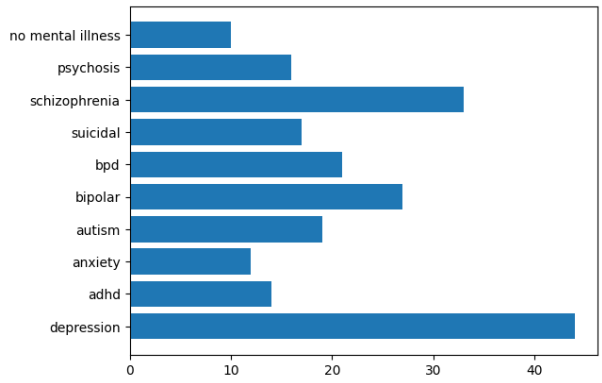

Figure 6: Incorrect classification counts for weighted average model

seen from 4, our accuracy improved substantially with an average increase of around **12.5%** when evaluating the top 2 accuracy for each model, with the weighted average ensemble achieving the highest accuracy of **0.8742**. This shows that our models were very close to correctly classifying a large section of the misclassified texts.

8.3 Baseline vs Best Model

Lastly, we wanted to compare our baseline and the best model for different properties both for semantic commonalities and differences. As mentioned in the baselines section we are using the BERT$_{base}$ as our baseline and we compared it with our best model which was the weighted average ensemble. Since our ensemble model bases its predictions on the predictions made by the other 6 models (one of which is our baseline) it is highly likely that it shares common properties with BERT.

We started by first finding a set of 100 examples sampled from the test set. We constructed it to include 35% of examples wrongly classified by BERT, 35% wrongly classified by the ensemble, and 30% randomly sampled examples that might or might not be correctly classified by each model. We then manually annotated them for properties that we found to be the reason behind the misclassification. The distribution of the properties for the models can be found in 7. We manually annotated the texts for 4 classes which are highlighted in the graph. We see that the most common properties for these difficult examples are that they contain features indicative of a different class or that they don't contain enough features of the true class label. Additionally, we also see that the remaining 2 properties, namely that the text contains other class names or multiple class names also contribute to misclassifications to a sizeable extent that can't be ignored. Again this is indicative of a dataset where the relationship between the text and the true label is not very robust.

9 Contributions of group members

- Chirag Uday Kamath: Data collection and pre-processing, fine-tuning BERT$_{large}$. Built Majority voting and Weighted Avg Ensembles, error analysis, and annotations. Also worked on report writing.
- Divya Maiya: Data collection and pre-processing, RoBERTa$_{base}$ and large fine-tuning, hyperparameter tuning, and error analysis. Also worked on report writing.
- Neha Prakash: Data collection, fine-tuning BERT$_{base}$ and XLNet$_{large}$, model evaluation metrics, error analysis, and annotations. Also worked on report writing.
- Akhil Arza: Fine-tuning XLNet$_{base}$, data annotation.

10 Conclusion and Future Works

We finetuned the base and large versions of 3 of the leading transformer-based large language models available today to help classify different mental health illnesses by using data scraped from Reddit posts. We found RoBERTa$_{large}$ to be the best-performing model and further improved its

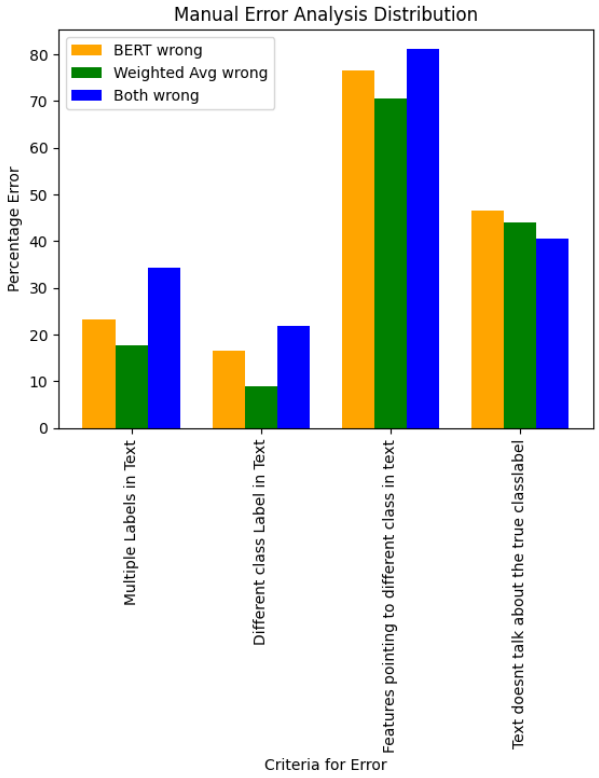

Figure 7: Features of manually annotated datapoints

performance by using 2 different ensemble techniques, namely weighted average and majority voting. We concluded from our experiments that the large language model with more layers and parameters, which was trained on more amount of data usually tends to outperform smaller models. We also found that having classes with overlapping and similar features tends to confuse the model, leading to misclassification.

One major pitfall of our approach was in the dataset collection. In the future, we hope to finetune our models on a more robust dataset that has examples with content that does not point to multiple classes. We hope to collect a much larger corpus of text to finetune the model thereby magnifying its prediction capabilities.

Overall, we found the entire process to be highly informative and fruitful. We learned to work with different models apart from BERT which was covered in our course and were also exposed to the different pre-training and tokenization methodologies that were used by the other models. One major learning was the importance and nuances involved in error analysis. The error analysis phase was probably the most time-consuming apart from the model building, but this allowed us to explore our data in-depth and also draw conclusions which in turn highlighted the weaknesses in our model which can be worked on in the future.

In conclusion, we firmly believe that our work, leveraging the power of social media and utilizing a diverse dataset from mental health-related subreddits, will significantly contribute to the development of more effective methods for detecting and identifying mental health illnesses.

Our code was mailed to the instructors account as a zip file. To access the code with model checkpoints please use the following drive link : click here

11 AI Disclosure

- Did you use any AI assistance to complete this proposal? If so, please also specify what AI you used.
  - Yes, ChatGPT

If you answered yes to the above question, please complete the following as well:

- If you used a large language model to assist you, please paste *all* of the prompts that you used below. Add a separate bullet for each prompt, and specify which part of the proposal is associated with which prompt. Used ChatGPT to help generate the introduction for the proposal which we partly reused. (These are the prompts we used when we created the proposal)

  Prompts:

  - Write a short introduction for a paper titled - "Detecting mental illness from user content in Reddit". The paper uses posts pulled from several mental health-related subreddits to classify them as one of the following: Borderline personality disorder, Anxiety, Depression, Suicidal, ADHD, and Autism.
  - Add some statistics

  Used ChatGPT to help with the Data section for the proposal which we partly reused. (These are the prompts we used when we created the proposal)

  - I'll collect 60% of the data from Reddit using Reddit's Pushshift API for the following subreddits: r/adhd,

r/anxiety, r/autism, r/bipolarreddit, r/bpd, r/depression, r/healthanxiety, r/socialanxiety, and r/suicidewatch I will collect the remaining 40% of the control data from subreddits related to family or friends

Now answer the following questions:

What text data do you plan to use in your project? Where will you get it from? Will you be annotating text yourselves? Convince us that it is available for you, and that you can easily get it, and that it is appropriate for the task and research questions you care about.

Used the following prompts for the Related Works section for the proposal which we partly reused. (These are the prompts we used when we created the proposal)

- Give me an introduction to the "Related Works" section of a report.
- Summarize this in 5 sentences - The Best of Both Worlds: Combining Engineered Features with Transformers for Improved Mental Health Prediction from Reddit Posts with NLP/ML model used and provide 2 pros and 2 cons
- Explain the features extracted in the above paper
- Explain Psycholinguistic features from above more
- summarize Detection of Depression-Related Posts in Reddit Social Media Forum
- provide a pro and con for above
- Summarize - A deep learning model for detecting mental illness from user content on social media - include NLP techniques used. Also tell the pros and cons.
- Summarize - MHA: a multimodal hierarchical attention model for depression detection in social media - include NLP techniques used. Also tell the pros and cons.

Used ChatGPT to generate BibTex references. Prompts looked like this: Generate a reference for Reddit's PushShift API

- **Free response:** For each section or paragraph for which you used assistance, describe your overall experience with the AI. How helpful was it? Did it just directly give you a good output, or did you have to edit it? Was its output ever obviously wrong or irrelevant? Did you use it to generate new text, check your own ideas, or rewrite text? Based on our proposal experience:
  - For the introduction section, chat GPT produced a very coherent and good output, which needed minimal changes. For the data section, the output was not comprehensive so we made extra edits. Using ChatGPT to generate BibTex reference was immensely helpful and time-saving.
  - For the related works, some summaries generated by ChatGPT was irrelevant and I had to do it on my own. For others, it generated good summaries.

References

[nim] National institute of mental health. https://www.nimh.nih.gov/health/statistics/mental-illness. Accessed on Month Day, Year.

[2] Baumgartner, J. (2016). Pushshift.io reddit api. https://github.com/pushshift/api. Accessed: 2023-03-08.

[3] Christofides, E., Muise, A., and Desmarais, S. (2009). Information disclosure and control on facebook: Are they two sides of the same coin or two different processes? *CyberPsychology &amp Behavior*, 12(3):341–345.

[4] DataReportal (2021). Social media users.

[5] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[6] Gamon, M., Choudhury, M., Counts, S., and Horvitz, E. (2013). Predicting depression via social media.

[7] InfamousCoder (2021). Depression reddit - cleaned. https://www.kaggle.com/infamouscoder/depression-reddit-cleaned. Accessed: 2023-03-08.

[8] Kim, J., Lee, J., Park, E., and Han, J. (2020). A deep learning model for detecting mental illness from user content on social media. *Scientific Reports*, 10(1).

[9] Li, Z., An, Z., Cheng, W., Zhou, J., Zheng, F., and Hu, B. (2023). MHA: a multimodal hierarchical attention model for depression detection in social media. *Health Information Science and Systems*, 11(1).

[10] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

[11] Oxman, T. E., Rosenberg, S. D., and Tucker, G. J. (1982). The language of paranoia. *The American Journal of Psychiatry*, 139(3):275–282.

[12] Poświata, R. and Perełkiewicz, M. (2022). OPI@LT-EDI-ACL2022: Detecting signs of depression from social media text using RoBERTa pre-trained language models. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 276–282, Dublin, Ireland. Association for Computational Linguistics.

[13] Reece, A. G. and Danforth, C. M. (2017). Instagram photos reveal predictive markers of depression. *EPJ Data Science*, 6(1).

[14] Rude, S. S., Valdez, C. R., Odom, S., and Ebrahimi, A. (2003). *Cognitive Therapy and Research*, 27(4):415–429.

[15] World Health Organization (2019). Mental disorders.

[16] Yadav, S. (2016). Detecting presence of mental illness using nlp sentiment analysis. In *Proceedings of the 6th International Conference on Advances in Computing and Communications*, number 06, pages 220–225.

[17] Yang, Z., Dai, Z., Yang, Y., Carbonell, J. G., Salakhutdinov, R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

[18] Zanwar, S., Wiechmann, D., Qiao, Y., and Kerz, E. (2022). The best of both worlds: Combining engineered features with transformers for improved mental health prediction from Reddit posts. In *Proceedings of The Seventh Workshop on Social Media Mining for Health Applications, Workshop & Shared Task*, pages 197–202, Gyeongju, Republic of Korea. Association for Computational Linguistics.