Project Group # 04

288R Final Report - Predicting Reddit Post Topics for Mental Health Resources

Authors: Nathaniel Do ntd002@ucsd.edu, Jesse Parent jeparent@ucsd.edu

Abstract

Identifying and categorizing mental health discussions on popular internet forums like Reddit can help improve accessibility to support resources and contribute to automated mental health assistance. This study applies machine learning techniques to classify Reddit posts into key mental health topics, facilitating potential future resource recommendation systems. We scraped and preprocessed a dataset of 12,951 Reddit posts, implementing two models: Multinomial Naïve Bayes (MultiNB) for classification and Latent Dirichlet Allocation (LDA) for topic modeling. Our results demonstrated that MultiNB, using both title and body text, achieved the highest classification accuracy, while LDA was ineffective for predefined topic classification. We also explored hyperparameter tuning and preprocessing techniques, which improved MultiNB's performance by 14.1%. Despite promising results, limitations such as data bias, ethical concerns around misclassification should be considered. Future work should explore transformer-based models (e.g., BERT) and integrate classification outputs into practical mental health support applications.

Introduction

The internet is a major source of mental health support and information, with forums like Reddit hosting large communities for individuals seeking advice and solidarity. Navigating vast amounts of text-based discussions can be overwhelming, making automated classification of mental health topics a valuable tool for improving accessibility to relevant resources. Our project aims to develop a machine learning-based system that categorizes Reddit posts into six mental health topics: anger, anxiety, bipolar disorder, depression, eating disorders, and panic.

To build our dataset, we scraped 12,951 Reddit posts from mental health-related subreddits using the Async PRAW API. Each post was categorized based on its originating subreddit, assuming that posts from r/Anxiety are related to anxiety, and so on. After preprocessing, which included removing stop words, lemmatization, and normalizing text, we obtained 9,078 cleaned posts, which were then split into 80% training and 20% test data.

We applied two machine learning techniques: Multinomial Naïve Bayes (MultiNB) for classification and Latent Dirichlet Allocation (LDA) for topic discovery. MultiNB, a probabilistic model designed for text data, was trained to predict the most likely topic of a post based on word frequencies. In contrast, LDA was used not for classification but to uncover latent themes within posts, helping us assess whether natural groupings align with our predefined categories. During exploratory data analysis, we found that certain topics, such as Panic and Anxiety, had overlapping common words, potentially making classification difficult. We experimented with different text feature selections (title-only, text-only, and title + text) and hyperparameter tuning to optimize performance. Our goal is to evaluate how well machine learning can classify mental health-related text and to explore how future research could improve automated support tools for mental health communities, and potentially aid in resource recommendations.

Related Work

Research in mental health classification and text-based topic modeling has explored multiple approaches. One major avenue involves supervised learning models, such as Support Vector Machines (SVMs), Logistic Regression, and Deep Learning architectures like BERT. SVMs and logistic regression are widely used for text classification tasks, leveraging term frequency-inverse document frequency (TF-IDF) representations (Joachims, 1998). More recently, transformers like BERT (Devlin et al., 2019) have demonstrated superior performance in text classification but require significant computational resources. For topic modeling, LDA (Blei et al., 2003) is a common probabilistic model used to uncover latent themes in text data. However, Non-negative Matrix Factorization (NMF) has been proposed as an alternative due to its ability to generate more interpretable topic distributions (Lee & Seung, 1999). Prior research suggests that LDA can be useful for exploratory analysis but struggles with classification when topic labels are predefined (Boyd-Graber et al., 2017).

Another relevant area of study involves sentiment analysis and mental health applications. Work by Guntuku et al. (2019) used pretrained language models to assess mental health discourse on social media, showing promising results for sentiment-based predictions. However, these methods require large labeled datasets, which are often unavailable or difficult to obtain for niche topics like mental health.

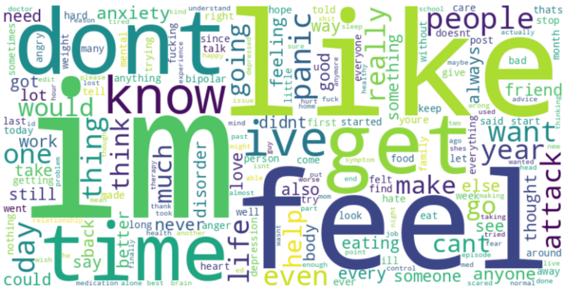
Compared to prior work, our project takes a pragmatic approach by leveraging lighter-weight, interpretable models (MultiNB and LDA) while avoiding computationally expensive deep learning techniques. We aim to assess whether these traditional methods remain viable for mental health text classification on Reddit, given the trade-offs between accuracy, efficiency, and interpretability.

Dataset

We scraped our own dataset from Reddit. By using the Async PRAW Reddit API, we were able to grab posts across a multitude of subreddits. After taking the top rated posts across the topics, our raw dataset ended up being 12,951 rows with 4 columns: Title, Text (post's body text), Score, and Topic. The "Topic" was manually assigned by us according to which subreddit it came from. Score was kept on as a column but never used in the models. Preprocessing the text data came next to normalize it for modeling. We removed things like special characters, stop/common words, and empty spaces. We also lemmatized the words to break them down into base form. After all our processing, we are left with a dataset with 9,078 rows and 7 columns: the former columns and processed versions. Later, we would split it into train sets and test sets with a 0.2 test size meaning about 1,816 in the test set. Validation sets were unnecessary: Multinomial Naive Bayes used GridSearchCV which is optimized on the training set and Latent Dirichlet Allocation studies a dataset instead of predicting with it.

We already had a plan in mind when scraping the dataset to use the text for topic-related predictive tasks. We could have used the text to predict the length of posts under certain topics. Maybe depression posts are short and to the point or bipolar posts have long stories. We also included the score of a post as a column but never ended up using it. We could have predicted if posts score higher when they are longer or with certain topics. Our task of being able to predict a post's topic helps us find a path to recommend mental health resources and it resonates with our team's shared background in cognitive science.

By performing some exploratory data analysis (EDA), we found that the most common words within our text include words like "im" and "like."



Although we had removed stop words, these common words were not found in the stop word library. We took note of them to potentially remove some in the future since they seemed too broad to be related to a topic.

We also listed out the common words for each topic. Many of these words made sense. The anger topic had words like "anger" and "angry." The eating disorder topic had words like "eating" and "weight." Panic and anxiety were interesting; anxiety's most common word was "anxiety," but panic's sixth most common word was also “anxiety.” When two topics have the same common words, the model may have trouble distinguishing between the two topics.

Methods

We ended up using two kinds of models; the first being a Multinomial Naive Bayes model. MultiNB is best used with discrete data for text classification problems. A model is trained with text and a label, then finds which words are frequent with which labels. Then the model is given text to predict the label based on the word frequency. To calculate the probability of the text, it uses this formula of multinomial distribution.

$$P(X) = \frac{n!}{n_1!n_2!\ldots n_m!}p_1^{n_1}p_2^{n_2}\ldots p_m^{n_m}$$

“n” is the total number of trials, “n_i” is the count of occurrences for outcome “i” (count of words for that label), and “p_i” is the probability of outcome “i” (probability of that label). We started with six mental health topics which benefit greatly from this kind of model. By training the model with posts, we had hopes that it could take a new post and predict what topic it would be.

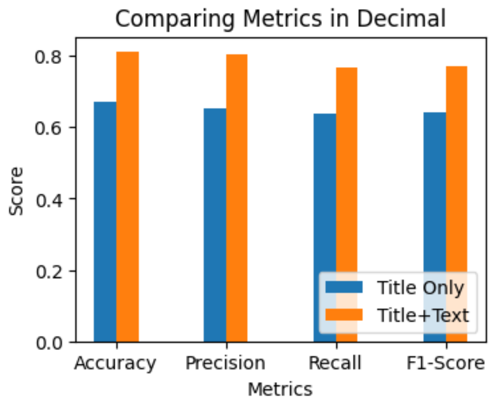
While MultiNB is more grounded with predefined labels, we also took an abstract approach with Latent Dirichlet Allocation. LDA is used for topic modeling. It assumes each document/post is a mixture of topics and maps probability distributions, which then creates its own topics and scores words based on how likely they appear in those topics.

Though our data already had labeled topics, we were curious to see what LDA could do. As mentioned earlier, the topics panic and anxiety had a shared common word, and so we tried LDA to see if it could distinguish between the two topics on its own.

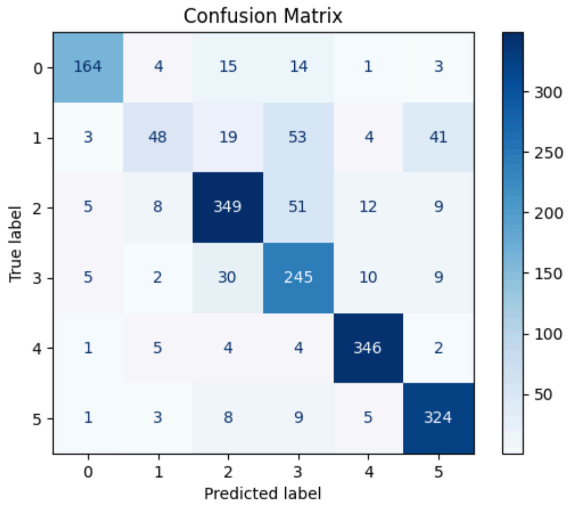
Experiments and Results: MultiNB

We took our preprocessed data and fitted it into our MultiNB model in several different configurations. We wanted to compare the model when we inserted just the Title Only compared to Text Only compared to Title and Text. Four metrics were used. Accuracy is the proportion of predictions thought to be correct over total predictions. Precision is the proportion of the real correct predictions over all the predictions that were thought to be correct. Recall is the proportion of our correct predictions over how many should be correct. F1-Score combines precision and recall, penalizing extreme negative values. All of these metrics range from 0-1 and are considered better the closer they are to 1.

By fitting three different models with the training data and using the test set to find scores, we found that the combined Title and Text ended up being the most accurate; however, the Title Only was higher in the other three metrics. Title Only had the benefit of being more “buzz words” to stand out in the model while Title and Text’s strength was a greater quantity of text for the model to learn. We held onto both of these models and optimized them with hyperparameter tuning. GridSearchCV cross-validates optimum parameters using a grid of parameters we input. This gave us the best alpha (smoothing parameter) and fit prior (will it learn class prior probabilities or not) values. We optimized both the Title Only and Title and Text models and scored them again.



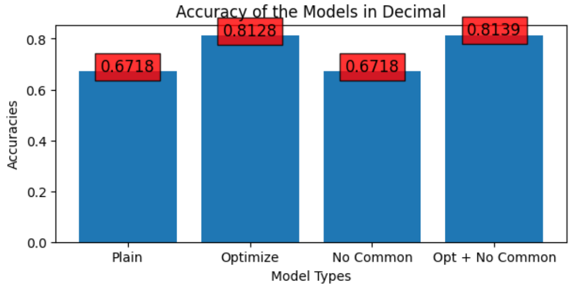
Beating Title Only in the optimized model ring, we decided to observe Title and Text from here on out. We produced a confusion matrix of the winning model to view a summary of predictions.



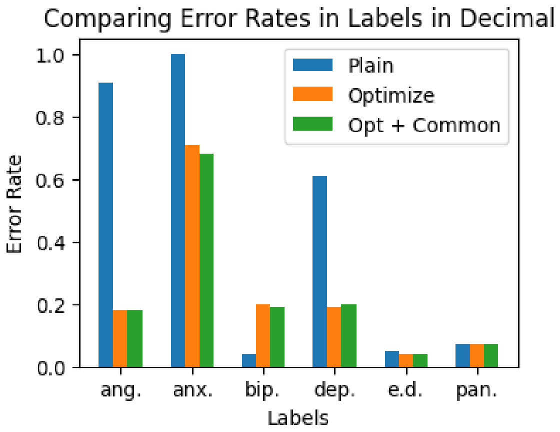
True label \ Predicted label	0	1	2	3	4	5
0	164	4	15	14	1	3
1	3	48	19	53	4	41
2	5	8	349	51	12	9
3	5	2	30	245	10	9
4	1	5	4	4	346	2
5	1	3	8	9	5	324

Labels in order from 0-5 are anger, anxiety, bipolar, depression, eating disorder, and panic. The most interesting point is anxiety which is confused for depression and panic but not the other way around. This may be from the shared common words found during our EDA.

Next we tried removing common words. Our EDA revealed common words that made it past removing stop words; we took some of the most common words and removed them from the dataset before rerunning the model.



As you can see from checking their accuracies, removing the common words from a plain model yielded no accuracy increase. Optimizing the parameters did boost the accuracy by 14.1%. And combining the two gave a slightly greater increase of 14.21% from the plain model. We compared the metrics of the optimized model with and without the common words, and we found that there is hardly any score increase when removing common words. We had expected that removing the common words would allow the model to latch onto the more unique words to make predictions, but as it turns out, it hardly makes a difference.



We also checked to see the error rate of predicting each topic, or how many times the model predicted the wrong label. A plain model struggled predicting anxiety, anger and depression, but after optimizations, anger and depression fell into a similar range as their peers. Strangely, the bipolar rate increased. Interestingly, removing the common words barely decreased anxiety and bipolar error rates while raising depression. For the future, this may mean we shouldn't bother with removing them.

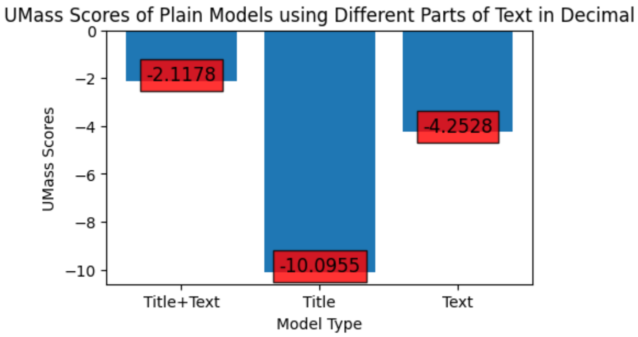
Experiments and Results: LDA

Like the MultiNB model, we also fit the LDA model with the Title Only vs Text Only vs Title and Text accuracy competition. We used UMass Coherence score as our main metric. It calculates how often word pairs appear together in the text.

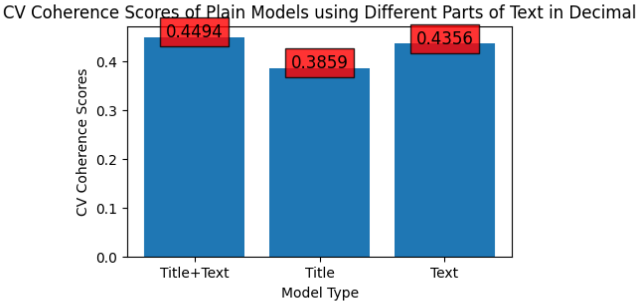
$$C_{UMass}(w_i, w_j) = \log \frac{D(w_i, w_j) + 1}{D(w_i)},$$

"D(w_{i}, w_{j}) indicates how many times words w_{i} and w_{j} appear together in documents, and D(w_{i}) is how many time word w_{i} appeared alone" (Baeldung, 2025). The closer the score is to 0, the more confident the model can separate its topics using word pairs. We also used CV Coherence, a popular metric that "calculates the score using normalized pointwise mutual information (NPMI) and the cosine similarity" (Baeldung, 2025). The higher the CV score is, the better. Although popular, CV's issues on randomly generated word sets cause UMass to be a better metric.

We fit three LDA models with Title and Text, Title Only, and Text Only to see initial standings.

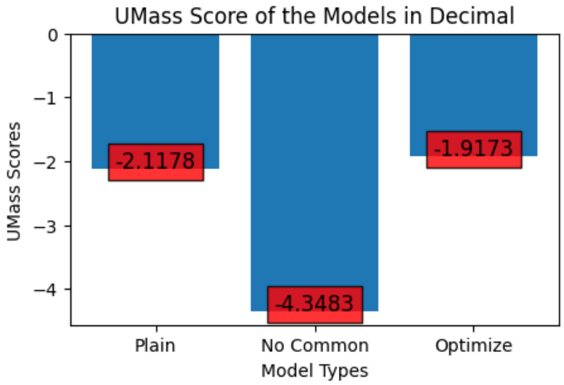


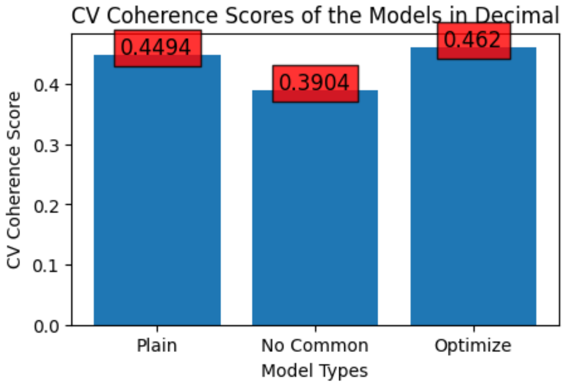
Since -2.1 is closest to 0, the combo is the best again in another model. We also tried to remove the common words just to see, but it performed worse with a score of -4.3.



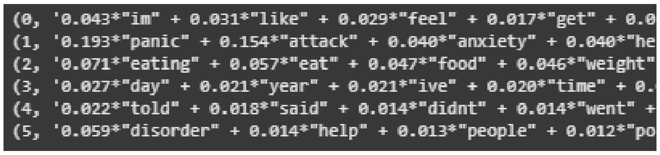
The CV Coherence scores revealed the same as the UMass scores with Title and Text being the clear winner. The same can be said about removing the common words, scoring 0.39 against Title and Text’s 0.45.

We began the process of tuning our hyper parameters. We checked the number of passes through the corpus during training which peaked at 30 in both UMass and CV. Next our chunk size needed to be found which is the number of documents used in each training chunk. 100 was also ideal for both UMass and CV.





Shown here are both UMass and CV Coherence Scores which both decreased with common word removal and increased slightly with optimization.



```
(0, '0.043*"im" + 0.031*"like" + 0.029*"feel" + 0.017*"get" + 0.0
(1, '0.193*"panic" + 0.154*"attack" + 0.040*"anxiety" + 0.040*"he
(2, '0.071*"eating" + 0.057*"eat" + 0.047*"food" + 0.046*"weight"
(3, '0.027*"day" + 0.021*"year" + 0.021*"ive" + 0.020*"time" + 0.
(4, '0.022*"told" + 0.018*"said" + 0.014*"didnt" + 0.014*"went" +
(5, '0.059*"disorder" + 0.014*"help" + 0.013*"people" + 0.012*"po
```

We looked at how the optimized LDA model abstracted our topics, giving us a list for each and telling us the most frequent words. The broad categories it abstracted to were feelings, anxiety/panic, eating, time, speech /movement, and disorders/health. Unfortunately this means we wouldn’t be able to use this model to help predict our 6 desired topics, but enlightening nonetheless to see how LDA abstracts.

Conclusion

Our study demonstrates that machine learning models can effectively classify mental health discussions on Reddit, with Multinomial Naïve Bayes outperforming LDA for predefined topic classification. Through hyperparameter tuning and preprocessing techniques, we improved classification accuracy while also exploring the challenges of distinguishing between closely related topics. These findings suggest that lightweight, interpretable models remain viable for text classification in resource-limited settings. However, our work also highlights limitations: our dataset, while diverse, is constrained by Reddit's user base and moderation policies, potentially biasing our results.

Furthermore, ethical concerns such as the risk of misclassification—where a user's distress signal may be categorized inaccurately—should be considered in future applications. The implications of false positives or false negatives in classifying mental health discussions are significant, as incorrect categorization could lead to users receiving inappropriate or misleading resources. Additionally, all mental health-related applications/products should be designed to account for overuse or overreliance; and even though this specific work does not address interactivity, caution even at sorting and classification stages of an application development is warranted. Moreover, privacy concerns must be considered, particularly when handling user-generated text from social media platforms. Future iterations of this research should incorporate methods that safeguard user anonymity, ensure data protection compliance, and explore the potential harms of algorithmic bias in mental health classification.

Future work should explore more robust NLP models such as transformer-based classifiers (e.g., BERT) and investigate hybrid approaches that balance interpretability with predictive power. Additionally, integrating external mental health resources into a recommendation system could enhance the real-world applicability of this research. Furthermore, interdisciplinary collaboration with mental health professionals could help ensure that automated classification systems provide ethical and meaningful support while minimizing harm.

Appendix

Contributions

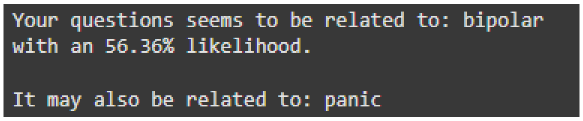
Our personal and professional lives had many changes that it bled into our academic ones. To combat this, we tried to switch off the lead when something came up. Jesse was able to contact Professor Weibel and created the main ideas we would work towards. Nathaniel headed the coding sections and GitHub upkeep (https://github.com/ntd002/DSC-288-Capstone.git). This was reflected in our video presentation and this very final report as Jesse was mainly in charge of discussing our motivations, problems and framing of efforts and Nathaniel covered the coding portions as the "Lead Engineer."

Gap Analysis/Reflection

We had a slightly different route planned in the beginning of this project. We were going to much more directly collaborate with Professor Weibel's team and study dialectic behavioral therapy and explore other ML methods.. They even allowed us to use a dataset from their current project. We had also had a desire to create a tool to help recommend mental health resources.

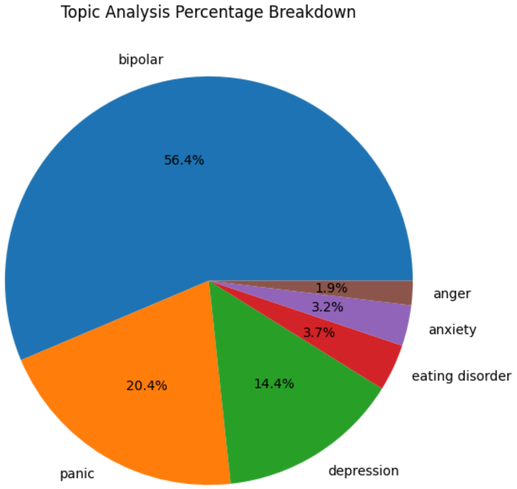
Given that this is a new endeavor by Professor Weibel, the dataset was found wanting in terms of machine learning so we scraped the data ourselves. Secondly, many trials appeared before our team in our personal lives which took our attention away from this project, particularly regarding being able to work more closely with the Weibel Lab.

If we had more time, we would have updated and taken advice from Professor Weibel's team rather than solely focus on our work; further dovetailing within their existing efforts would have been a somewhat more tenable goal.. Using our experience from DSC 209R: Data Visualization, we would've been able to create a website for recommending mental health resources. We created the skeleton of this project recently. It currently has two websites per topic and recommends some depending on how closely a query relates to a topic using the MultiNB model.

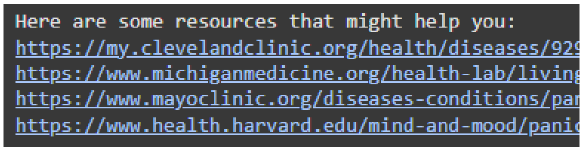


```
Your questions seems to be related to: bipolar
with an 56.36% likelihood.

It may also be related to: panic
```



Topic Analysis Percentage Breakdown



```
Here are some resources that might help you:
https://my.clevelandclinic.org/health/diseases/929
https://www.michiganmedicine.org/health-lab/living
https://www.mayoclinic.org/diseases-conditions/par
https://www.health.harvard.edu/mind-and-mood/panic
```

Redesign Considerations

In one sense, the redesign would have been the original project itself, although given that we did discover or modify our dataset needs along the way, perhaps not an exact match. In its simplest sense, if we chose to make this more a research-exploratory based project, working with the Weibel Lab to discern more about potential datasets and adjust the scope may have been wise. Specifically, the selection of a dataset may be different during a redesign - while it may have been tenable for the classification project we centered on, finding something for additional training before engaging Reddit posts directly may be useful - or similarly, a dataset that was more targeted towards specific end-use or application level goals.

In the spirit of innovating and not simply taking an existing Kaggle dataset, a redesign (especially with more time or less interruptions) may include a refined dataset exploration and selection period, and more clarity upfront about towards which end our efforts would aim; instead of not knowing from the start where in a particular pipeline of data-to-product, or specifically a mental health application, focusing more intentionally on classification from the beginning, or being at a place where classification is supported by recommendation or retrieval, may be a more ideal way to redesign or advance this project.

Peer Review Responses

A concern from group 5 was our limited dataset size of 9,078 rows. We could have grabbed data posted on forums about our specific topics. We chose to align with Professor Weibel's team and stayed within Reddit. We searched for other subreddits to increase the amount of data points, but some topics are more popular than others. Only one subreddit deals with anger management. Eating disorders could have been expanded by incorporating topics like bulimia and anorexia. Overall, we're satisfied with our dataset.

This group also wanted to see us explore how a bot might be deployed to classify mental health posts. If we had more time, we could have investigated how a Reddit bot is made. Instead we have a barebones application since we were running low on time.

They also criticized our focus on only 6 topics. We focused on these 6 because those were the only ones from Professor Weibel's team that had existing subreddits we could scrape data from. We could have lumped panic with anxiety but left it on as a challenge for our model. We could have split eating disorders into separate topics like bulimia and anorexia. In either case, there are many paths we could have taken, but we still believe that we made the right choice with our topics chosen.

Group 18 was worried that anxiety may overlap with depression and panic and suggested implementing Word2Vec. We did not have time to implement Word2Vec, but the error rate on our MultiNB proved that it could somewhat see past the overlap. They also wanted more discussion on ethical considerations and biases which were addressed earlier.

Finally, they asked for more evaluation metrics, MultiNB started with only accuracy as a metric. We added precision, recall, and F1-score at their advice. LDA only had UMass at first, and we added the CV Coherence score.

References

Journal and Conference Papers

[1] T. Joachims, "Text categorization with Support Vector Machines: Learning with many relevant features," in Proc. Eur. Conf. Mach. Learn. (ECML), 1998.

[2] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint, arXiv:1810.04805, 2019.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," J. Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.

[4] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," Nature, vol. 401, pp. 788–791, 1999.

[5] J. Boyd-Graber, Y. Hu, and D. Mimno, "Applications of Topic Models," Found. Trends Inf. Retr., vol. 11, no. 2-3, pp. 143–296, 2017.

[6] S. C. Guntuku, D. B. Yaden, M. L. Kern, L. H. Ungar, and J. C. Eichstaedt, "Detecting depression and mental illness on social media: an integrative review," Curr. Opin. Behav. Sci., vol. 18, pp. 43–49, 2019.

Web-Based Sources and Documentation

[7] "Comprehensive Guide to Classification Models in Scikit-Learn," GeeksforGeeks, Jun. 17, 2024. [Online]. Available: https://www.geeksforgeeks.org/comprehensive-guide-to-classification-models-in-scikit-learn/. [Accessed: Mar. 11, 2025].

[8] R. Řehůřek, "GENSIM: Topic Modelling for Humans," Aug. 10, 2024. [Online]. Available: https://radimrehurek.com/gensim/. [Accessed: Mar. 13, 2025].

[9] "Matplotlib: Visualization with Python," Matplotlib. [Online]. Available: https://matplotlib.org. [Accessed: Mar. 13, 2025].

[10] "Multinomial Naïve Bayes," GeeksforGeeks, Jan. 29, 2025. [Online]. Available: https://www.geeksforgeeks.org/multinomial-naive-bayes/. [Accessed: Mar. 11, 2025].

[11] "MultinomialNB," Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html. [Accessed: Mar. 11, 2025].

[12] "NLTK," NLTK, Aug. 19, 2024. [Online]. Available: https://www.nltk.org. [Accessed: Mar. 13, 2025].

[13] "NumPy," NumPy. [Online]. Available: https://numpy.org. [Accessed: Mar. 13, 2025].

[14] "pandas," Pandas.pydata. [Online]. Available: https://pandas.pydata.org. [Accessed: Mar. 13, 2025].

[15] J. Payne, "Async PRAW: The Asynchronous Python Reddit API Wrapper," AsyncPRAW Documentation. [Online]. Available: https://asyncpraw.readthedocs.io/en/stable/index.html. [Accessed: Mar. 14, 2025].

[16] "Public Content Policy," Reddit Help, May 2024. [Online]. Available: https://support.reddithelp.com/hc/en-us/articles/26410290525844-Public-Content-Policy. [Accessed: Mar. 11, 2025].

[17] "Removing stop words with NLTK in Python," GeeksforGeeks, Jan. 3, 2024. [Online]. Available: https://www.geeksforgeeks.org/removing-stop-words-nltk-python/. [Accessed: Mar. 11, 2025].

[18] "scikit-learn: Machine Learning in Python," Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/index.html. [Accessed: Mar. 13, 2025].

[19] K. Sidak, "A Comprehensive Guide to Text Preprocessing with NLTK," Codefinity, Dec. 2023. [Online]. Available: https://codefinity.com/blog/A-Comprehensive-Guide-to-Text-Preprocessing-with-NLTK. [Accessed: Mar. 11, 2025].

[20] "spaCy," spaCy. [Online]. Available: https://spacy.io. [Accessed: Mar. 13, 2025].

[21] "Text Preprocessing in NLP," GeeksforGeeks, Oct. 3, 2024. [Online]. Available: https://www.geeksforgeeks.org/text-preprocessing-for-nlp-tasks/. [Accessed: Mar. 11, 2025].

[22] T. Tigerschiöld, "What is Accuracy, Precision, Recall and F1 Score?" Labelf, Nov. 17, 2022. [Online]. Available: https://www.labelf.ai/blog/what-is-accuracy-precision-recall-and-f1-score. [Accessed: Mar. 11, 2025].

[23] "Topic Extraction with Non-negative Matrix Factorization and Latent Dirichlet Allocation," Scikit-learn. [Online]. Available:

https://scikit-learn.org/stable/auto_examples/applications/plot_topics_extraction_with_nmf_lda.html. [Accessed: Mar. 11, 2025].

[24] "Topic Modeling Using Latent Dirichlet Allocation (LDA)," GeeksforGeeks, Jun. 11, 2024. [Online]. Available: https://www.geeksforgeeks.org/topic-modeling-using-latent-dirichlet-allocation-lda/. [Accessed: Mar. 11, 2025].

[25] E. Zvornicanin, "When Coherence Score Is Good or Bad in Topic Modeling?" Baeldung, Feb. 28, 2025. [Online]. Available: https://www.baeldung.com/cs/topic-modeling-coherence-score. [Accessed: Mar. 11, 2025].