Andrew Seal

Capstone 3: Natural Language Processing for Suicide Detection

1. Introduction:

According to Baylor University, most people who die by suicide talk about it first, often reaching out to others for help or because they are in pain or distress.[1] Many people who do talk seriously about suicide can be helped with appropriate intervention and professional help[2].

While the rate of suicide has declined since peaking in 2018, nearly 46,000 people died by suicide in the United States in the year 2020. Today there are more ways than ever to connect digitally and many conversations about suicide happen online and on social media.

The purpose of this project is to develop a model to interpret written text and categorize it as suicidal or non-suicidal. This type of model could be used to review large text datasets and flag text using language consistent with suicidal postings. Such a system could be used by social media companies or by schools or parents who wish to monitor for suicidal tendencies in authors in order to intervene before anything catestrophic occurs. Additionally, a number of private firms including Betterhelp, Talkspace and 7 Cups offer online counseling including chat functions through an app. The ability to screen written communication through counseling apps could help therapy providers more easily identify patients who begin using language consitent with suicidal patients. With appropriate permissions, these apps could even monitor outgoing text messages or other written communications of patients to flag language consistent with suicidal tendencies even if it is not directed at the therapist.

2. Dataset

For this project, I used a dataset consisting of 232,074 social media posts from Reddit. I found the dataset on Kaggle, https://www.kaggle.com/datasets/nikhileswarkomati/suicide-watch.

Exactly half of the posts were taken from the 'SuicideWatch' subreddit. These have been given a categorization of 'Suicide.' The other half of the posts were from a 'teenager' subreddit, and these have been labeled as 'Non-Suicide.'

[1] https://www.baylor.edu/counseling_center/index.php?id=937125
[2] https://www.mayoclinic.org/diseases-conditions/suicide/in-depth/suicide/art-20044707

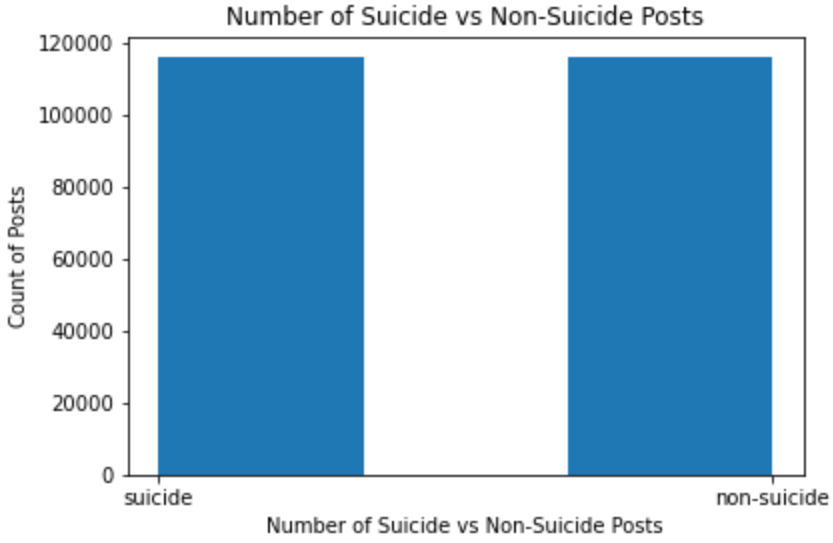

Figure 1. Histograms showing the balanced dataset.

3. Data Cleaning and Data Wrangling

To begin cleaning the data I ensured that the dataset did not have any duplicate posts or posts with null values (no text). Next, I began the process of normalizing the text in order to convert the messy social media posts into a neat machine-readable format.

The first steps in the process of text normalization include:

- Converting all text to lowercase
- Converting all hyperlinks and urls to standard text
- Converting all emojis and emoticons to text
- Removing punctuation and numerals
- Removing white spaces
- Ensuring all posts are written in the English language
- Expanding contractions

Additionally, there was an issue where the word “filler” was repeated throughout various postings where the word did not make sense in the context of the social media post. Given the prevalence of this word throughout the dataset, I decided to remove all instances of this particular word.

```
iteration 253 can constant masturbation lead to disinterest in girls asking for a friend
filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler f
iller filler filler filler filler filler
iteration 414 what are some good halloween movies filler filler filler filler filler filler filler filler filler filler filler
filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler filler f
iller
```

Figure 2. Example of posts containing multiple instances of the word ‘filler’.

The final steps in normalizing the text included:

- Removing stopwords utilizing the spacy dictionary
- Lemmatizing verbs to convert them to their lemma or stem

4. Exploratory Data Analysis

In the exploratory data analysis step, the relationship between variables is assessed. This particular dataset consisted simply of the text of the social media posting and a variable indicating whether the posting was 'suicide' if it was in the Suicide Watch forum or 'non-suicide' if the posting came from the Teenager forum.

Post Length

The first characteristic that I analyzed was the length of each post. For the dataset as a whole, the average (mean) length of a post was 344 characters. However, within the suicide forum, the average length of a post was 498 characters, while the average length of a post in the teenager forum was merely 190 characters.

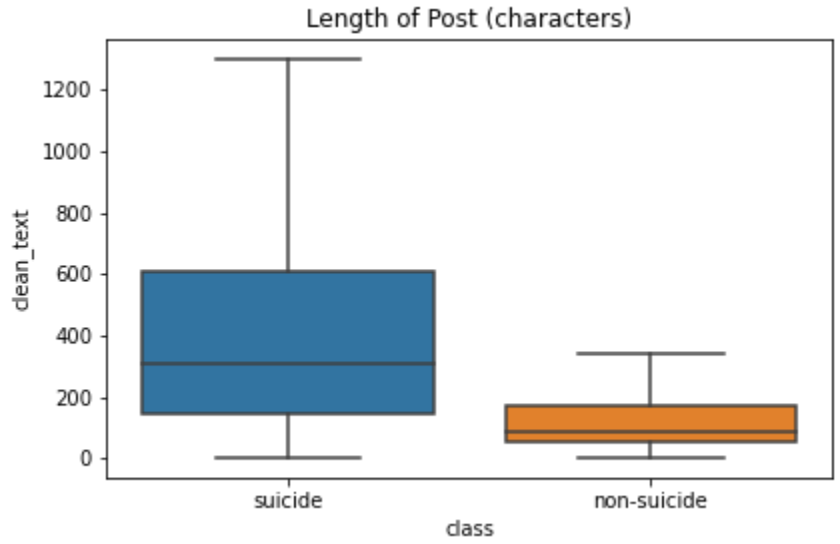

Figure 3. Boxplot comparing the length of posts from each subreddit

An independent t-test can check for statistically significant differences in the mean of two groups. In order to evaluate whether this difference in means in the lengths of the posts were statistically significant, an independent t-test was conducted. The p-value was very near to zero meaning we can reject the null hypothesis that the mean length of posts was equal between the two groups. Given the

differences in the length of the posts between the two groups was statistically significant, a new variable was created indicating the length of each post.

Sentiment

Next, a sentiment analysis of the text was conducted. A python library for natural language processing called TextBlob was utilized to conduct the sentiment analysis. This tool categorizes words as having either a negative or positive sentiment and then calculates an overall sentiment of the text based on the pool of words in each post. The score that I analyzed for this project was the 'polarity' of the text which is assigned a value between -1 and 1 where a score of -1 would indicate a very negative sentiment and a score of +1 would indicate a positive sentiment.

Each post was evaluated using the TextBlob measure of polarity and a score was assigned to each post. Unsurprisingly the average sentiment of posts in the suicide forum was lower than the average post in the teenager forum. The average sentiment of a post in the suicide forum was scored at -0.0223 while the average sentiment of a post in the teenager forum was 0.03.

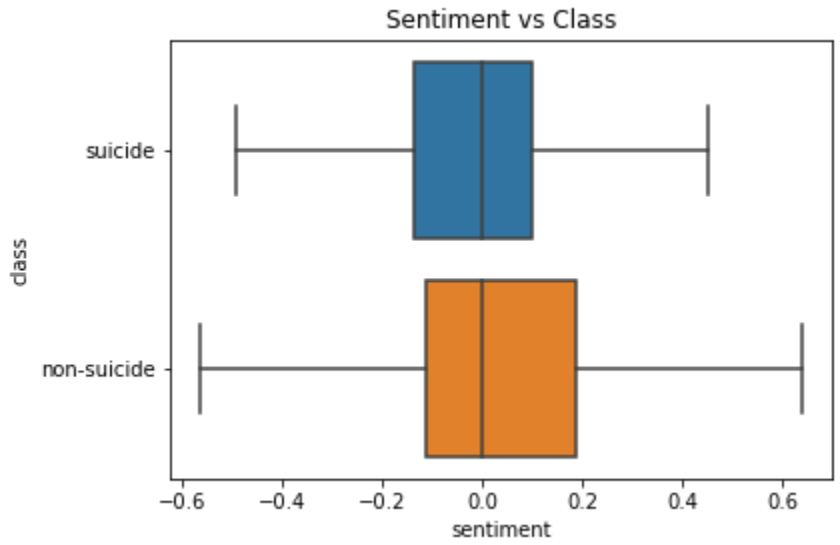

Figure 4. Boxplot comparing the sentiment of posts from each group

Once again an independent t-test was conducted to see if the differences in the mean between these two groups was statistically significant, and again the p-value was very near zero meaning the null hypothesis that these two groups have the same mean, can be rejected. Given the differences in the sentiment between the two groups was statistically significant, a new variable was created showing the sentiment of each post.

Reading Level

Next, the reading level of each post was analyzed. To analyze reading level a tool from the textstat library called the Flesch-Kincaid grade level was employed. This tool analyzes the complexity of words (total syllables) and the length of sentences to assign a 'grade level' to text. The grade level assigned to the text is supposed to roughly correspond to a US grade level meaning that a score of 9.3 could be read by a highschool freshman. However, there is no upper bound to the Flesch-Kincaid grade level and a negative score is possible.

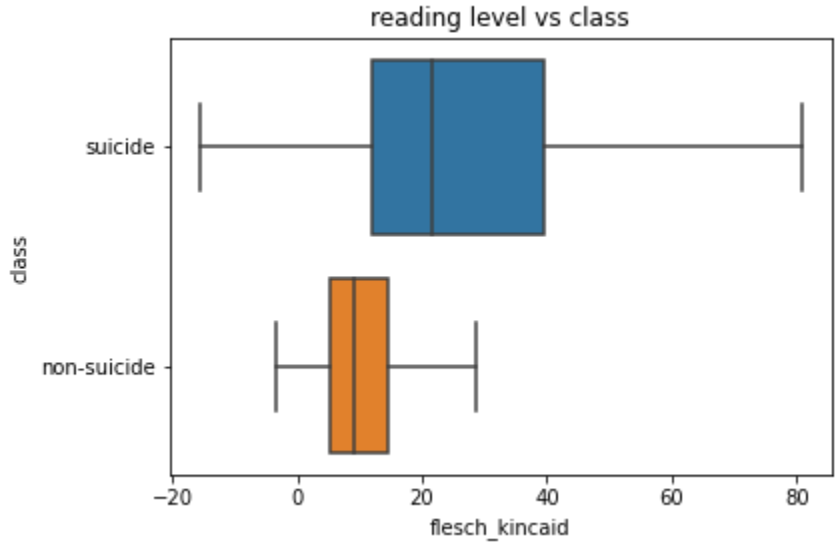

Figure 5. Boxplot comparing the reading level of posts from each subreddit

The mean Flesch-Kincaid score for posts in the teenager forum was 17.6 while the mean score in the suicide watch forum was 32.6. Again, an independent t-test was conducted to evaluate whether the differences in the groups were statistically significant. The p-value on this t-test was very near zero meaning the null hypothesis (that the means are equivalent) can be rejected. Given the fact that the differences in reading level between these two groups was statistically significant, a new variable was created indicating the reading level of each post.

Repetition

Next, the use of repetition was analyzed as often repeating words can be used to convey emphasis or to indicate the importance of certain words. For this project repetition was simply defined as having the same word in the text two or more times. Unsurprisingly, most of the posts did not employ repetition, so the median post had no instances of repetition in each class. However, the average did vary among the two groups where on average the 'non-suicide' posts employed repetition more frequently than the posts from the 'suicide' class. Once again an independent t-test confirmed the difference in the means between the two groups was statistically significant and a new variable was created indicating the amount of repetition in each post.

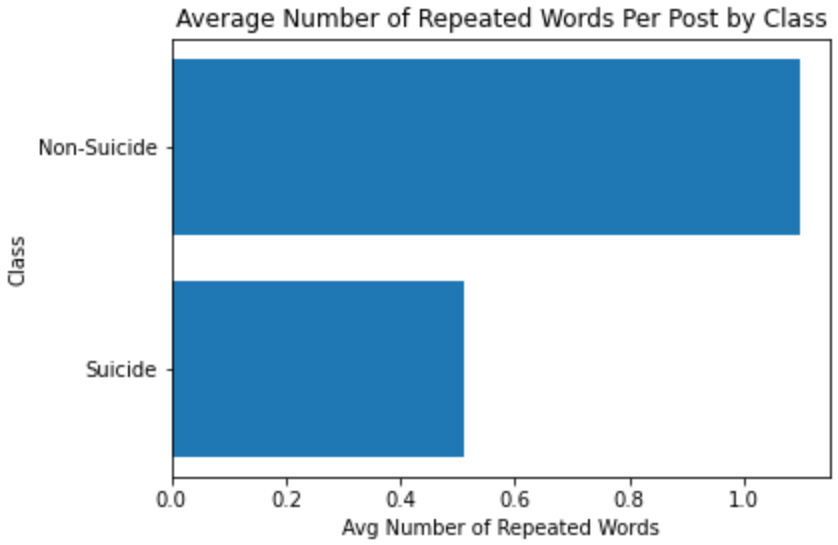

Figure 6. Bar chart comparing average number of repeated words in each post by class

6. Vocabulary

The next step was to compare the vocabulary used in the 'Teenager' forum with the vocabulary used in the 'Suicide Watch' forum to see if there were differences between the two groups. The first method used to compare vocabulary was to look at which words were used most frequently by each group.

Word clouds were generated showing the most frequently used words largest in each word cloud.

The most frequently used words in the suicide watch forum are shown below:

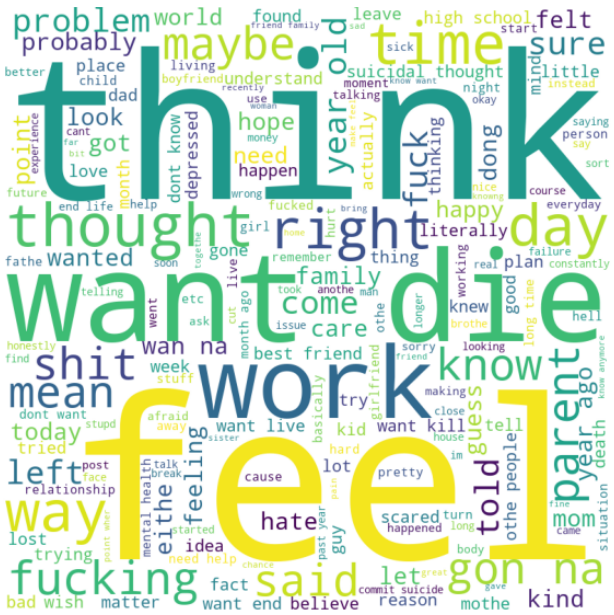

Figure 7. Wordcloud showing most frequently used words in 'Suicide Watch' Forum

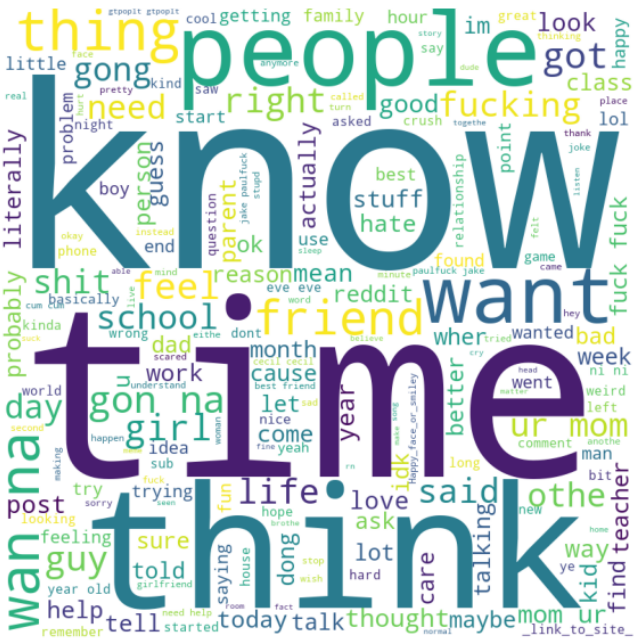

Figure 8. Wordcloud showing most frequently used words in 'Teenager' Forum

7. Finding Most Predictive Words

Methodology

The second method used to compare vocabulary was to look at the probability that the presence of a certain word predicts the post belongs to one class or another. To do so, a count vectorizer is fitted to the text of the posts and converted to a sparse matrix. A predictive model, in this case a naive bayes model, is trained on the sparse matrix.

Next, a data set is created where each row contains exactly one word (the identity matrix) for all of the words used in all of the posts . We use the naive bayes model to make predictions on this identity matrix and are able to get the predicted probabilities of each word. The output is the word and the probability of a post being from the 'suicide' class given the presence of the word.

Results

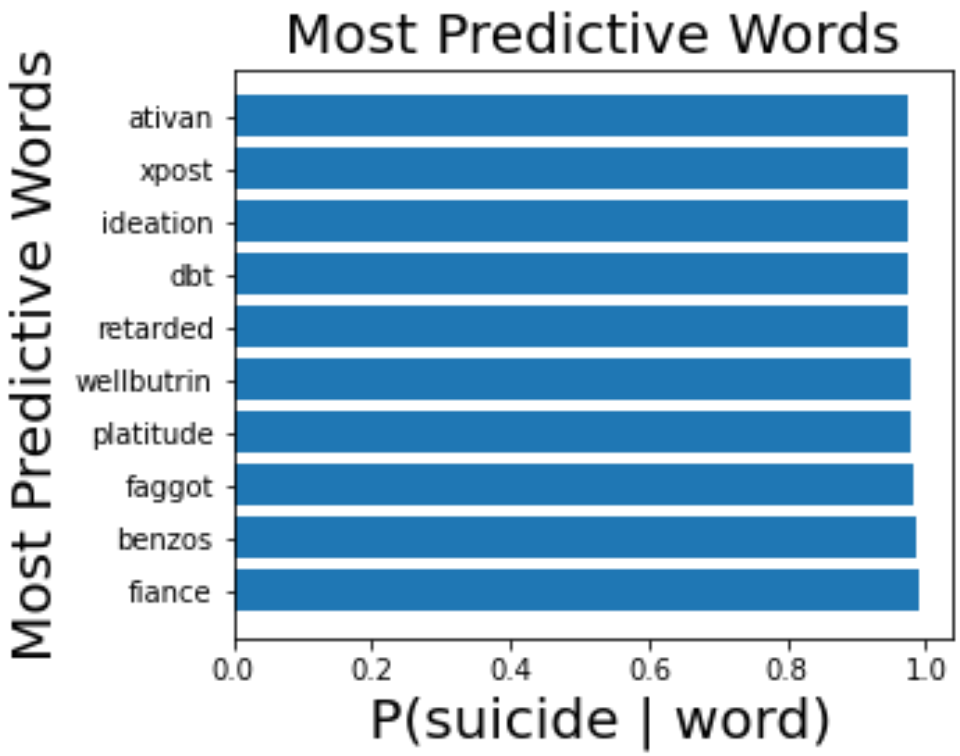

Figure 9. Most predictive words for suicide class

Interpretation of Most Predictive Words

The word 'fiance' is most predictive of a post being in the suicide class. In terms of differentiating between the teenager class, this makes sense as most teenagers are not yet engaged to be married.

In analyzing the texts from the suicide class that mention the word fiance, the most common refrain is that the finance left or cheated and the person posting is lamenting the loss of their significant other like the poster who wrote "posted last week about being dumped by my verbally abusive fiance i made it through the weekend but…".

Other uses describe leaving their fiance behind when they kill themselves and other posts using this word seem to list their fiance among the good things they have in their lives.

The next most predictive word is 'benzos'. This is an abbreviated form of the word benzodiazepines, which are a class of drug used to treat anxiety, insomnia and seizures. Posts using the word 'benzos' most commonly describe taking benzodiazepine as part of a suicide attempt like the author who posted "i planned my suicide for my last night in my childhood home this weekend i will … overdose on a combo of

benzos and opiates and…". Some posters describe taking benzos recreationally or to self medicate and sometimes in combination with alcohol. Other posts describe attempting to stop taking benzodiazepine but having difficulty with withdrawal and problems weaning off the drug. A few talk about using benzos to help treat anxiety.

The next most predictive word is 'faggot' which is a derogatory term for homosexual individuals. Many posts using this particular term are self loathing such as the poster who wrote: 'no one fucking likes me at all mainly cause i was a massive faggot in middle school'. Others describe being insulted for being perceived as homosexual.

The word 'platitude' was also shown to be very predictive of posts in the 'suicide' class. The word platitude is defined as 'a remark or statement, especially one with a moral content, that has been used too often to be interesting or thoughtful.' Many of the posts using this word describe not buying into platitudes about not killing yourself including the author who wrote "i have heard all the suggestions and tried treatments and mulled over every fucking platitude i still hate myself and want to die." The platitudes they reject are not explicit in the text but may include common refrains like 'suicide is a permanent solution to a temporary problem'. The people using this term seem to suggest the only rationale others can give them for not killing themselves are not meaningful but are in fact platitudes.

The next word, 'wellbutrin' is also shown to be highly predictive of posts in the 'suicide' class. Wellbutrin is the brand name of Bupropion, which is an antidepressant used to treat depression and to help people quit smoking. Most people who are using this term are either describing their treatment with wellbutrin or are talking about overdosing with the drug.

The word 'retarded' is also highly predictive of a post being from the 'suicidal' class. Most people using this term in the posts are writing self-loathing posts or describing how they think people perceive them. The poster who wrote, "i am functionally retarded i should give up i am…" seems typical of posters who use this particular word.

DBT stands for Dialectical Behavior Therapy. It is a type of 'talk therapy' designed for people who feel emotions very intensely. The use of this word in our dataset is highly predictive of belonging to the 'suicide' class. People who post using this word most commonly are relating their personal experience with DBT including the person who wrote "i will not say it is totally ineffective but i kind of expected more from a dbt intensive theapist."

Use of the word 'ideation' is highly predictive of a post belonging to the 'suicide' class. Most people using this word are describing suicidal ideation or simply thinking about suicide. Many of the people using this word simply tell others that they are experiencing or living with suicidal ideation. Some describe seeking treatment or being hospitalized for their suicidal ideation and a few describe thinking through a plan or method for suicide.

The next word, 'xpost' is used when someone crossposts their concern to other forums. For the most part people using this word indicate they have crossposted their issue on the depression forum, however some say they have crossposted on a relationship forum or survivor of abuse forum.

Finally, the word 'ativan' is highly predictive of a post belonging to the 'suicide' class. Ativan is a medication used to treat anxiety. It is in a class of drugs called benzodiazepine which is to say it is a brand name of the second most predictive word 'benzos'. People mentioning 'ativan' in their posts talk

about overdosing on ativan and combining ativan and alcohol either to intoxicate themselves or to attempt to overdose. One example of this is the author who wrote, “i had my first suicide attempt when i was in th grade i slashed my wrists and took the ativan.” Some posts mention taking ativan in an attempt to treat their depression or other symptoms or to help with insomnia.

Overall the most predictive words seem to be used to talk about one or more of the following:

- Relationships gone bad (fiance)
- Talking about suicide (xpost, platitudes)
- Treatment for depression (dbt, benzos, ativan, wellbutrin)
- Self loathing (faggot, retarded)
- Suicide attempts (benzos, ativan, wellbutrin, ideation, fiance)

8. Train Test Split

In advance of modeling, I split the data into a training set consisting of 75% of the recent sales and a training set consisting of the remaining 25% of the data. Each of the models was fit and optimized on the training set and then tested on the test data in order to evaluate the performance of the model.

9. Modeling

The textual data were evaluated using three different models, Multinomial Naive Bayes , Random Forest and Logistic Regression. Each of the three models was run after transforming text into a sparse matrix using both Count Vectorizer and also TF-IDF Vectorizer. Then the dataset was modeled using only the text data and also using the engineered features that we created in the Exploratory Data Analysis step. These engineered features were ‘Length of Post’, ‘Sentiment’, ‘Reading Level’, and ‘Repetition’.

There were a total of 12 models that were analyzed. To ensure consistency, each model was analyzed using a minimum document frequency (min_df) of 3 and a ngram range of (1,2). Min_df of 3 means the vectorizer will ignore words that appear in fewer than 3 documents. Setting an n_gram range of (1,2) establishes a minimum and maximum for the n-grams to be analyzed. In this case we are only analyzing unigrams and bigrams.

For each model the relevant hyperparameters were tuned and the results of each model were compared in order to select the best model. For the Random Forest model, it was computationally very expensive to tune the hyperparameters given the large size of our dataset. For this reason, the hyperparameters were tuned on a subset of the data.

Multinomial Naive Bayes

The first model employed was Multinomial Naive Bayes. For this algorithm I elected to optimize the Alpha hyperparameter which is an additive smoothing parameter. I then used a random search to optimize the Alpha hyperparameter.

Model: Naive Bayes	Count Vect. Text Only	Count Vect. All Features	TF-IDF Text Only	TF-IDF All Features
Accuracy	0.92	0.923	0.922	0.925
Precision	0.893	0.904	0.89	0.9
Recall	0.953	0.946	0.962	0.956
ROC-AUC Score	0.971	0.971	0.98	0.98
True Suicide	27518	27290	27751	27593
False Suicide	3295	2902	3433	3076
True Non-Suicide	25856	26249	25718	26075
False Non-Suicide	1343	1571	1110	1268
Optimal: Alpha	0.4	0.4	0.1	0.1

Figure 10. Naive Bayes Evaluation Metrics and Optimal Hyperparameters

The Naive Bayes models produced fairly robust results with ROC-AUC scores ranging between 0.971 and 0.98. Overall the TF-IDF vectorizer seemed to perform slightly better than the Count Vectorizer as measured by accuracy, which is a measure of how many times the machine learning model was overall.

For our use case, the False Non-Suicide is the most important measure. This is when the model predicts the author is not suicidal but actually may be. If the model predicts someone is not suicidal and actually is, then parents or concerned parties may have not opportunity to intervene in a crisis situation. The two models that only examined the text seemed to predict fewer False Non Suicide and therefore had slightly higher recall scores.

Random Forest

The next model I chose to test and optimize was a Random Forest.

The hyperparameters of n-estimators and maximum depth were optimized using Grid Search cross validation. However, given the size of the dataset, with more than 200,000 posts, running a grid search was computationally very expensive, so the hyperparameters were optimized using a subset of the dataset. Once again both Count Vectorizer and TF-IDF Vectorizer were utilized and the Random Forest was run looking at only the text and again using the engineered features.

Model: Random Forest	Count Vect. Text Only	Count Vect. All Features	TF-IDF Text Only	TF-IDF All Features
Accuracy	0.881	0.841	0.878	0.865
Precision	0.909	0.888	0.905	0.895
Recall	0.847	0.779	0.842	0.824
ROC-AUC Score	0.952	0.926	0.949	0.942
True Suicide	24434	22470	24301	23790
False Suicide	2449	2821	2544	2779
True Non-Suicide	26702	26330	26607	26372
False Non-Suicide	4427	6391	4560	5071
Optimal: Max Depth	150	55	135	108
Optimal: n_estimators	841	177	3162	5011

Figure 11. Random Forest Evaluation Metrics and Optimal Hyperparameters

The Random Forest was not as robust of a model as the Naive Bayes. The top performing Random Forest Model utilized the Count Vectorizer and only the text data. This model resulted in an ROC-AUC score of 0.952 and a Recall Score of 0.847 . It also produced the fewest False Non-Suicide predictions.

Logistic Regression

The final model I utilized was a logistic regression. For this model the hyperparameter of C (the strength of regularization) was optimized using a grid search cross validation. As with the other models ,both Count Vectorizer and TF-IDF Vectorizer were compared and the Logistic Regression was run looking at only the text and then again using the engineered features in addition to the text.

Logistic Regression	Count Vect. Text Only	Count Vect. All Features	TF-IDF Text Only	TF-IDF All Features
Accuracy	0.931	0.931	0.996	0.996
Precision	0.955	0.955	0.995	0.995
Recall	0.903	0.903	0.996	0.996
ROC-AUC Score	0.978	0.978	0.999	0.999
True Suicide	26057	26069	28753	28750
False Suicide	1220	1215	143	148
True Non-Suicide	27931	27936	29008	29003
False Non-Suicide	2804	2792	108	111
Optimal: C	0.1	0.1	10	10

Figure 12. Logistic Regression Evaluation Metrics and Optimal Hyperparameters

The Logistic Regression produced more robust models than the Random Forest or the Naive Bayes. On all measures the Logistic Regression with the TF-IDF Vectorizer outperformed the model using the Count Vectorizer. When we analyzed only the text, the model was slightly better than when analyzing all features but the difference was quite small. The Logistic Regression model using a TF-IDF vectorizer and analyzing only the text produced the best results. It had the highest ROC-AUC score, the highest Recall Score and resulted in the fewest False Non-Suicide prediction.

7. Selecting a Model

For this project I used three different regression algorithms to model the dataset with the objective of finding the best model to predict a dependent variable (Class: Suicide or Non-Suicide) based on the features of the text. Each of these three algorithms was evaluated using both the full feature set (the text data and the engineered features) and only the text data. For each model, vectors from the Count Vectorizer and the TF-IDF vectorizer were used. In all, a total of 12 models were run.

For each model I used a grid search cross-validation to optimize the hyperparameters.

A key metric to compare the overall performance of the models is the ROC-AUC score. Other metrics include accuracy, precision and Recall. In addition, the F1 Score, the harmonic mean of precision and recall can be considered.

The importance of each metric depends to some extent on the business case. For this project, we are hoping to come up with a model that will predict text that is consistent with suicidal authors. The idea is to intervene before a crisis occurs. In this case, a false positive may bring unwanted attention to an author who is not suicidal. This is not really a life or death mistake. However, a false negative would mean ignoring someone who is truly suicidal which is a more grave error. Therefore, I would consider

Recall more important than Precision when analyzing these models. Furthermore, I found it helpful to look at confusion matrices to compare models.

Model	Accuracy	Precision	Recall	F1	ROC_AUC
Naive Bayes (Count Vectorizer - Text Only)	0.92	0.893	0.953	0.922	0.971
Naive Bayes (Count Vectorizer - All Features)	0.923	0.904	0.946	0.924	0.971
Naive Bayes (TF-IDF - Text Only)	0.922	0.89	0.962	0.924	0.98
Naive Bayes (TF-IDF - All Features)	0.925	0.9	0.956	0.927	0.98
Random Forest (Count Vectorizer - Text Only)	0.881	0.909	0.847	0.877	0.952
Random Forest (Count Vectorizer - All Features)	0.841	0.888	0.779	0.83	0.926
Random Forest (TF-IDF - Text Only)	0.878	0.905	0.842	0.872	0.949
Random Forest (TF-IDF - All Features)	0.865	0.895	0.824	0.858	0.942
Logistic Regression (Count Vectorizer - Text Only)	0.931	0.955	0.903	0.928	0.978
Logistic Regression (Count Vectorizer - All Features)	0.931	0.955	0.903	0.929	0.978
Logistic Regression (TF-IDF - Text Only)	0.996	0.995	0.996	0.996	0.999
Logistic Regression (TF-IDF - All Features)	0.996	0.995	0.996	0.996	0.999

Figure 13. Evaluation Metrics of each model

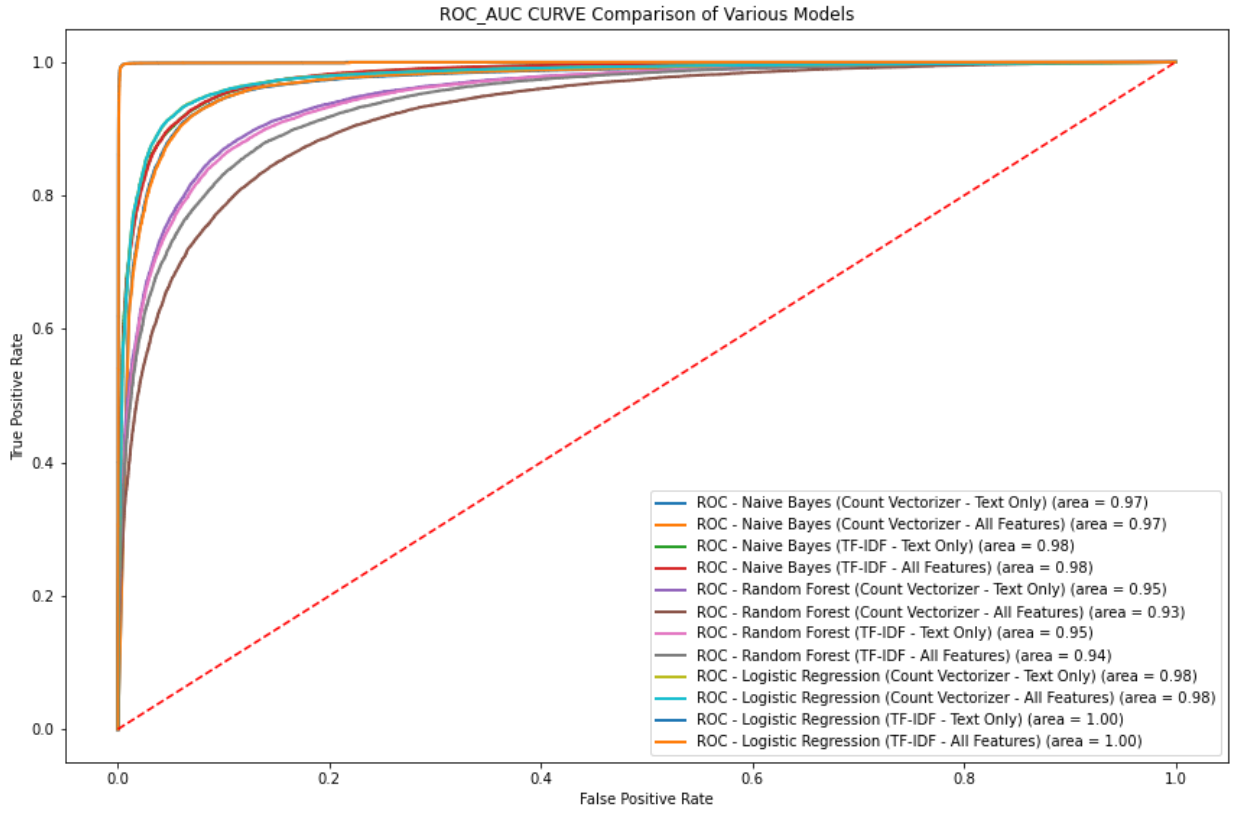

Figure 14. Comparing ROC-AUC curves of each model

Model	True Positive	False Positive	True Negative	False Negative
Naive Bayes (Count Vectorizer - Text Only)	27518	3295	25856	1343
Naive Bayes (Count Vectorizer - All Features)	27290	2902	26249	1571
Naive Bayes (TF-IDF - Text Only)	27751	3433	25718	1110
Naive Bayes (TF-IDF - All Features)	27593	3076	26075	1268
Random Forest (Count Vectorizer - Text Only)	24434	2449	26702	4427
Random Forest (Count Vectorizer - All Features)	22470	2821	26330	6391
Random Forest (TF-IDF - Text Only)	24301	2544	26607	4560
Random Forest (TF-IDF - All Features)	23790	2779	26372	5071
Logistic Regression (Count Vectorizer - Text Only)	26057	1220	27931	2804
Logistic Regression (Count Vectorizer - All Features)	26069	1215	27936	2792
Logistic Regression (TF-IDF - Text Only)	28753	143	29008	108
Logistic Regression (TF-IDF - All Features)	28750	148	29003	111

Figure 15. Comparing Confusion Matrix Results of each model

8. Review of Selected Model

After comparing the models the Logistic Regression (TF-IDF - Text Only) model was selected. This model had the highest ROC-AUC score and the highest recall score. It had the highest number of true positive predictions and the lowest total number of false negative predictions.

Selected Model	Logistic Regression: TF-IDF Text Only
Accuracy	0.996
Precision	0.995
Recall	0.996
ROC-AUC Score	0.999
---	---
True Suicide	28753
False Suicide	143
True Non-Suicide	29008
False Non-Suicide	108
---	---
Optimal: C	10

Figure 16. Select Metrics of Prefered Model

Confusion Matrix of Prefered Model

	Predicted Non-Suicide	Predicted Suicide
Actual Non-Suicide	29008	143
Actual Suicide	108	28753

Figure 17: Confusion Matrix of Prefered Model

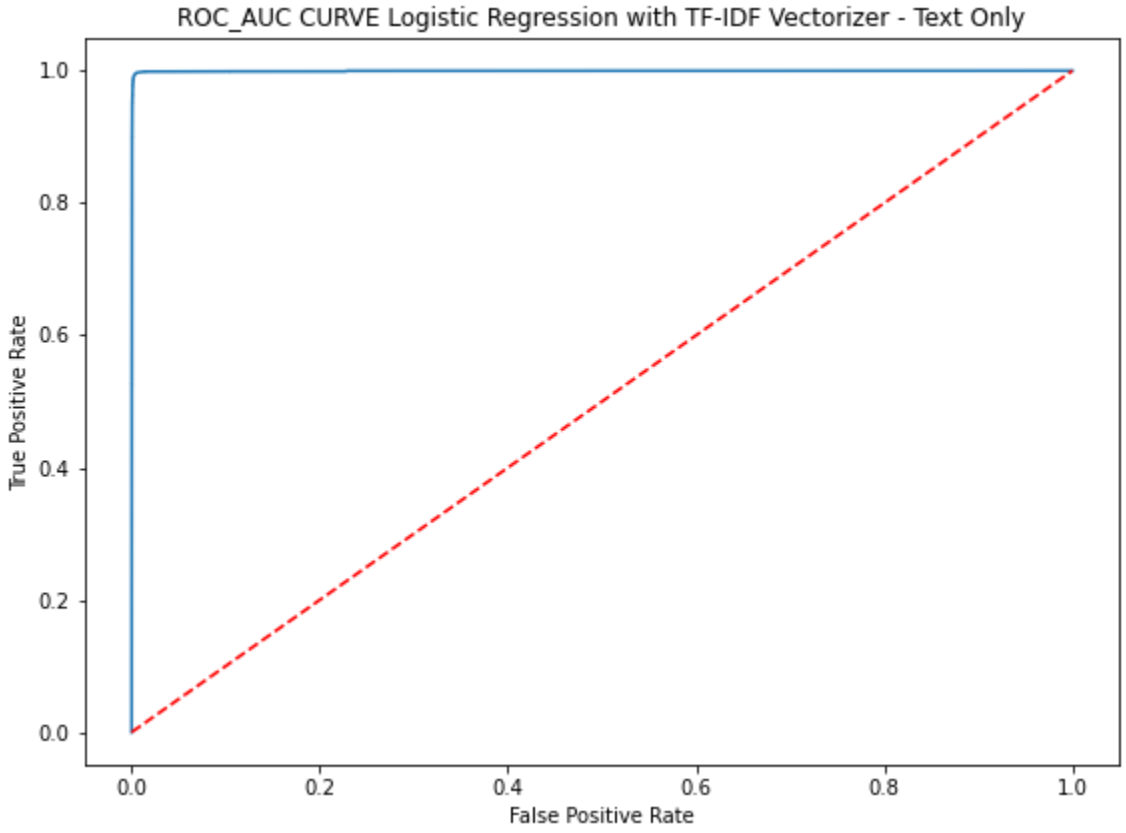

Figure 18: ROC-AUC Curve of Prefered Model

9. Look at Mis-Identified Predictions

The next part of the analysis was to delve into which text was mis-predicted to see if that would give us insights into how or why text was mis-identified. In order to do this, I looked at data that were mischaracterized and then looked at the predicted probabilities of each post to evaluate the cases where a given post was predicted most strongly to be of one class but actually belonged to a different class.

False Positives

There are two types of mis-identified data. The first is the False Positive case where a post was predicted to be a suicidal author but was actually a teenager. As mentioned previously, I would characterize this as the less serious case. I reviewed the postings of the posts most strongly predicted to be suicide but that were actually teenager postings.

Many of the false positive cases were long posts, so I will summarize the findings.

1) Text Read, “i hate myself i want to die” - i can see how this would be categorized as suicide
2) The next post was about how to speak to people who are suicidal. It was posted in the teenager forum but was dealing with suicide.
3) Text read: “i am killing myself bye y’all “ ’ - I can see how this would be categorized in suicide
4) The next post talks about how the person posting it wants to die / no longer wishes to live. It was about in part about suicide but posted in the teenager forum
5) The next post was from a teenager who was having a hard time. They indicate they do not want to live but do not really want to die. While it was posted in the teenager forum, it dealt with issues that would likely be found in a suicide watch forum.

For the false positive cases I looked at the five posts most highly predicted to be in the suicide class. They mostly dealt with suicide even though they were posted in the teenager forum..

False Negatives

The next case was the more serious case. This is the False Negative case where a post was predicted to be a teenager forum but was actually from the suicide class. Many of the posts were brief so I have pasted them below.

iteration	5	i am suicidalnuts
iteration	6	nopeend
iteration	7	helpresolved
iteration	8	hiidk
iteration	9	liablei do it to myself
iteration	10	i am overdosinggoodbye
iteration	11	asdgagssadgafds
iteration	12	am i asshole_link_to_site_
iteration	13	i am about todisappointed_face
iteration	14	i have no friendssee_link_to_site_

Figure 19: Text from False Negative Cases

Looking at the false negative cases, it seems that in general the poster was not always clear about their desire to kill themselves. Also, these posts tended to be very short. Additionally, many posts have misspellings or words that are not spaced properly. For future work, therefore, it might be worth looking into ways to identify and correct these spacing and spelling errors prior to prediction. However, it does not seem to me that obvious posts about suicide are being overlooked as many of these errors could have been made my a human annotator.

9. Thresholding

When determining which class a given post belongs to, the model assigns a probability to each piece of text. The higher the probability, the more likely the model believes the post belongs to the suicide class.

All of the probabilities will lie between 0 and 1. By default, any posting with a probability higher than 0.5 are predicted to belong in the suicide class while those posts with a probability lower than 0.5 are predicted to belong to the non-suicide class.

However, it is possible to select a number other than 0.5 as the threshold for determining the class of a given post.

For this project, I looked at the impact of setting a threshold such that the F-2 Score would be optimized.

We will recall that the F-1 score is the harmonic mean of precision and recall. F-beta adjusts the weighting of precision and recall so that F-2 score increases the importance of recall and F-0.5 increases the weighting of precision. For the suicide detection, a False Positive would mean that someone is flagged as suicidal who is not in fact suicidal. However a False Negative would mean that someone who is really suicidal could be overlooked or ignored. Therefore, the False Negative case is more important in this instance, so an F-Beta score should weigh recall above precision.

Since we care more about recall than precision we can set our F-beta to 2 and adopt a threshold so that the F-2 score is optimized.

```
df_threshold[df_threshold['F-2 Score'] == df_threshold['F-2 Score'].max()]
```

	Threshold	Precision	Recall	F-2 Score	True Positive	False Positive	True_Negative	False Negative
9	0.45	0.993032	0.997436	0.996552	28787	202	28949	74

Figure 20: Finding the threshold so the F-2 score is optimized

Using the preferred model of Logistic Regression (TF-IDF - Text Only) the Precision, Recall, and F-2 Score is computed at a variety of thresholds. Then a threshold is selected so that the F-2 Score is highest. In this case the F-2 score was highest when the threshold was set at 0.45.

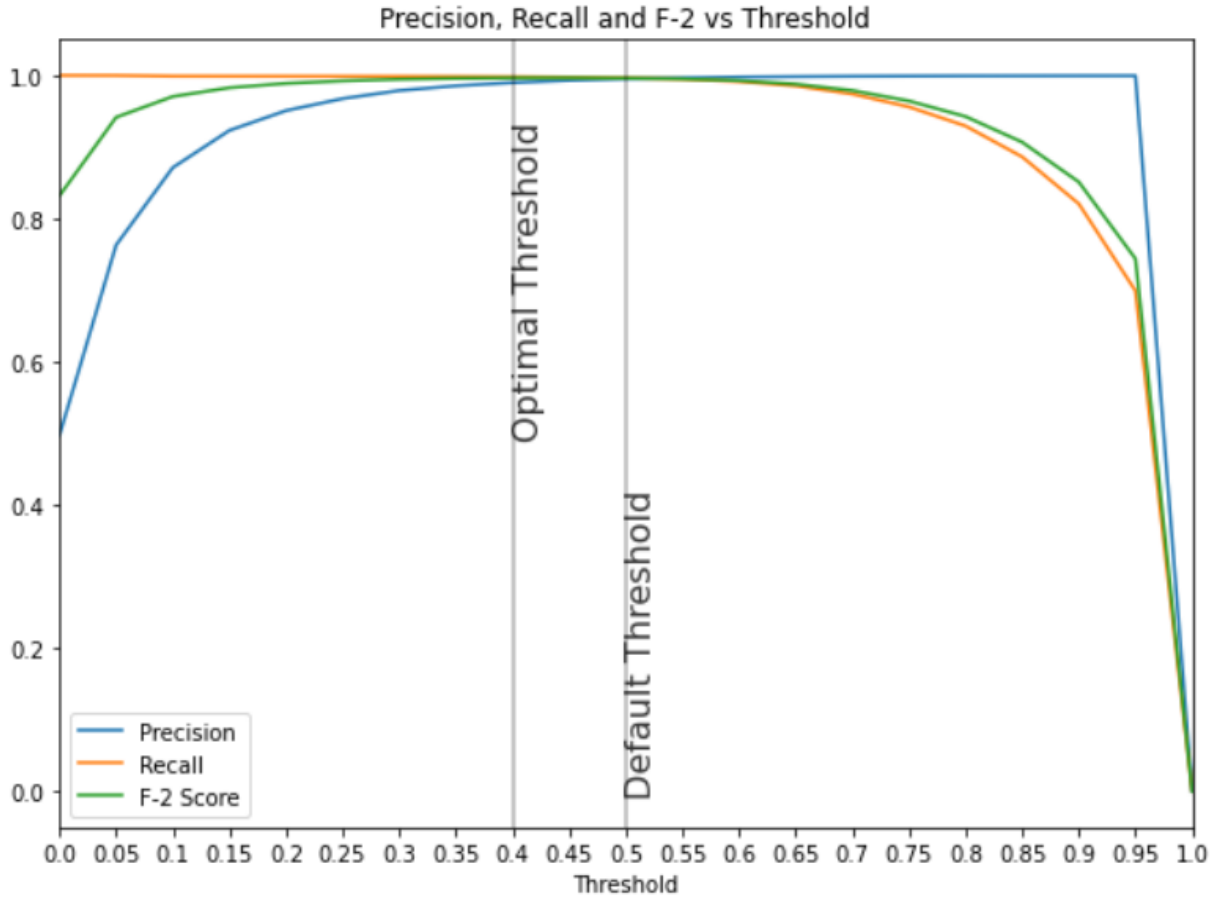

Figure 21: Plotting Default and Optimal Threshold

```
df_threshold[df_threshold['Threshold'].isin([0.45, 0.5])]
```

	Threshold	Precision	Recall	F-2 Score	True Positive	False Positive	True_Negative	False Negative
9	0.45	0.993032	0.997436	0.996552	28787	202	28949	74
10	0.50	0.995051	0.996258	0.996016	28753	143	29008	108

Figure 22: Key Metrics setting threshold to 0.45 vs setting threshold to 0.5 (default)

As expected, setting the optimal threshold to 0.45 reduced the number of False Negative predictions, increased the Recall and decreased Precision. In this case the false negatives decreased from 108 to 74, a decrease of 31.5%.

10. Evaluate Model with Text from a Different Source

The selected model does a very good job of differentiating the Suicide Watch social media posts from posts originating from the Teenager Forum. However, I thought it would be interesting to see how well the model predicts suicidal language outside of the context of social media postings. Therefore, I decided to apply the model to song lyrics. A quick search online identified some songs about suicide, and I also included songs that were not about suicide. The top performing model was applied to see how well the model performed with this new material from a different medium.

One of the major reasons that I decided to apply the model to songs, is that often songwriting is obscure and not necessarily as obvious as a social media posting would be.

	title	song	length	class	repetition	sentiment	flesch_kincaid
0	Smells Like Teen Spirit	load up on guns bring your friends it is fun t...	1280	0	24	-0.19892	102.7
1	Walking on Sunshine	oh ohhhh yeeeh i used to think maybe you loved...	1679	0	3	0.48771	141.7
2	Everybody Hurts	when your day is long and the night the night ...	886	1	2	0.1125	67.6
3	Happy and You Know It	if you are happy and you know it clap your han...	165	0	0	0.725	11.8
4	Wonderful World	i see trees of green red roses too i see them ...	598	0	0	0.373333	47.3
5	Never Gonna Give You Up	we are no strangers to love you know the rules...	1741	0	0	-0.158796	139.4
6	Save Myself	i gave all my oxygen to people that could brea...	1632	1	0	0.010606	135.5
7	Adams Song	i never thought i would die alone i laughed th...	1364	1	3	0.076	110.5
8	Cemetary Drive	this night walk the dead in a solitary style a...	911	1	0	-0.114418	73.8
9	Haunted	louder louder the voices in my head whispers t...	1181	1	3	-0.02381	100.7

Figure 23: Song Database including lyrics and features of selected songs

The lyrics of the songs were cleaned and entered into a dataframe along with the engineered features of repetition, sentiment, length and reading level for each song.

The top performing model was used to make predictions on the songs and made surprisingly accurate predictions.

	Predicted Non-Suicide	Predicted Suicide
Actual Non-Suicide	5	0
Actual Suicide	0	5

Figure 24: The selected model accurately predicted the class of all our songs

11. Summary

The goal of this project was to build a model that would use natural language processing to determining whether social media postings originated from a suicide forum or from a teenager forum. The idea was to develop a tool that would be able to detect suicidal tendencies of the person posting on social media. In the end the tuned model was able to predict the source of the posted text very well.

One potential use for this type of natural language processing is to integrate with online counseling and therapy providers such as Betterhelp, Talkspace or 7 cups. These companies offer online therapy and counseling and have chat functions and an app. These providers may benefit from having the ability to automatically screen written communication from their clients for language consistent with suicidal intentions.

Other stakeholders may include social media companies who may wish to monitor their content on behalf of parents or schools to help flag users who may be having suicidal thoughts.

Overall, the selected model seems to perform well enough to be useful although further conversations with these potential stakeholders could shed light on the needs for improved performance.

11. Next Steps

The model did a good job of predicting the source of the text. However, the language used in the teenager forum had real differences between the language used in the suicide watch forum meaning that it may be easier to predict the class of these different groups. It may be beneficial to conduct an analysis between more similar groups ie between depressed group and suicidal group in order to see how well this model performs in the case of more subtle differences in the language between groups.

In addition, while a few features were engineered such as length of the posting, grade level, sentiment, and repetition, in the end the inclusion of these engineered features did not substantially improve the performance of the model. Conversations with potential stakeholders or experts in the field may yield additional feature engineering that could be performed to improve upon the performance of the model.

Next, when investigating the text of the False Negatives, many of the posts had misspelled words or lacked spacing between words. For future work, therefore, it might be worth looking into ways to identify and correct these spacing and spelling errors prior to prediction.

Finally, a significant amount of time was spent optimizing the hyperparameters and each of our models includes one or two hyperparameters to tune. However it is likely that including additional parameters or larger and more refined grid searches could result in a more robust model.