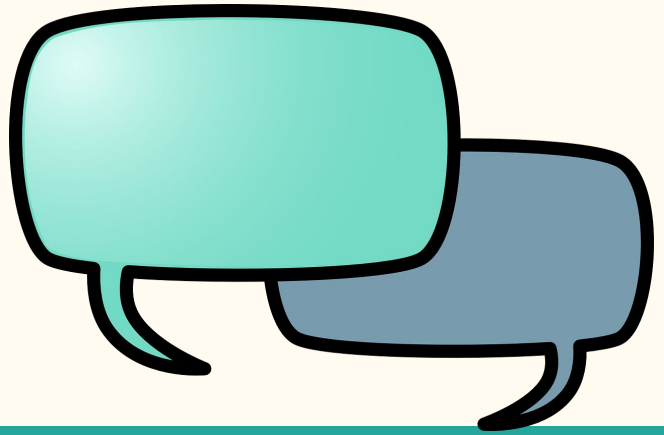
Suicide and Depression Sentiment Analysis - Team 13

Mohammad Ali Zahir - 40077619
Samantha Guillemette - 26609198
Marita Brichan - 40138194
Souvik Polol Alam - 40044092

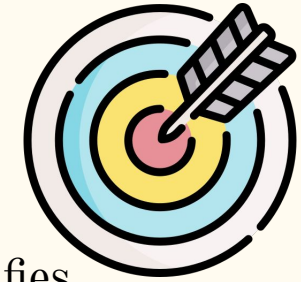
Why is this Important?

- Social media's impact on mental health
- Increase in depression and suicide rates

Predicting suicide and depression through text messages will help preventing suicide



Research Question



- Create a proper sentiment analysis algorithm which accurately identifies suicidal and non-suicidal messages

Goals:

1. Will our project be able to receive an accuracy measure of at least 90%, as we want to classify the most amounts of messages properly?

2. What are the keywords from a corpus which would lead to a message being clasified as suicidal/non-sucidal?

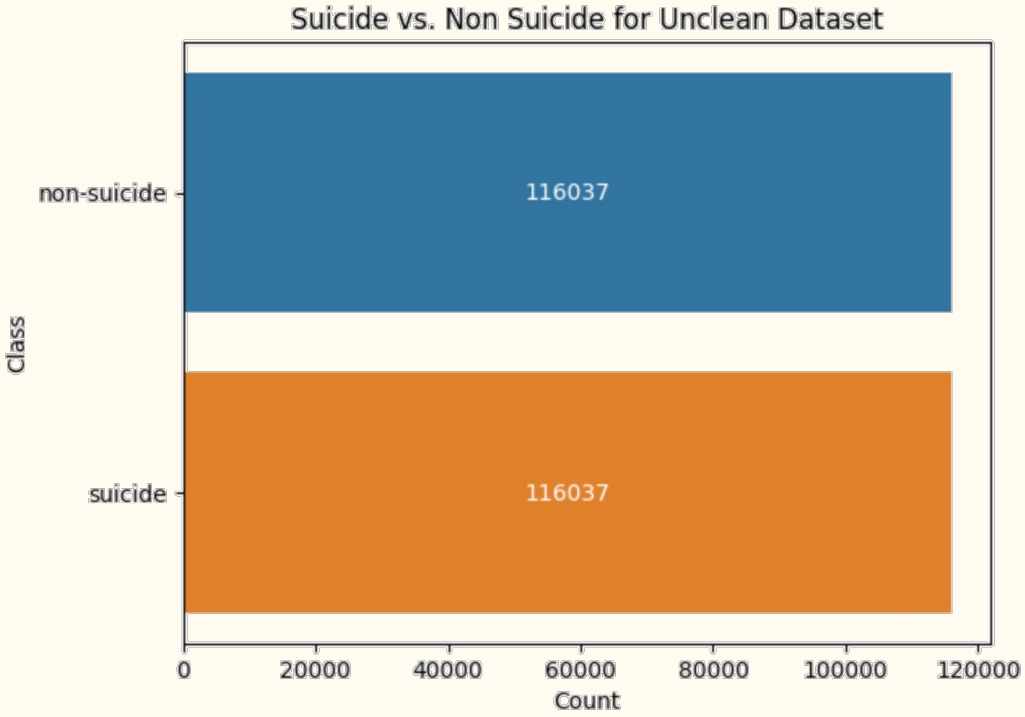
Dataset

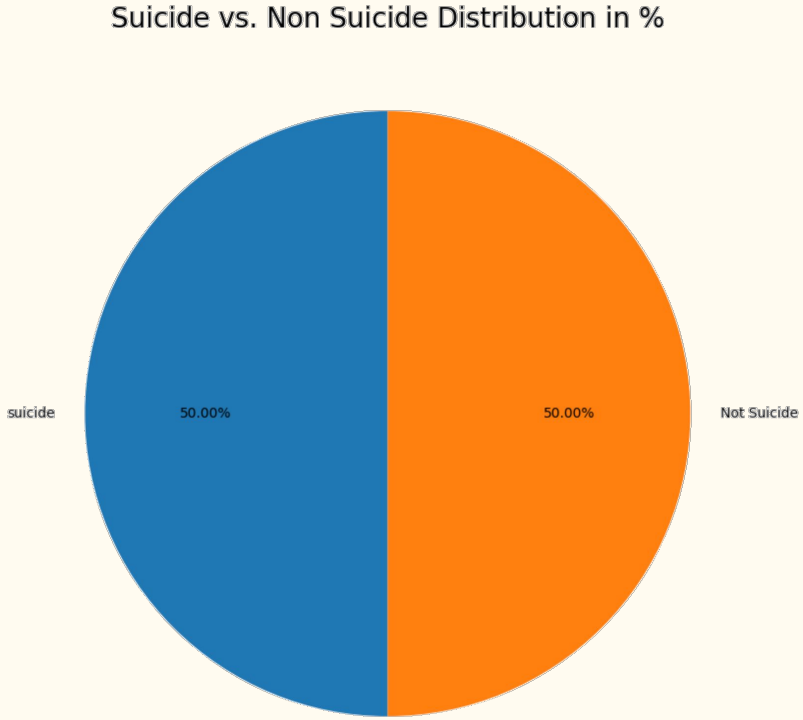
- Suicide and Depression detection from Kaggle
- Text messages from various social medias
- 232 074 unique values
- 177 MB
- Texts classified as SUICIDAL and NON-SUICIDAL

Detail | Compact | Column — 3 of 3 columns

#	text	class
2 – 348k	232074 unique values	2 unique values
	my last.Everything's becoming too overwhelming and once it's late enough into t...	
39	I'm trashLol I normally cringe at the self loathing posts here but honestly I'm such trash. Like lit...	suicide
40	Nice songs <3 Nice songs to vibe to- Loverboy/a-wall Come true/khai dreams Weak when ur around/bl...	non-suicide
41	What is the best way to do it?I'm not looking to be talked out of it. What would be the most effecti...	suicide
42	Man I hope someone finds thisI am drunk as fuck. I found that I have hodgkins lymphoma. I don't wan...	suicide
43	Today's fact is Reddit awards are expensive emojis	non-suicide

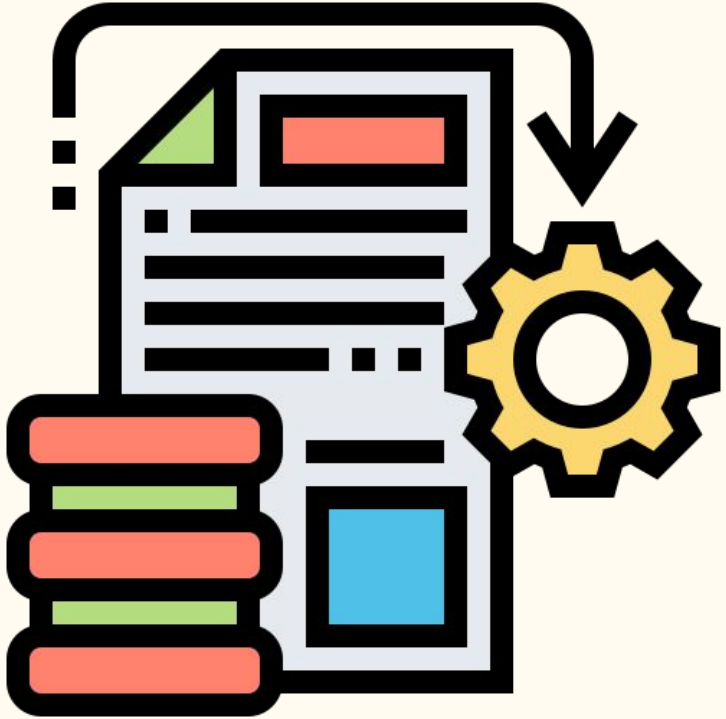
Data Preview



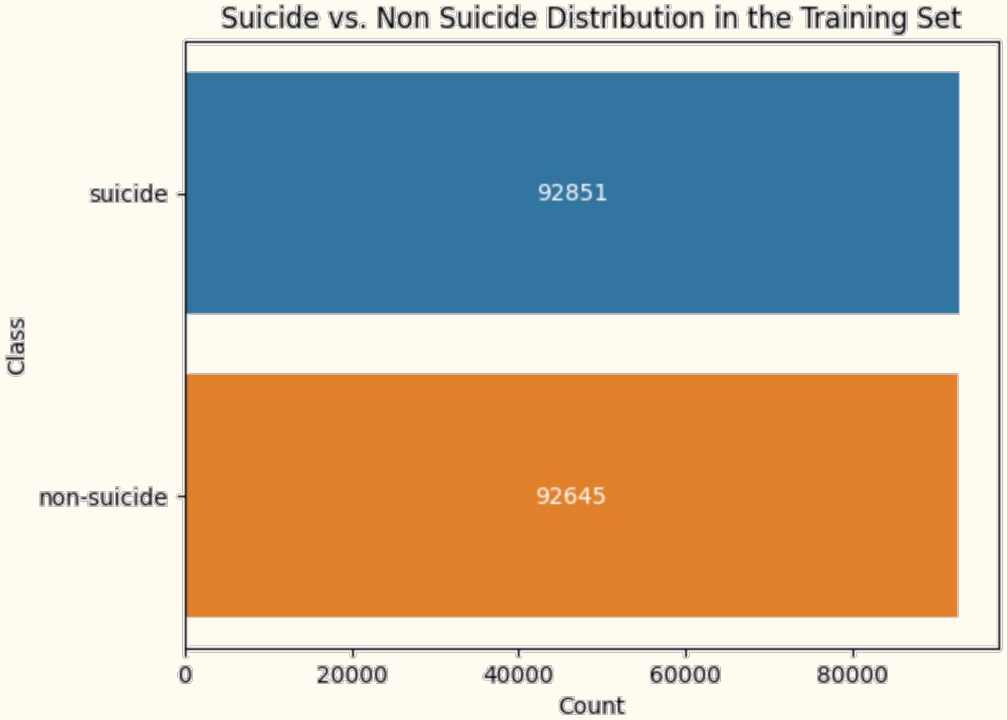


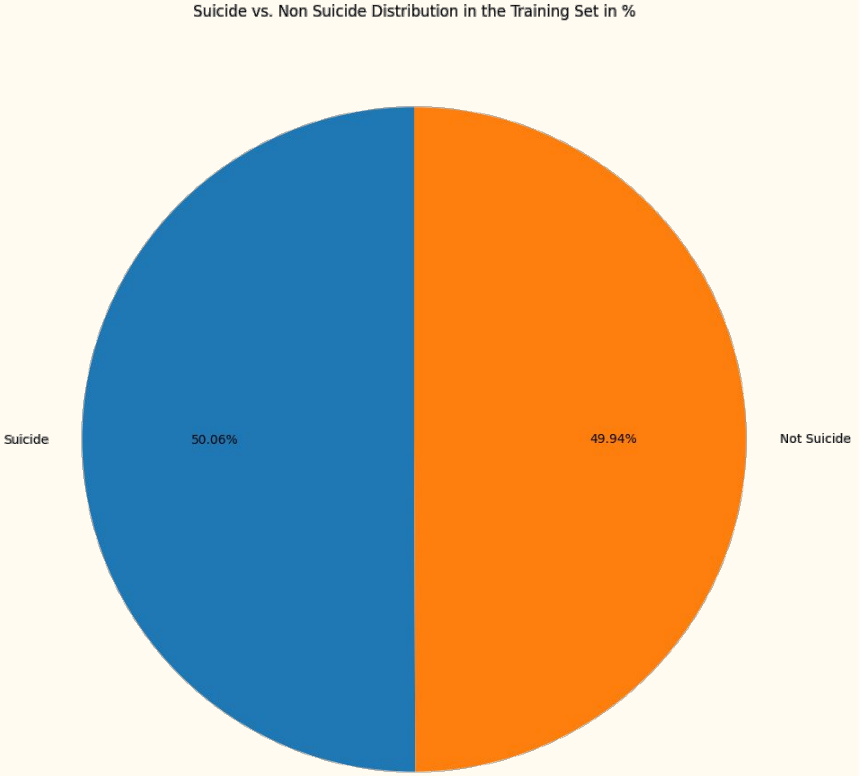
Preprocessing

- Remove noise in the text → neattext
- Example: stop words, email addresses, special characters
- Text transformed into lower cases
- Split into words
- Lemmatization to better group them → WordNetLemmatizer

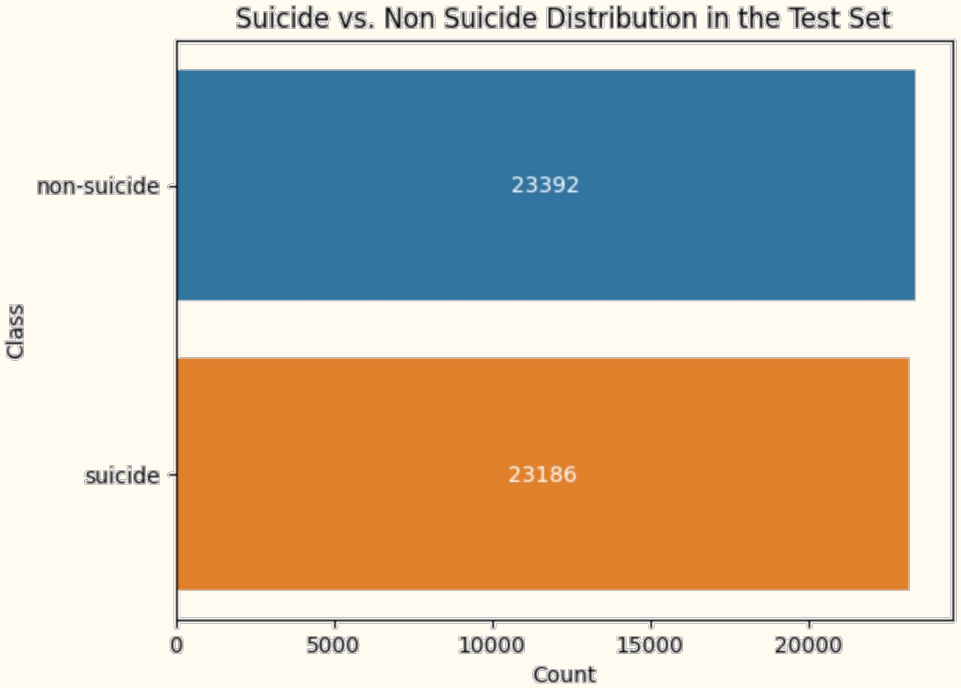


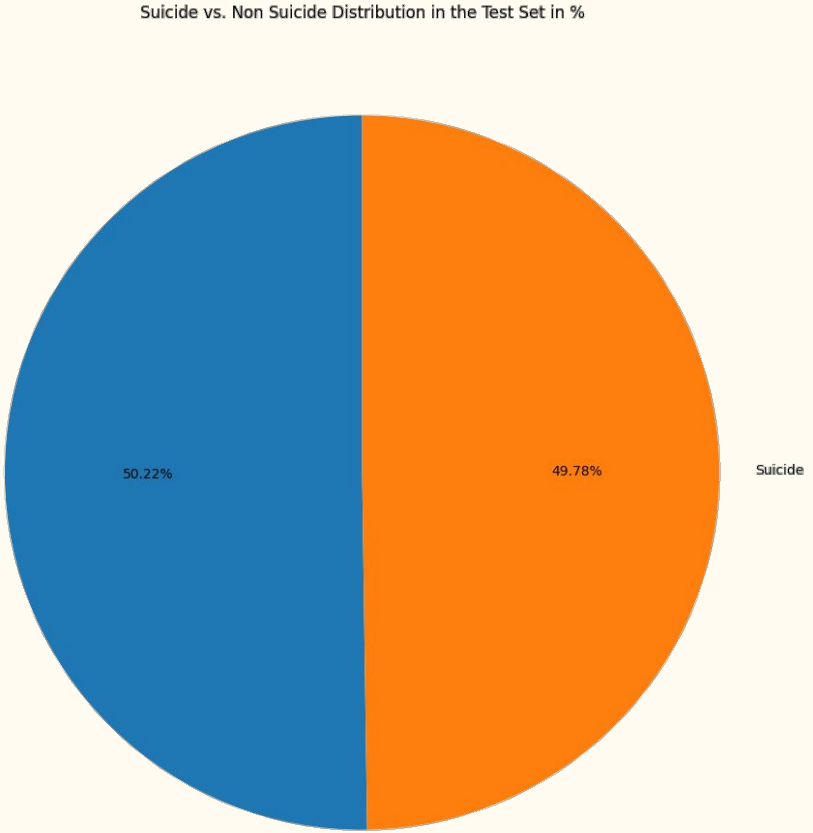
Training Data



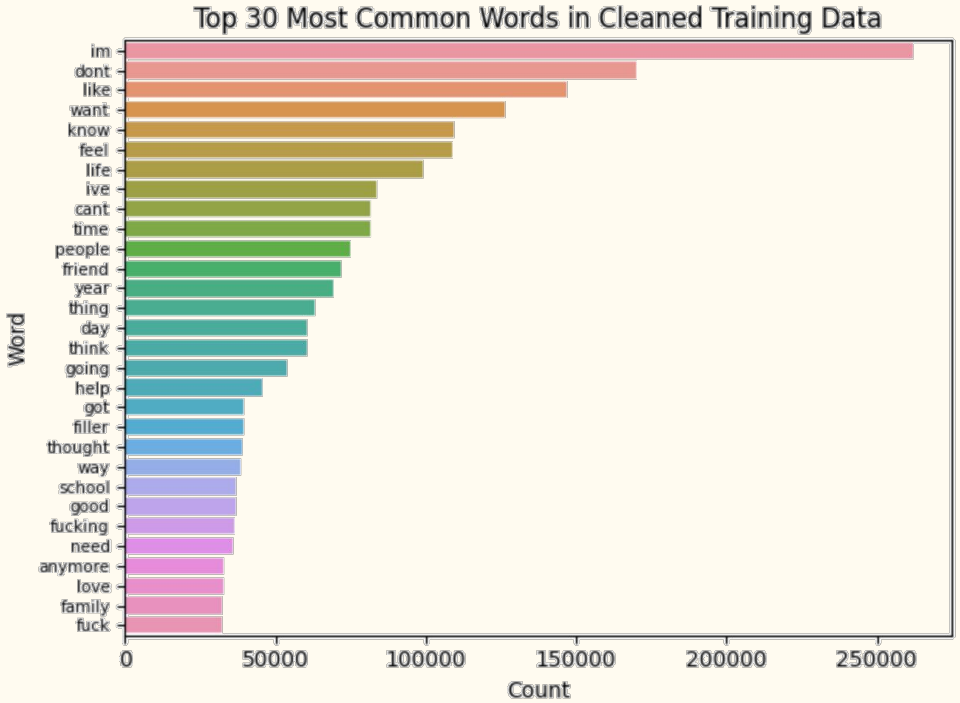


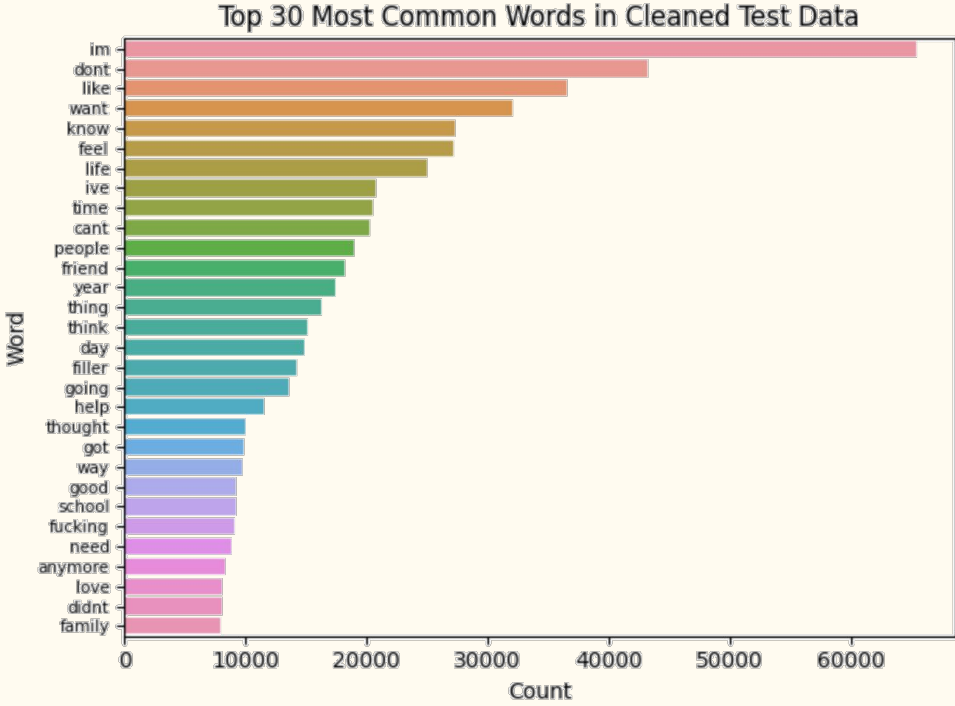
Testing Data





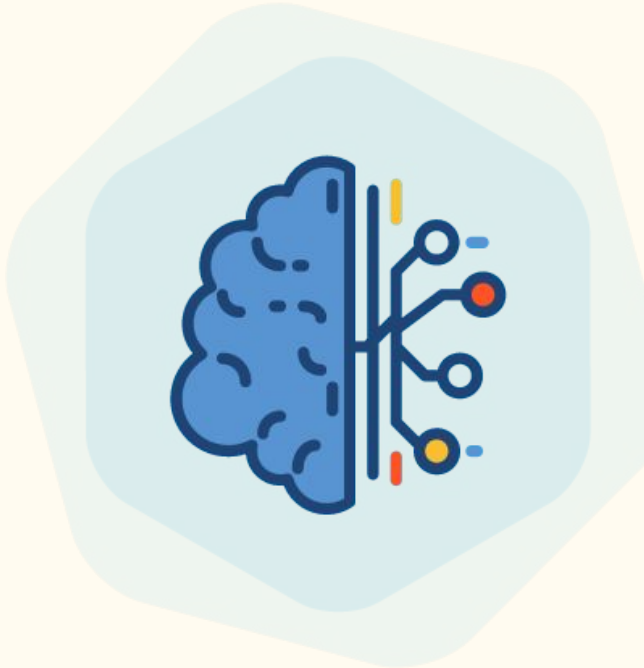
Clean Data


Top 30 Most Common Words in Cleaned Training Data
Word
im
dont
like
want
know
feel
life
ive
cant
time
people
friend
year
thing
day
think
going
help
got
filler
thought
way
school
good
fucking
need
anymore
love
family
fuck
0
50000
100000
150000
200000
250000
Count


Top 30 Most Common Words in Cleaned Test Data
Word
im
dont
like
want
know
feel
life
ive
time
cant
people
friend
year
thing
think
day
filler
going
help
thought
got
way
good
school
fucking
need
anymore
love
didnt
family
0
10000
20000
30000
40000
50000
60000
Count

Naive Bayes Model

- Algorithm → sklearn.naive_bayes.MultinomialNB
- Sentiment analysis
- Low amount of training
- Faster computation time
- Widely used in NLP tasks
- Used with Bag of Words Model (BOW)



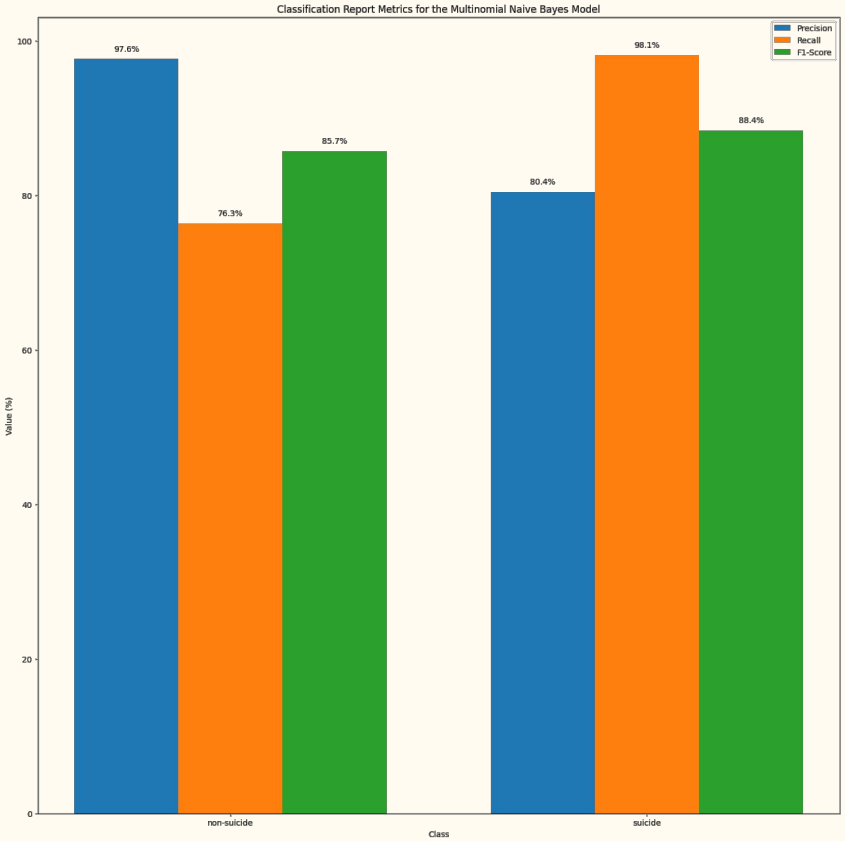
Naive Bayes Model Result

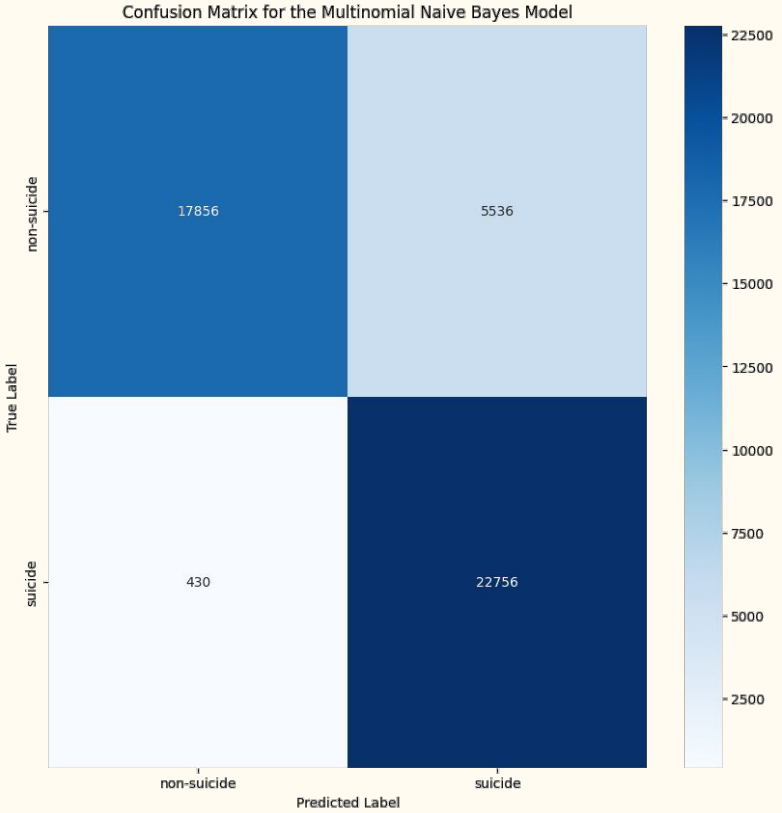
The Classification report for the Multinomial Naive Bayes Model

	precision	recall	f1-score	support
non-suicide	97.65%	76.33%	85.69%	23392
suicide	80.43%	98.15%	88.41%	23186
macro avg	89.04%	87.24%	87.05%	46578
weighted avg	89.08%	87.19%	87.04%	46578

Accuracy Score of the Multinomial Naive Bayes Model: 87.19%

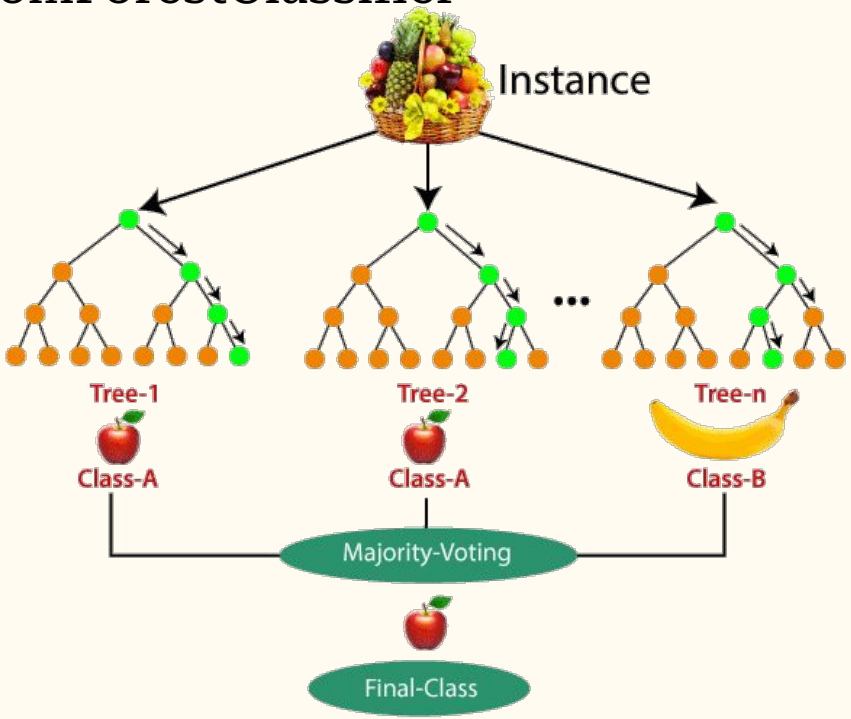
Naive Bayes Model Result





Random Forest Algorithm

- Algorithm → sklearn.ensemble.RandomForestClassifier
- Used for regression and classification
- constructs multiple training trees
  → data classified
- Classified using regression: gives a score of similarity with the given text and the ones previously identified



Random Forest Algorithm Results

Number of estimators: 10

The Classification report for the Random Forest Model

	precision	recall	f1-score	support
non-suicide	86.08%	90.08%	88.03%	23392
suicide	89.50%	85.30%	87.35%	23186
macro avg	87.79%	87.69%	87.69%	46578
weighted avg	87.78%	87.70%	87.69%	46578

Accuracy Score of the Random Forest Model: 87.70%

Number of estimators: 50

The Classification report for the Random Forest Model

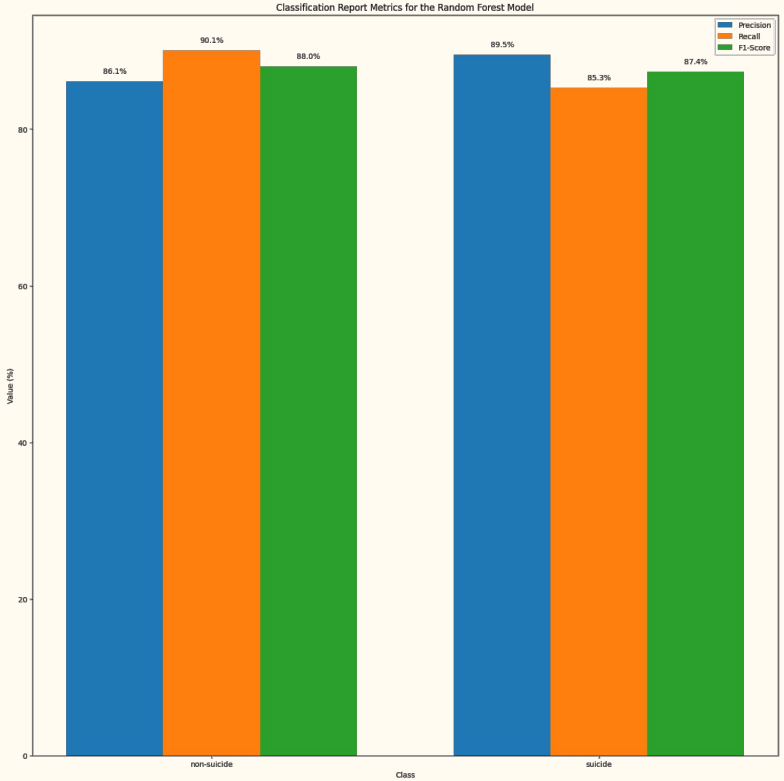
	precision	recall	f1-score	support
non-suicide	90.61%	90.04%	90.32%	23392
suicide	90.01%	90.58%	90.30%	23186
macro avg	90.31%	90.31%	90.31%	46578
weighted avg	90.31%	90.31%	90.31%	46578

Accuracy Score of the Random Forest Model: 90.31%

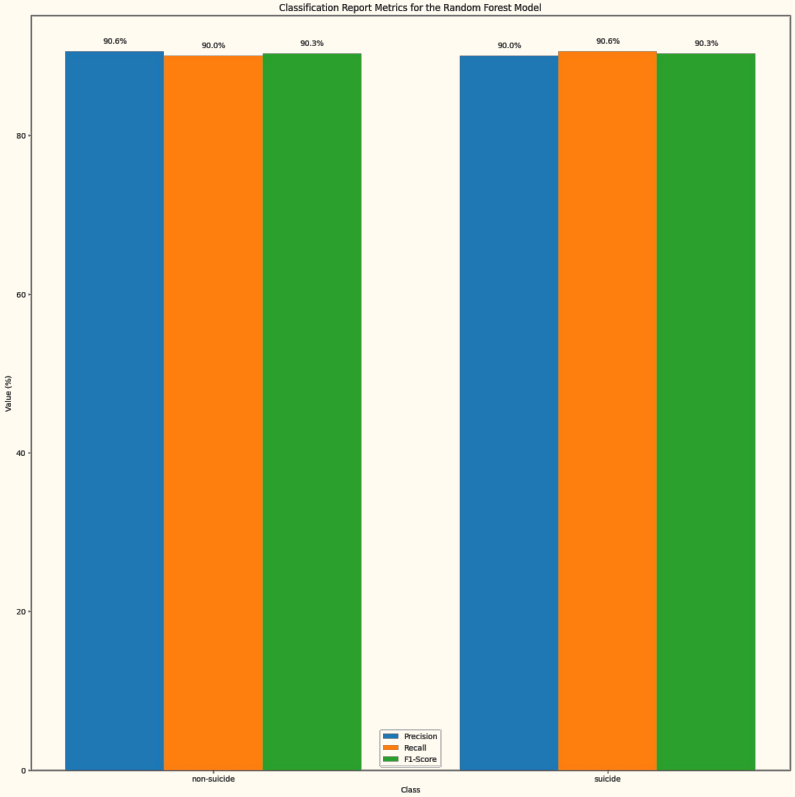
Kept it at 10 because 50 was taking over 30 minutes to run on non M1 macs

Random Forest Algorithm Results - Metrics

Number of estimators: 10

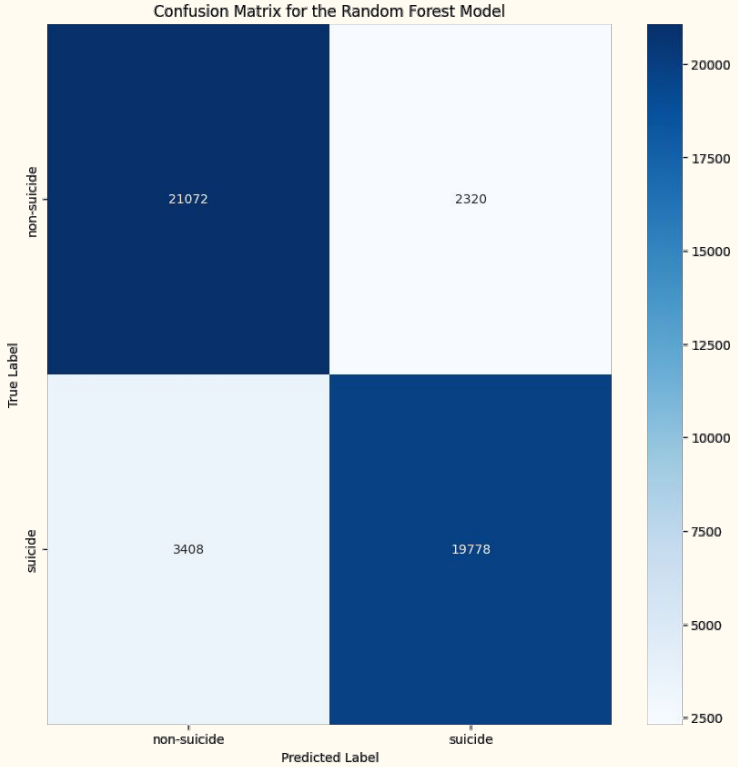


Number of estimators: 50

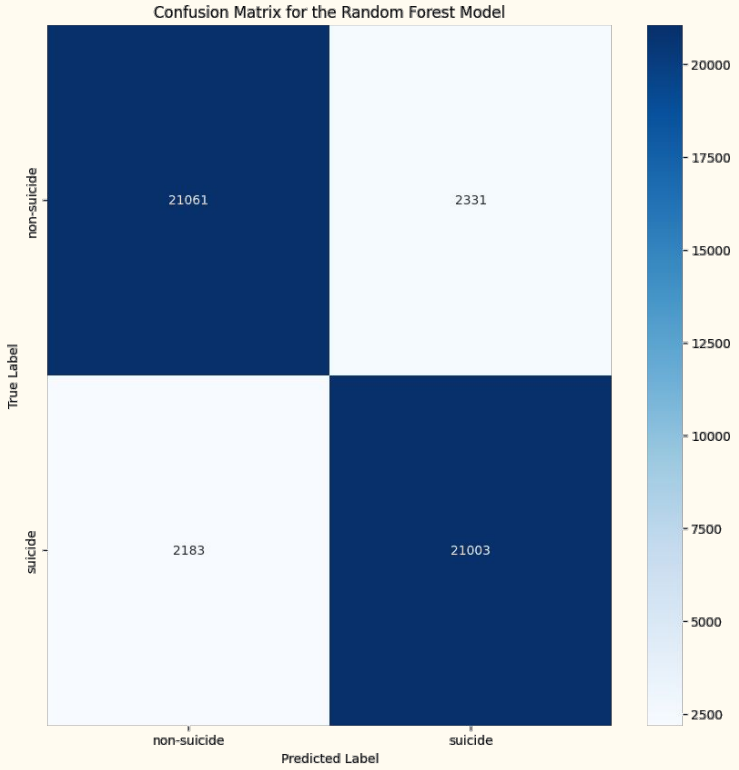


Random Forest Algorithm Results - Confusion Matrix

Number of estimators: 10



Number of estimators: 50



Prediction Function

1. Preprocess the text
2. Transform text with TfidfVectorizer (compute word occurrence)
3. Get predicted probabilities for each class using Random Forest model
4. Get predicted probabilities for each class using Naive Bayes model
5. Calculate the overall predicted probability for each class by taking the mean of the predicted probabilities
6. Highest probability → predicted class

Examples

```
predicted_class("I want to hurt myself.", clf_rf, clf)
✓ 0.0s

100%|██████████| 1/1 [00:00<00:00, 3876.44it/s]
Random Forest Probabilities:
- Suicide: 0.65
- Non-Suicide: 0.35

Naive Bayes Probabilities:
- Suicide: 0.8222551417922808
- Non-Suicide: 0.1777448582077195

The predicted class for the string above is Suicide
```

```
predicted_class("This is getting too hard. I can't handle this anymore.", clf_rf, clf)
✓ 0.0s

100%|██████████| 1/1 [00:00<00:00, 1089.43it/s]
Random Forest Probabilities:
- Suicide: 0.78
- Non-Suicide: 0.22

Naive Bayes Probabilities:
- Suicide: 0.9445937040767407
- Non-Suicide: 0.05540629592325873

The predicted class for the string above is Suicide
```

```
predicted_class("I went for a walk in the park today and enjoyed the beautiful weather.", clf_rf, clf)
✓ 0.0s

100%|██████████| 1/1 [00:00<00:00, 4951.95it/s]
Random Forest Probabilities:
- Suicide: 0.0
- Non-Suicide: 1.0

Naive Bayes Probabilities:
- Suicide: 0.45574028828824464
- Non-Suicide: 0.5442597117117538

The predicted class for the string above is Non-Suicide
```

```
predicted_class("Today I am going for a walk at night alone.", clf_rf2, clf)
✓ 0.0s

100%|██████████| 1/1 [00:00<00:00, 6061.13it/s]
Random Forest Probabilities:
- Suicide: 0.3
- Non-Suicide: 0.7

Naive Bayes Probabilities:
- Suicide: 0.6237290495274881
- Non-Suicide: 0.3762709504725121

The predicted class for the string above is Non-Suicide
```

```
predicted_class("I am really happy right now", clf_rf2, clf)
✓ 0.0s

100%|██████████| 1/1 [00:00<00:00, 1949.03it/s]
Random Forest Probabilities:
- Suicide: 0.1
- Non-Suicide: 0.9

Naive Bayes Probabilities:
- Suicide: 0.6828426869049415
- Non-Suicide: 0.31715731309505907

The predicted class for the string above is Non-Suicide
```