



Suicide **detection** in social media posts

Done by: Kimberley Chen

Why I chose to take up this project?

Worldwide

1. More than 700 000 people die due to suicide every year.
2. Suicide is the fourth leading cause of death in 15-19-year-olds.

Regional



Article from Channel News Asia



Problem Statement

This project aims to apply **machine learning abilities** in particular **text classification techniques** in order to detect suicidal tendencies in social media posts.

Early detection of these risk factors can help in preventing or reducing the number of suicides and even provide help to parties that urgently need it.



Workflow

Data Collection

1. Dataset
2. Data cleaning and data processing

01

02

**EDA
(Exploratory Data Analysis)**

Modelling & Evaluation

03

04

**Limitations and
Recommendations**

The background is a vibrant collage of abstract elements. It includes several large, jagged lightning bolts in yellow and orange, some pointing towards the top and others towards the bottom. There are also smaller blue lightning bolts. The background is decorated with large, soft-edged blobs of color in shades of pink, purple, and blue. Scattered throughout are small circles and dots in black, blue, and pink. On the right side, there is a stylized illustration of a person with long black hair, wearing a pink t-shirt and black pants, standing with their arms raised in a frustrated or overwhelmed pose. The person's face shows a grimace. The overall aesthetic is modern and energetic.

01

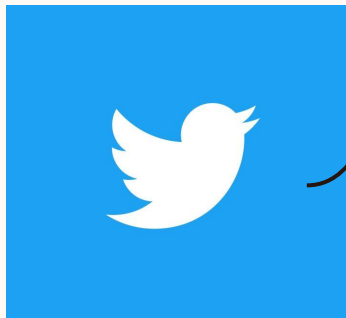
Data Collection

How I obtained my datasets from
Twitter and Reddit.

Introduction to Datasets used:

Reddit

1. From Kaggle
2. Collection of posts from "SuicideWatch" and "depression" subreddits
3. "SuicideWatch" from Dec 16, 2008(creation) till Jan 2, 2021
4. "Depression posts" from Jan 1, 2009, to Jan 2, 2021



Twitter

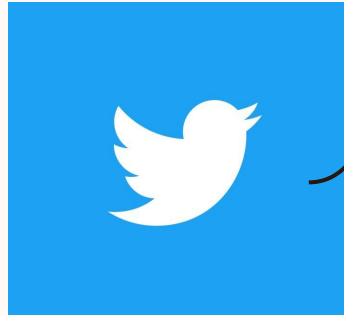
1. Used Snsrape
2. Key words such as "i want to kill myself", "depressed and suicidal", "sadness and hopelessness", "i am worthless", "no one cares"
3. Data was scraped between January to August 2020.



Original Datasets:

Reddit

1. No. of rows: 232,074
2. 2 columns:
 - a. Text, Class



Twitter

1. No. of rows: 3,000
2. 4 columns:
 - a. Datetime, Tweet ID, Text and Username.
3. Manually classified



Finalised Dataset

Total rows: 4000

Suicide (2,000)

Non-suicide (2,000)

2 Columns: Text, Class

Reddit

Twitter

Rows:

- a. Suicide (1,000)
- b. Non-suicide (1,000)

2 Columns:
Text and Class

Rows:

- a. Suicide (1,000)
- b. Non-suicide (1,000)

4 Columns:
Datetime, Tweet
ID, Text and
Username.

Data Cleaning / Data Processing



Data Cleaning

1. Removing of any URLs
2. Removing any mentions (@XXX)
3. Removing any hashtags (#XXX)



Natural Language Processing

1. Remove punctuation
2. Tokenise
3. Remove stopwords
4. Stemming
5. Lemmatization



Vectorization

1. Countvectorizer
2. TF-IDF

02

EDA

Gain deeper insights into the dataset that we are working with



Based on Twitter Data

Weekdays vs Weekends

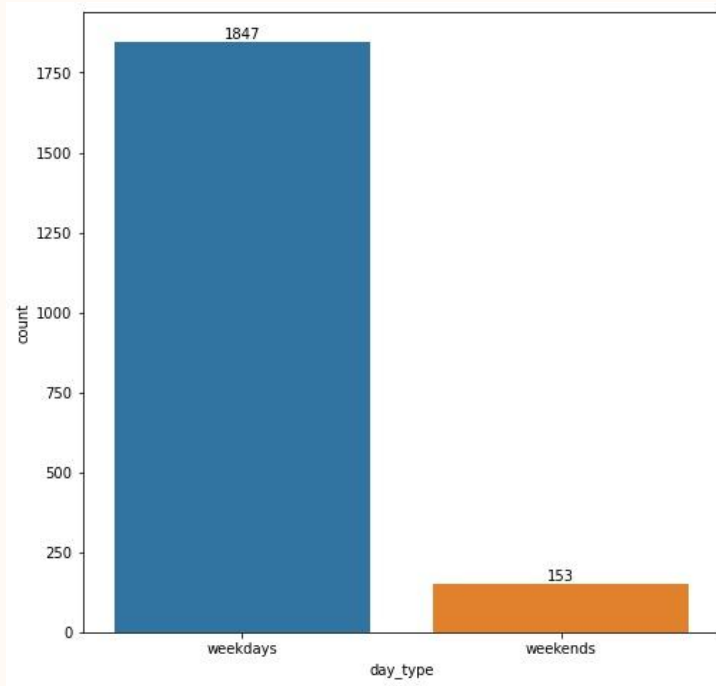


Fig 1: Twitter Weekdays vs Weekends

Day of the week

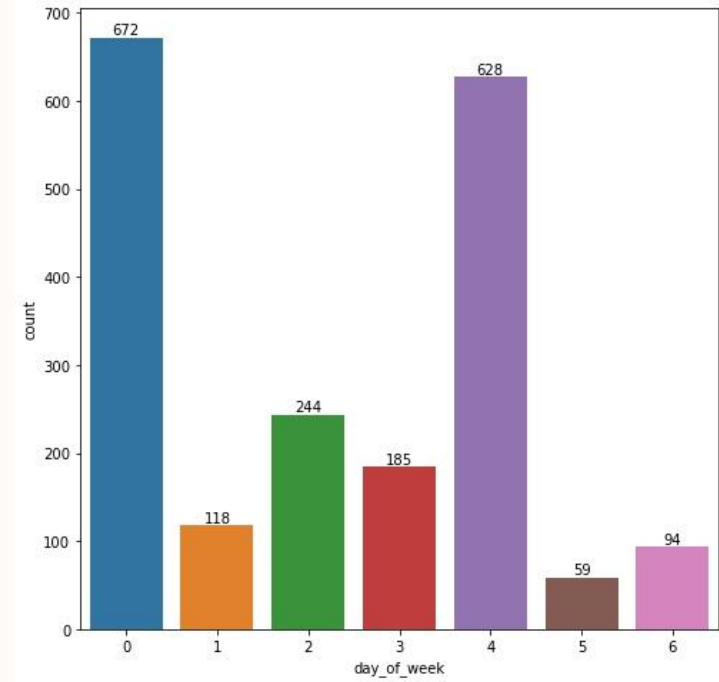


Fig 2: Twitter Day of the Week

Total tweets by part of day

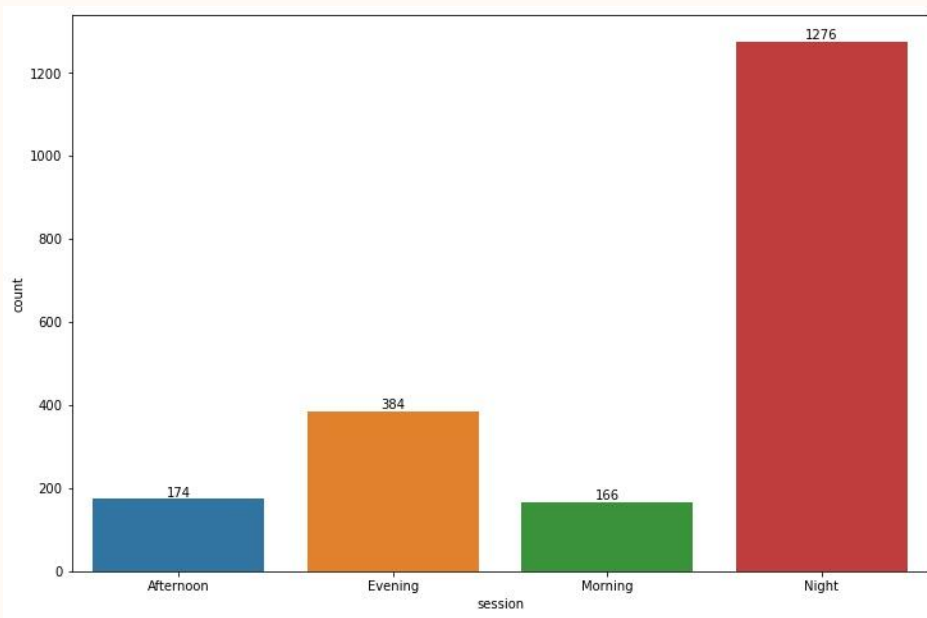


Fig 3: Twitter Total tweets by part of day

Total tweets by hour

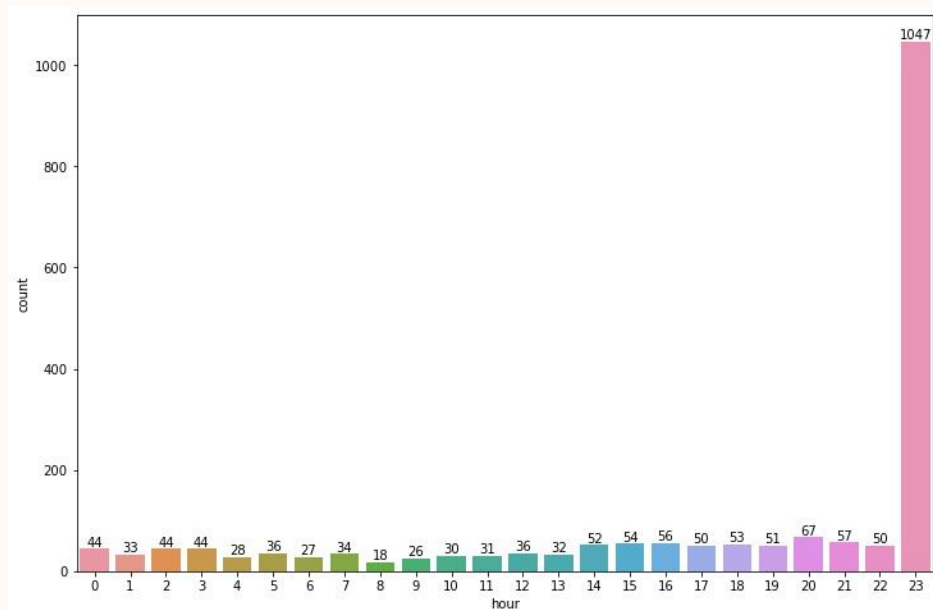


Fig 4: Twitter tweets by hour

Sentiment Analysis

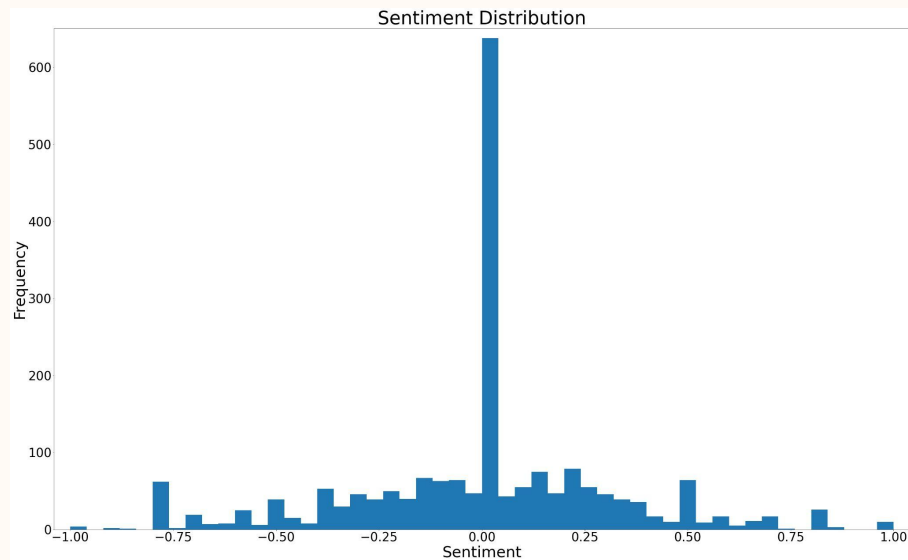


Fig 5: Twitter Sentiment Analysis

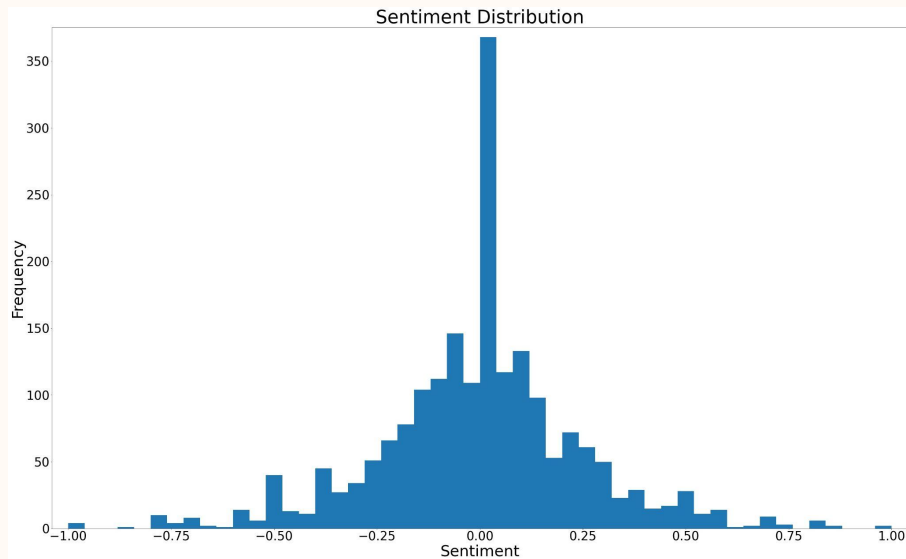


Fig 6: Reddit Sentiment Analysis

Word Count

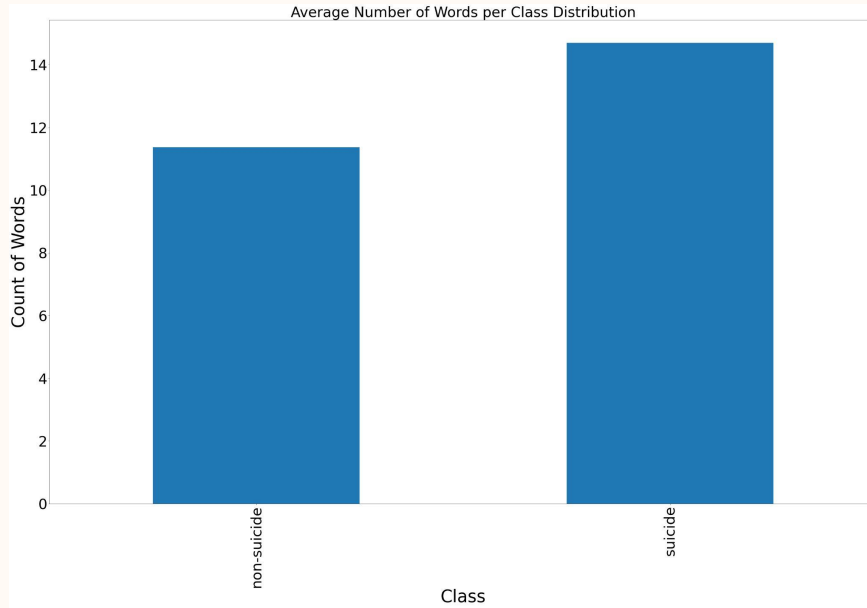


Fig 7: Twitter Word Count

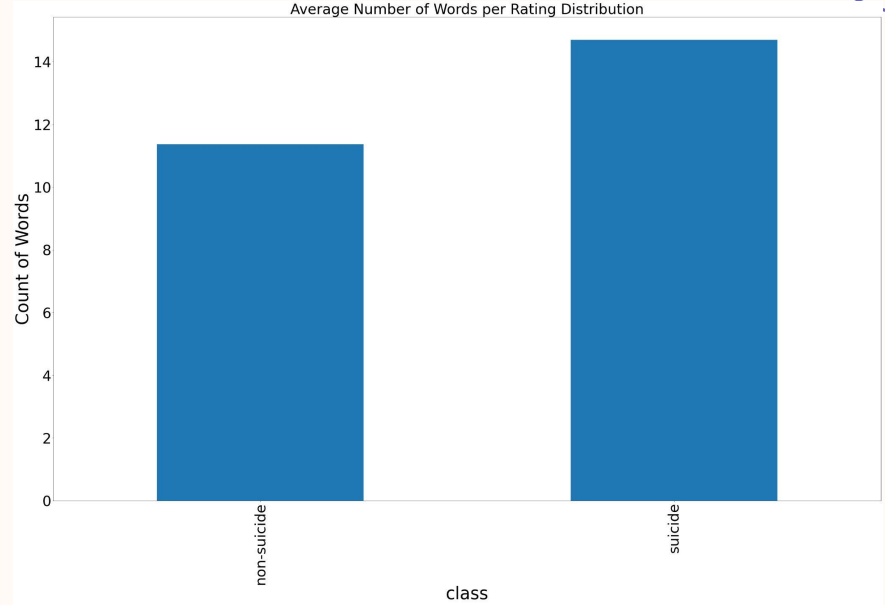


Fig 8: Reddit Word Count

Word Cloud (Twitter)



Fig 9: Twitter (Suicide) Word Cloud



Fig 10: Twitter (Non-Suicide) Word Cloud

Word Cloud (Reddit)



Fig 11: Reddit (Suicide) Word Cloud



Fig 12: Reddit (Non-Suicide) Word Cloud

Combined EDA (2-gram)

(oof, oof)	286
(want, kill)	279
(filler, filler)	168
(im, going)	154
(like, im)	124

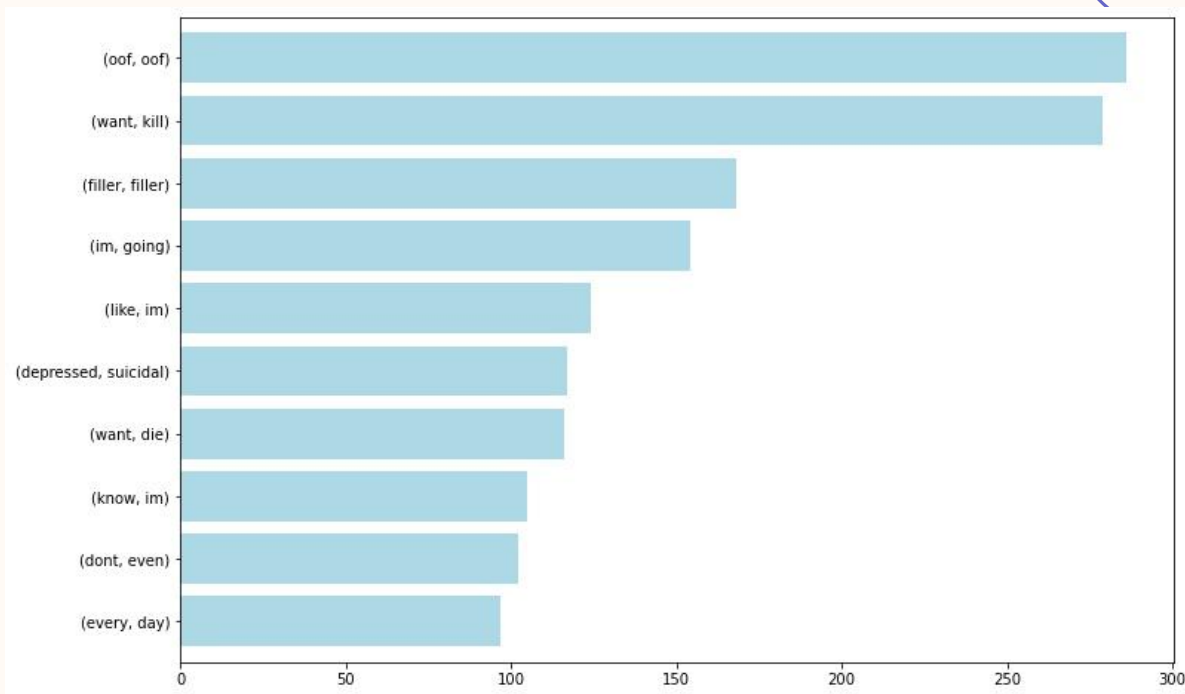


Fig 13: 2-gram

Combined EDA (3-gram)

(im, sorry, im)	32
(dont, know, im)	32
(sorry, im, sorry)	30
(dont, want, live)	26
((alt, alt, alt)	24

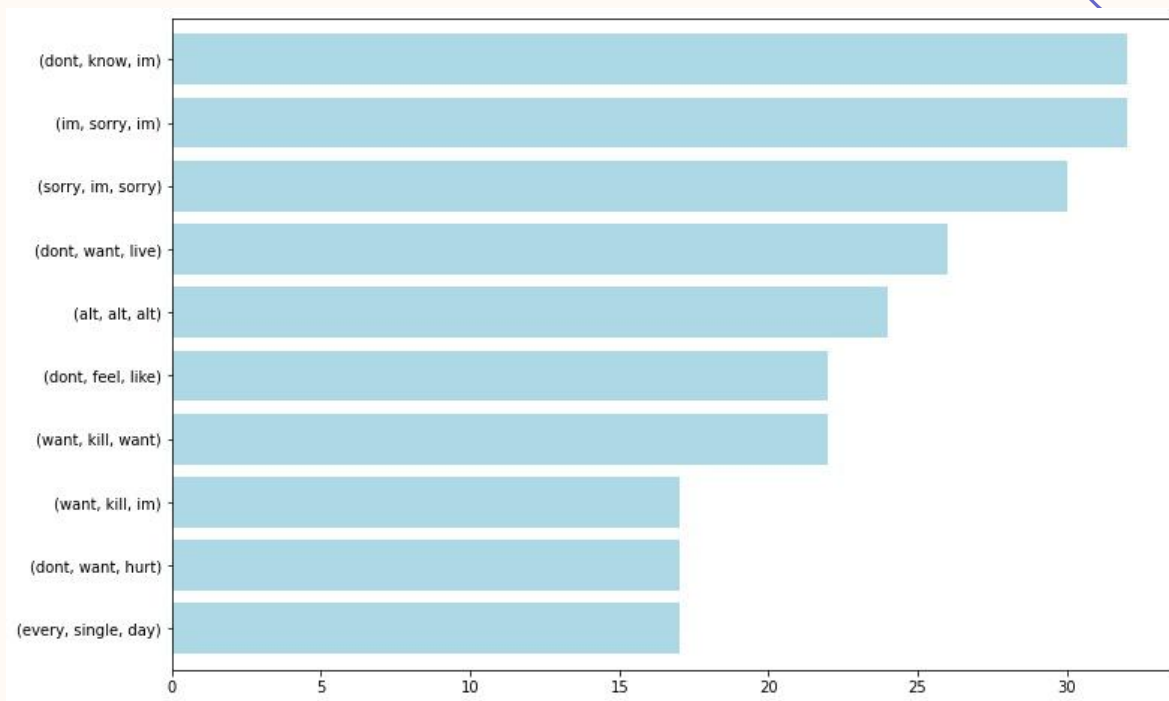


Fig 13: 3-gram

03

- **Modelling & Evaluation**



Count Vectorizer

(Baseline Score: 0.5)

	Accuracy (train)	Accuracy (test)	Sensitivity
Bernoulli NB	0.88	0.75	0.80
Gaussian NB	0.87	0.52	0.75
Logistic Regression	0.96	0.72	0.71
KNN (K-Nearest-Neighbor)	0.74	0.70	0.60
SVM Classifier	0.88	0.70	0.65

Count Vectorizer

(Area under the curve)

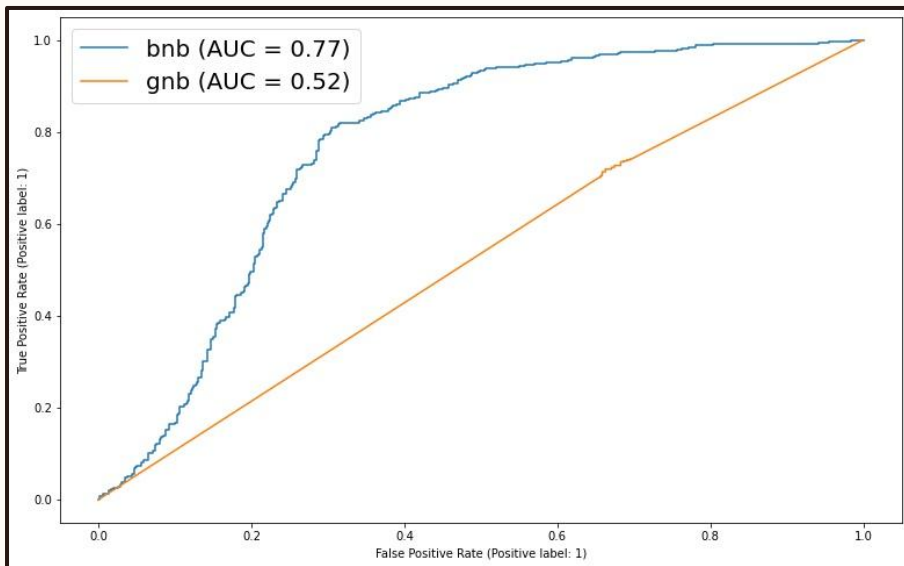


Fig 14: BernoulliNB
GaussianNB

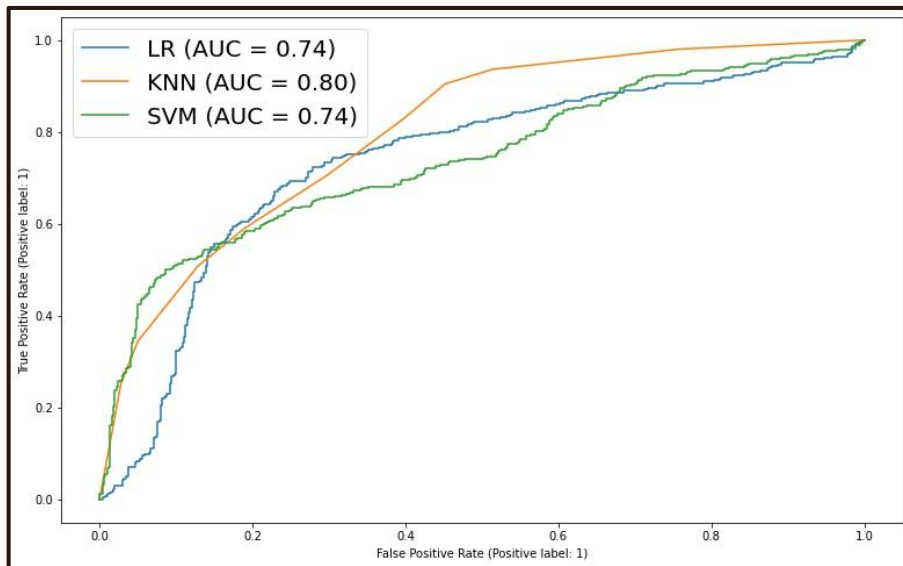


Fig 15: Linear Regression
KNN
SVM

Count Vectorizer

(Baseline Score: 0.5)

	Accuracy	Precision	Sensitivity
Random Forest Classifier	0.94	0.89	0.98
Neural Network	0.64	0.625	0.70
RNN (LSTM)	0.77	0.75	0.86

TF-IDF

(Baseline Score: 0.5)

	Accuracy (train)	Accuracy (test)	Sensitivity
Bernoulli NB	0.88	0.75	0.80
Gaussian NB	0.89	0.52	0.6
Logistic Regression	0.96	0.66	0.67
KNN (K-Nearest-Neighbor)	0.72	0.69	0.90
SVM Classifier	0.96	0.64	0.6

TF-IDF

(Area under the curve)

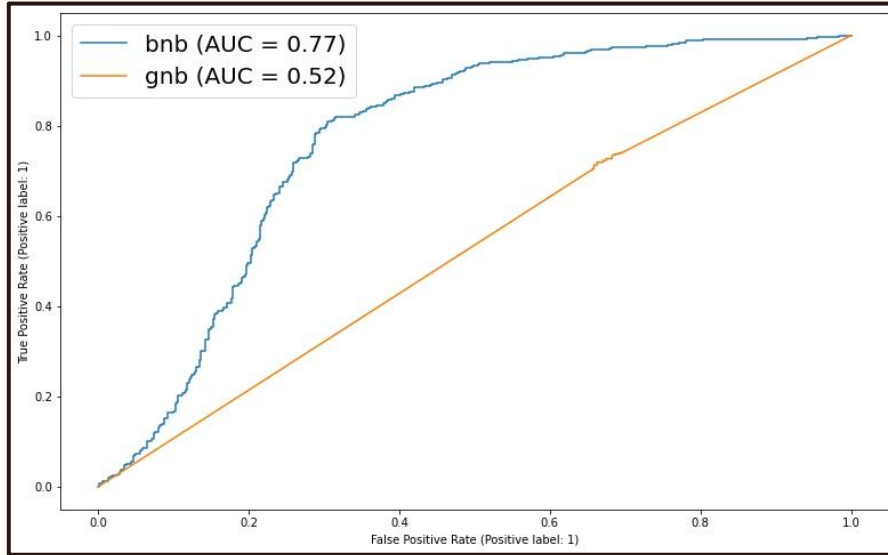


Fig 15: BernoulliNB
GaussianNB

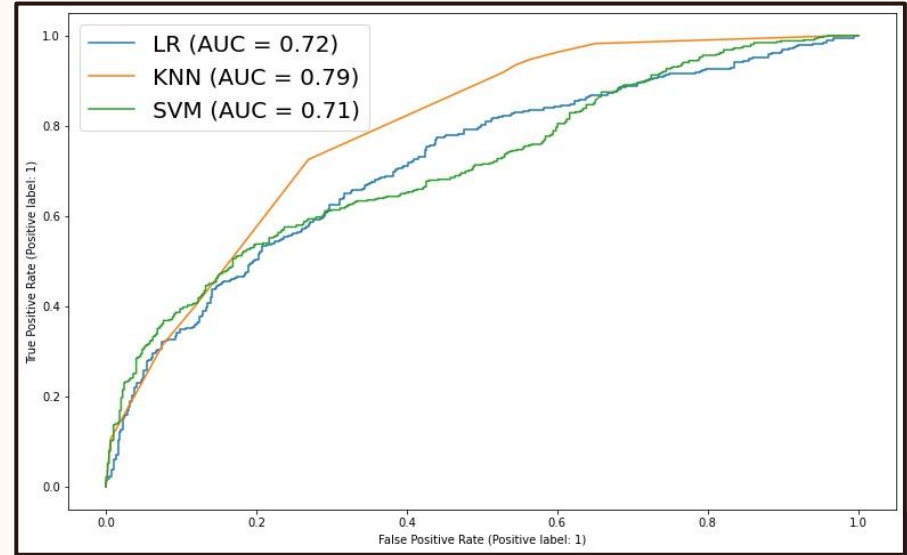


Fig 16: Linear Regression
KNN
SVM

TF-IDF

(Baseline Score: 0.5)

	Accuracy	Precision	Recall
Random Forest Classifier	0.94	0.92	0.98
Neural Network	0.606	0.60	0.71
RNN (LSTM)	0.79	0.78	0.81



The Best Model is:

Random Forest Classifier (TF-IDF)



Flask

(Web Platform)



Suicide Example

Machine Learning App with Flask

Suicide Detector

Enter Your Message Here

I want to die

predict

ML App

Suicide Detector

[Results for Comment](#)

Suicide

Non-Suicide Example

Machine Learning App with Flask

Suicide Detector

Enter Your Message Here

I like corn dog #awesome

predict

ML App

Suicide Detector

Results for Comment

Non-suicide

04

Limitation and Recommendation





Limitations

1. It is not possible to generalise all human behaviour into simple lines of code. Hence this model won't be able to capture every single aspect of human behaviour
2. The data was limited to only Reddit and Twitter, other social media platforms data can also be introduced.

Recommendations

(future projects)

1. Train with a bigger data set
2. Train with different language
3. Image classifier
4. Video classifier



Thank You!

Feel free to ask me any questions :)



CRÉDITOS: este modelo de apresentação foi criado pelo **Slidesgo**, inclui ícones da **Flaticon** e infográficos e imagens da **Freepik**

Por favor, mantenha este slide para atribuição

