FLIP ROBO

MALIGNANT COMMENT CLASSIFIER PROJECT

Submitted by:

Priya Rajput

ACKNOWLEDGMENT

I would like to express my gratitude towards Flip-Robo for providing me this opportunity to show case my talent and also for their constant support and guidance. Also It is indeed a pleasure for me to have worked on this project.

I express my deepest thanks to Miss Sapna Verma, for taking part in useful decision & giving necessary advices and guidance and arranged all facilities to make my project easier. I choose this moment to acknowledge his contribution gratefully.

TABLE OF CONTENTS

INTRODUCTION

Business Problem Framing

The proliferation of social media enables people to express their opinions widely online. However, at the same time, this has resulted in the emergence of conflict and hate, making online environments uninviting for users. Although researchers have found that hate is aproblem across multiple platforms, there is a lack of models for online hate detection.

Online hate, described as abusive language, aggression, cyberbullying, hatefulness and many others has been identified as a major threat on online social media platforms. Social media platforms are the most prominent grounds for such toxic behaviour.

There has been a remarkable increase in the cases of cyberbullying and trolls on various social media platforms. Many celebrities and influences are facing backlashes from people and have to come across hateful and offensive comments. This can take a toll on anyone and affect them mentally leading to depression, mental illness, self-hatred and suicidal thoughts.

Internet comments are bastions of hatred and vitriol. While online anonymity has provided a new outlet for aggression and hate speech, machine learning can be used to fight it. The problem we sought to solve was the tagging of internet comments that are aggressive towards other users. This means that insults to third parties such as celebrities will be tagged as inoffensive, but “u are an idiot” is clearly offensive.

Our goal is to build a prototype of online hate and abuse comment classifier which can used to classify hate and offensive comments so that it can be controlled and restricted from spreading hatred and cyberbullying.

Conceptual Background of the Domain Problem

1. Collect the data.
2. Check whether the project is a regression type or a classification type.
3. Check whether our dataset is balanced or imbalanced. If it is an imbalanced one, we will apply sampling techniques to balance the dataset.
4. Model building and find the accuracy of the model.
5. Build a model with good accuracy and also go for hyper parameter tuning.

Motivation

Our main objective of doing this project is to build a model to predict malignant comment We are going to predict by using Machine Learning algorithms.

The sample data is provided to us from our client database. I have worked on this on the bases of client requirements and followed all the steps till model deployment

Analytical Problem Framing

Mathematical/ Analytical Modelling of the Problem

Various Classification analysis techniques were used to build Classification models to determine whether an input Message content is benign or malignant. Machine Learning Algorithms such as Multinomial Naïve Bayes and Complement Naïve Bayes were employed which are based on the Bayes Theorem: P(message is malignant | message content) = P(message content | malignant). P(malignant) / P(message content) The probability of message being Malignant, knowing that Message Content has occurred could be calculated. Event of "Message Content" represents the evidence and "Message is Malignant", the hypothesis to be approved. The theorem runs on the assumption that all predictors/features are independent and the presence of one would not affect the other. The approach to classify a comment as malignant would depend on training data labelled as various categories of malignant messages and benign messages.

Data Set Description

The data set contains the training set, which has approximately 1,59,000 samples and the test set which contains nearly 1,53,000samples. All the data samples contain 8 fields which includes 'Id', 'Comments', 'Malignant', 'Highly malignant', 'Rude', 'Threat', 'Abuse' and 'Loathe'.

The label can be either 0 or 1, where 0denotes a NO while 1 denotes a YES. There are various comments which have multiple labels. The first attribute is a unique ID associated with each comment.

The data set includes:

- **Malignant:** It is the Label column, which includes values 0 and 1, denoting if the comment is malignant or not.
- **Highly Malignant:** It denotes comments that are highly malignant and hurtful.
- **Rude:** It denotes comments that are very rude and offensive.
- **Threat:** It contains indication of the comments that are giving any threat to someone.

- **Abuse:** It is for comments that are abusive in nature.
- **Loathe:** It describes the comments which are hateful and loathing in nature.
- **ID:**It includes unique Ids associated with each comment text given.
- **Comment text:** This column contains the comments extracted from various social media platforms.

2. Data Analysis

Data Inputs- Logic- Output Relationships The comment tokens so vectorised using TfidVectorizer are input and classified as benign(0) or malignant(1) as output by classification models. • State the set of assumptions (if any) related to the problem under consideration The comment content made available in Train and Test Dataset is assumed to be written in English Language in the standard Greco-Roman script. This is so that the Stopword package and WordNetLemmatizer can be effectively used.

Data Inputs- Logic- Output Relationships

After collecting the data, you need to build a machine learning model. Before model building do all data pre-processing steps. Try different models with different hyper parameters and select the best model.Follow the complete life cycle of data science. Include all the steps like

1. Data Cleaning
2. Exploratory Data Analysis
3. Data Pre-processing
4. Model Building
5. Model Evaluation
6. Selecting the best model

DATA SOURCES AND THEIR FORMATS

Data Source: The read_csv function of the pandas library is used to read the content of a CSV file into the python environment as a pandas DataFrame. The function can read the files from the OS by using proper path to the file.

Data description: Pandas describe() is used to view some basic statistical details like percentile, mean, std etc. of a data frame or a series of numeric values. When this method is applied to a series of string, it returns a different output which is shown below.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings0
warnings.filterwarnings('ignore')
```

```
df=pd.read_csv(r'C:\Users\Admin\Desktop\train.csv')
df.head()
```

	id	comment_text	malignant	highly_malignant	rude	threat	abuse	loathe
0	0000997932d777bf	Explanation\nWhy the edits made under my usern...	0	0	0	0	0	0
1	000103f0d9cfb60f	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0
2	000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0
3	0001b41b1c6bb37e	"\nMore\nI can't make any real suggestions on ...	0	0	0	0	0	0
4	0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0

Data Pre-processing Done

For the data pre-processing step, I checked through the dataframe for missing values and renamed values that needed a better meaningful name.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 159571 entries, 0 to 159570
Data columns (total 8 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   id                159571 non-null  object
 1   comment_text      159571 non-null  object
 2   malignant         159571 non-null  int64
 3   highly_malignant  159571 non-null  int64
 4   rude              159571 non-null  int64
 5   threat            159571 non-null  int64
 6   abuse             159571 non-null  int64
 7   loathe            159571 non-null  int64
dtypes: int64(6), object(2)
memory usage: 9.7+ MB
```

```
df_test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 153164 entries, 0 to 153163
Data columns (total 2 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   id            153164 non-null  object
 1   comment_text  153164 non-null  object
dtypes: object(2)
```

Data Inputs- Logic- Output Relationships

EDA was performed by creating valuable insights using various visualization libraries.

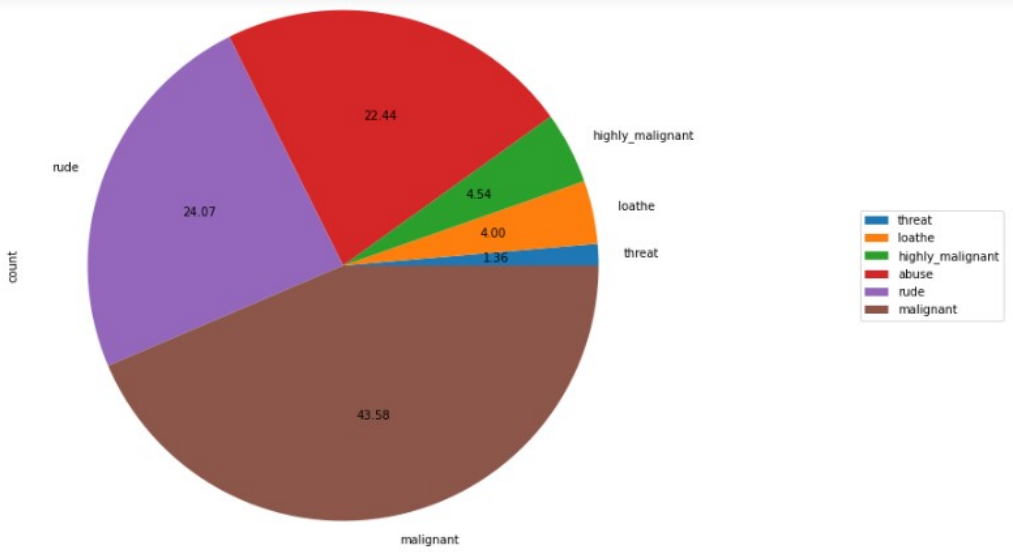
Hardware and Software Requirements and Tools Used

Hardware required:
Processor: core i3
RAM: 8 GB
Software required:
Anaconda 3- language used Python 3
Microsoft Excel

Exploratory Data Analysis

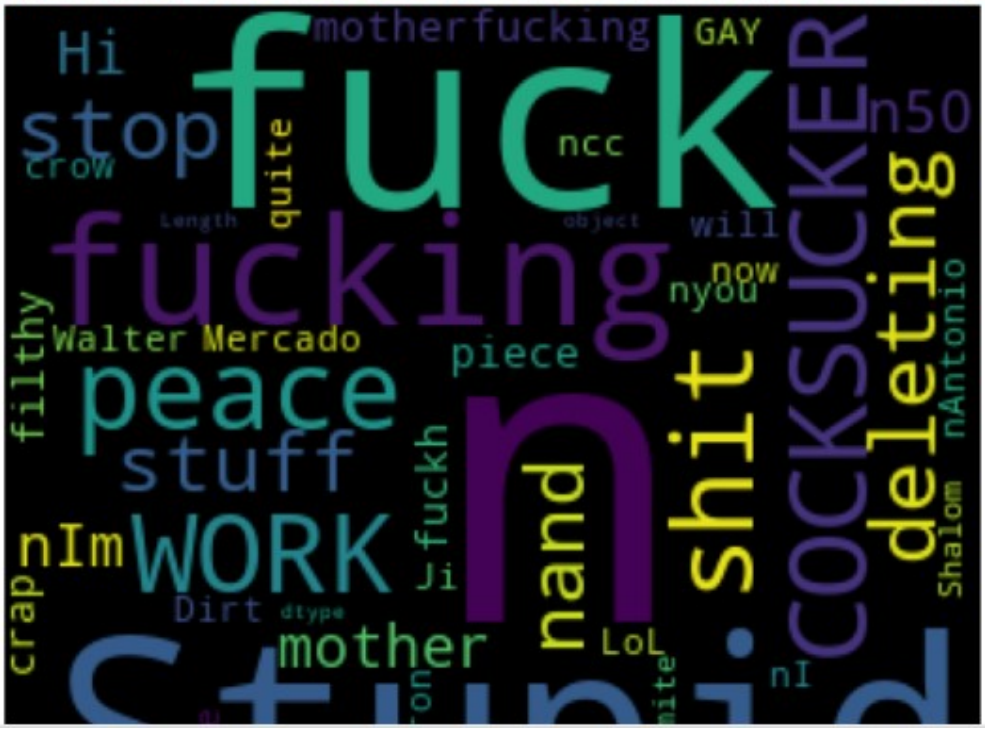
Let us explore our data and visualize

```
# Label distribution comments in pie chart

distribution = df[column].sum()\
                            .to_frame()\
                            .rename(columns={0: 'count'})\
                            .sort_values('count')

distribution.plot.pie(y = 'count', title = 'Label distribution over comments', autopct='%.2f', figsize = (10, 10))\
                            .legend(loc='center left', bbox_to_anchor=(1.3, 0.5))
```

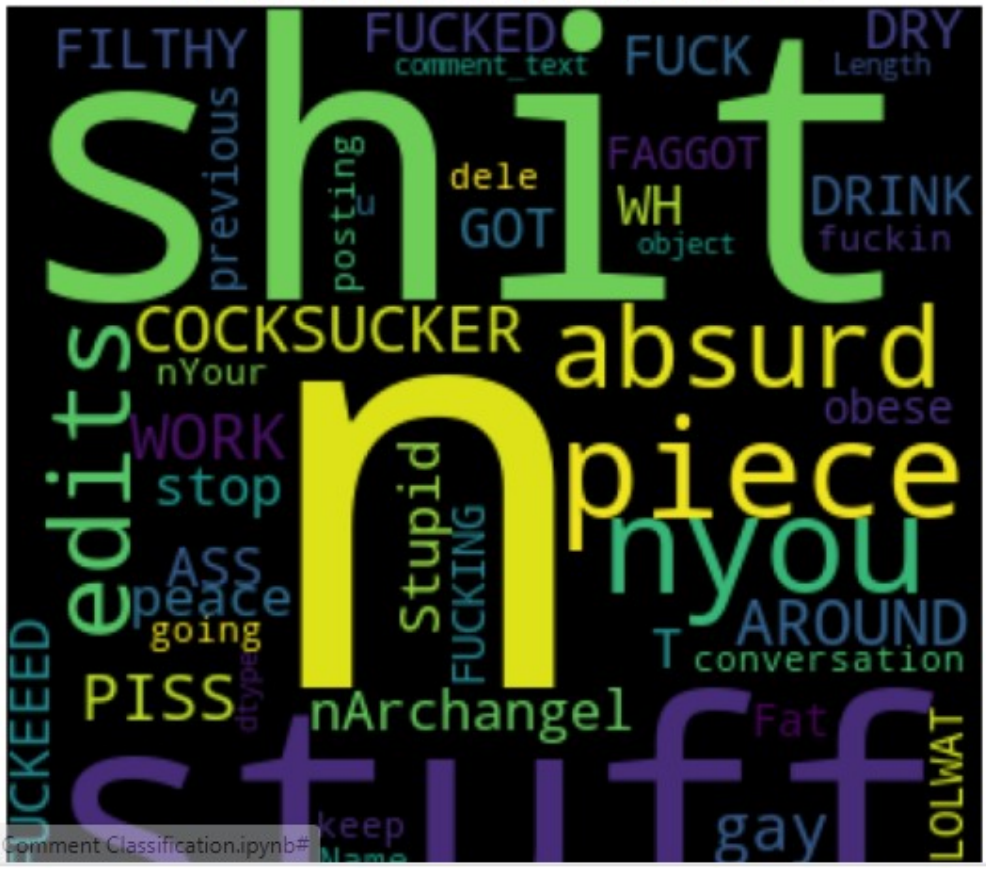



```
malignant=df[(df['malignant']==1)]

wordcloud=WordCloud(height=300,width=300,max_words=300).generate(str(malignant['comment_text']))
plt.figure(figsize=(8,8))
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.title(label='WORDS TAGGED AS MALIGNANT',fontdict={'fontsize':30, 'fontweight':30, 'color':'red'})
plt.show()
```

WORDS TAGGED AS HIGHLY-MALIGNANT

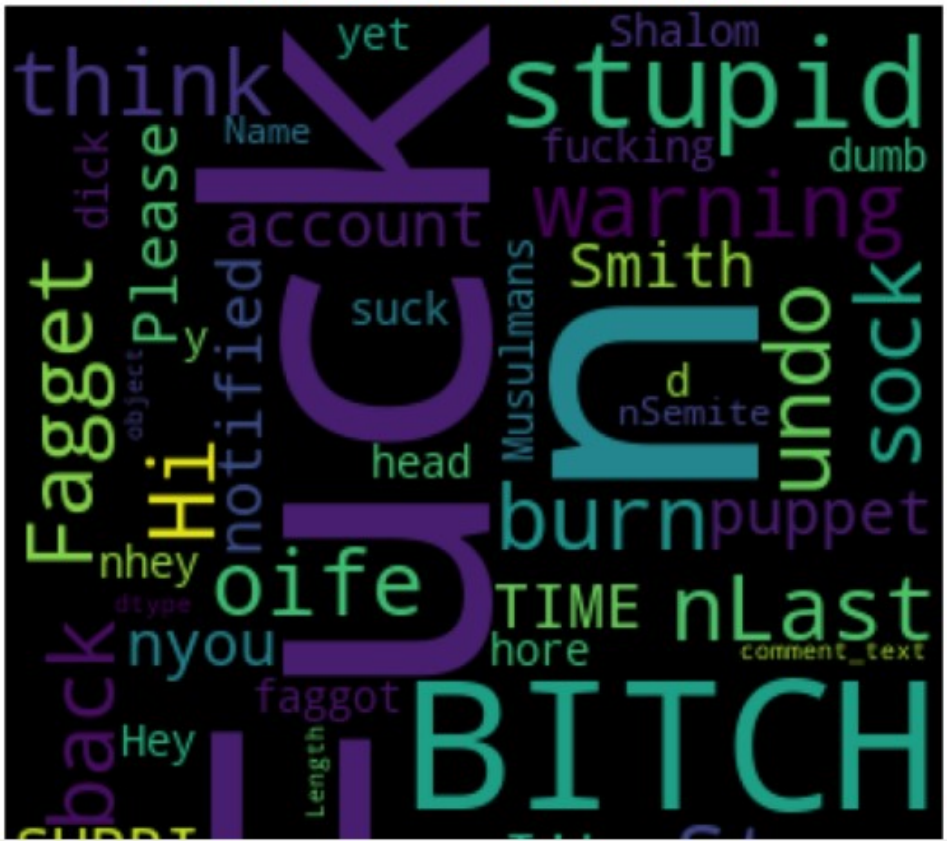


WORDS TAGGED AS RUDE



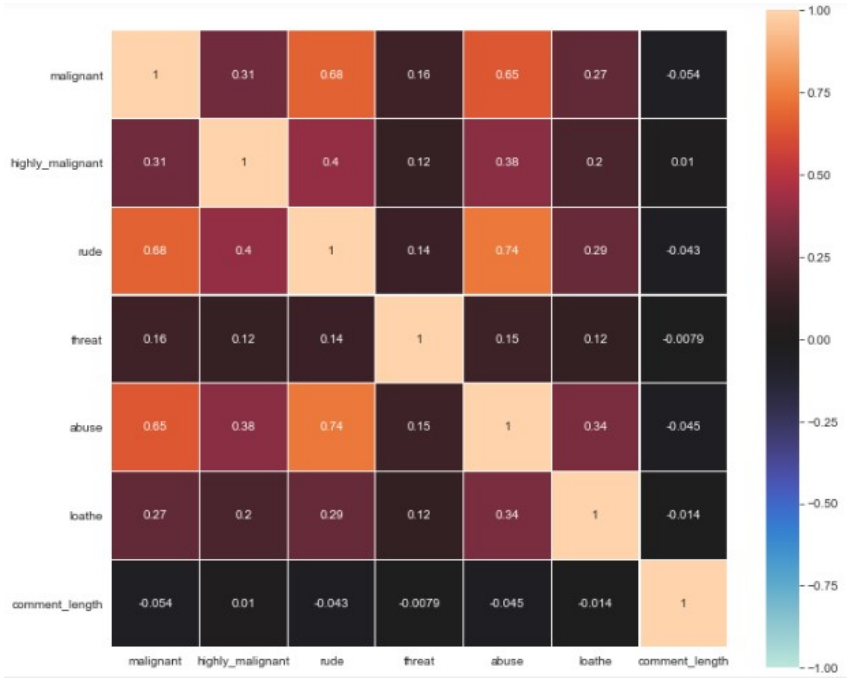
WORDS TAGGED AS ABUSE

WORDS TAGGED AS THREAT



Heatmap

This is the heatmap where I have checked the correlation between the data and also got to know that there are columns or independent variables which are 0% correlated with the target variable.



Interpretation of the Results

In the visualization part, we have seen how my data looks like using heatmap. In the pre-processing part, we cleaned the data using wordnet,wordcloud

In the modeling part, we have designed our model using algorithms like Random Forest Classifier. The accuracy score, confusion_matrix, classification_report are achieved for each model.

Conclusion

• Key Findings and Conclusions of the Study

The Model has 93.76% accuracy. But since the dataset was highly imbalanced that is not the best metric for measuring its efficiency. However, there is a need to strike a balance between precision and recall and have low false positives, which unnecessarily consume time and low false negatives which means only very few toxic comments deceive the model. F1 score provides a nuanced way to catch positive results without harming the usefulness of the model.

• Learning Outcomes of the Study in respect of Data Science

The various data pre-processing and feature engineering steps in the project lent cognizance to various efficient methods for processing textual data. The NLTK suite is very useful in pre-processing text-based data and building classification models.

• Limitations of this work and Scope for Future Work

The models were trained on a highly imbalanced dataset where the total malignant comments formed only 10% of the entire available data, which seriously affected the training and accuracy of the models. By training the models on more diverse data sets, longer comments, and a more balanced dataset, more accurate and efficient classification models can be built.