FLIP ROBO

MALIGNANT COMMENT CLASSIFIER

Submitted by:

Ibrahim Abdul Shukoor

Internship – 30

ACKNOWLEDGMENT

I want to express my gratitude to my mentor, Mr. Kashif Mohd, who always supported me and helped me develop my talents. I want to thank Flip Robo Technologies for providing me with a platform to learn, comprehend, and carry out various projects like this one. Finally, I would want to express my sincere gratitude to Data Trained for teaching me cutting-edge Python, Statistics, and machine learning approaches.

INTRODUCTION

As the social media space grows wider, thousands of new users can be seen logging into their accounts and leaving comments on posts or reacting to the comments left by other users. This gives space to the users to express their views publicly and on occasions their views may not be worded in the nicest way possible. The aim of today's project is to build a model which can predict if the comments left by users are *malignant* or not.

I. Business Problem Framing

The objective of the project is to predict if the comment let by a user can be considered malignant or not.

Given a group of sentences or paragraphs, used as a comment by a user in an online platform, and specified into— malignant, highly-malignant, rude, threat, abuse or loathe, must be classified into *malignant* or *not malignant* with either approximate probabilities or discrete values (0/1).

II. Conceptual Background of the Domain Problem

Following are the study's primary goals:

a. To understand the data present in the dataset
b. To clean the data present in the dataset (if applicable)
c. To visualize different traits that can have an impact on the target variable.
d. To create machine learning models that can predict the malignancy
e. To select the ideal model and use it with the test data.

III. Review of Literature

Machine Learning: With the use of machine learning (ML), which is a form of artificial intelligence (AI), software programs can predict outcomes more accurately without having to be explicitly instructed to do so. In order to forecast new output values, machine learning algorithms use historical data as input. The kind of data that data scientists wish to predict determines the kind of algorithm they use.

The three categories of machine learning algorithms are:

i. Supervised Learning: In supervised learning, data scientists describe the variables they want the algorithm to look for connections between and provide the algorithms with labelled training data. The algorithm's input and output are both described.

ii. Unsupervised learning: Algorithms trained on unlabelled data are used in this sort of machine learning. The algorithm searches through data sets in search of any significant relationships. Both the input data that algorithms use to train and the predictions or suggestions they produce are predefined.

iii. Data scientists generally employ reinforcement learning to instruct a computer to carry out a multi-step procedure for which there are set rules. An algorithm is programmed by data scientists to fulfil a goal, and they provide it with positive or negative feedback as it determines how to do so. However, the algorithm typically chooses the course of action on its own.

IV. Motivation for the Problem Undertaken

Every individual needs a platform deserving of love and positivity, if the malignant comments can be positively identified, then social media could be a safe place for people to express their opinions without getting attacked for it.

Analytical Problem Framing

I. Mathematical/ Analytical Modelling of the Problem

There are several methods for data analysis, but the two that are used most frequently are as follows:

- Supervised learning, which includes classification and regression models.
- Unsupervised learning, which includes association rules and clustering methods

Classification Model: Classification models are used to look at how different variables relate to one another. When determining which independent factors have the most impact on dependent variables, classification models are frequently utilized to gather crucial information.

II. Data Sources and their formats

The sample data is provided by FlipRobo's client database.

Format: Comma Separated Values (CSV).

Data Types:

- int64 – 6.
- object – 1

III. Data Pre-processing Done

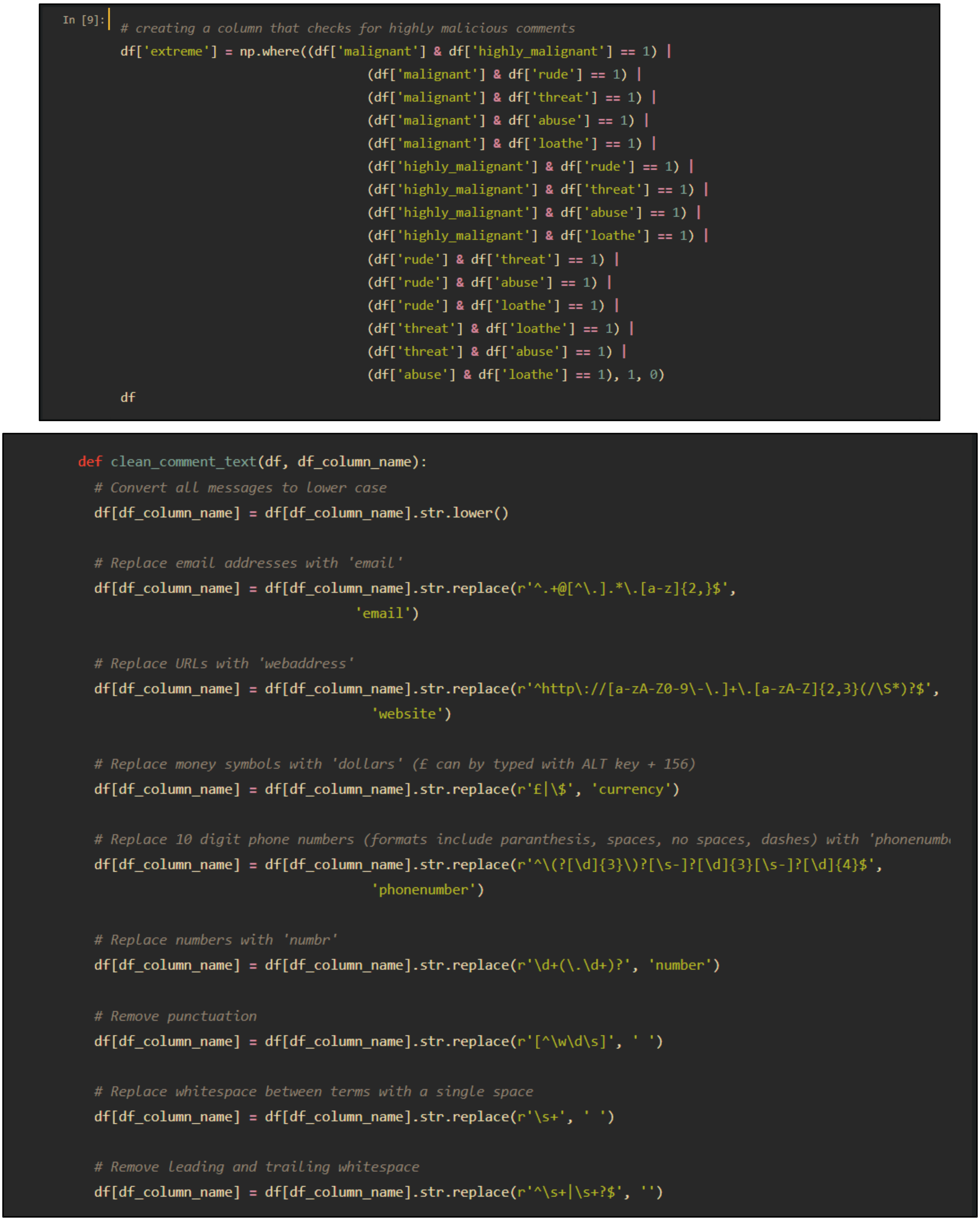
The following actions were taken during the data pre-processing:

i. The "ID" column was dropped due to the column containing various unique values.

ii. Created a column called "Extreme", which containing binary values (0/1). Which checks if the comment checks if any of the two columns (i.e; threat and loathe or malignant and threat) are 1.

iii. Comments were cleaned:

   a. Emails were replaced with "email"
   b. Websites were replaced with "website"
   c. Currencies were replaced with "currency"
   d. Phone numbers were replaced with "phonenumber"
   e. Cleaned additional things such as trailing and leading white spaces, removes words that are made of two letters, removed stop words.



```
In [9]: # creating a column that checks for highly malicious comments
        df['extreme'] = np.where((df['malignant'] & df['highly_malignant'] == 1) |
                                 (df['malignant'] & df['rude'] == 1) |
                                 (df['malignant'] & df['threat'] == 1) |
                                 (df['malignant'] & df['abuse'] == 1) |
                                 (df['malignant'] & df['loathe'] == 1) |
                                 (df['highly_malignant'] & df['rude'] == 1) |
                                 (df['highly_malignant'] & df['threat'] == 1) |
                                 (df['highly_malignant'] & df['abuse'] == 1) |
                                 (df['highly_malignant'] & df['loathe'] == 1) |
                                 (df['rude'] & df['threat'] == 1) |
                                 (df['rude'] & df['abuse'] == 1) |
                                 (df['rude'] & df['loathe'] == 1) |
                                 (df['threat'] & df['loathe'] == 1) |
                                 (df['threat'] & df['abuse'] == 1) |
                                 (df['abuse'] & df['loathe'] == 1), 1, 0)
        df
```

```
def clean_comment_text(df, df_column_name):
  # Convert all messages to lower case
  df[df_column_name] = df[df_column_name].str.lower()

  # Replace email addresses with 'email'
  df[df_column_name] = df[df_column_name].str.replace(r'^.+@[^\.].*\.[a-z]{2,}$',
                                  'email')

  # Replace URLs with 'webaddress'
  df[df_column_name] = df[df_column_name].str.replace(r'^http\://[a-zA-Z0-9\-\.]+\.[a-zA-Z]{2,3}(/\S*)?$',
                                    'website')

  # Replace money symbols with 'dollars' (£ can by typed with ALT key + 156)
  df[df_column_name] = df[df_column_name].str.replace(r'£|\$', 'currency')

  # Replace 10 digit phone numbers (formats include paranthesis, spaces, no spaces, dashes) with 'phonenumb
  df[df_column_name] = df[df_column_name].str.replace(r'^\(?[\d]{3}\)?[\s-]?[\d]{3}[\s-]?[\d]{4}$',
                                    'phonenumber')

  # Replace numbers with 'numbr'
  df[df_column_name] = df[df_column_name].str.replace(r'\d+(\.\d+)?', 'number')

  # Remove punctuation
  df[df_column_name] = df[df_column_name].str.replace(r'[^\w\d\s]', ' ')

  # Replace whitespace between terms with a single space
  df[df_column_name] = df[df_column_name].str.replace(r'\s+', ' ')

  # Remove leading and trailing whitespace
  df[df_column_name] = df[df_column_name].str.replace(r'^\s+|\s+?$', '')
```

Figure 1: Data Pre-processing

I. Hardware/Software and Tools

i. Jupyter Notebook
ii. NumPy
iii. Pandas
iv. Matplotlib
v. Seaborn
vi. nltk
vii. string

Model/s Development and Evaluation

I. Testing of Identified Approaches (Algorithms)

Five algorithms altogether were executed.

i. K Neighbours Classifier
ii. Multinomial Naïve Bayes
iii. Linear Support Vector Classifier
iv. Logistic Regression
v. Decision Tree Classifier

II. Run and evaluate selected models

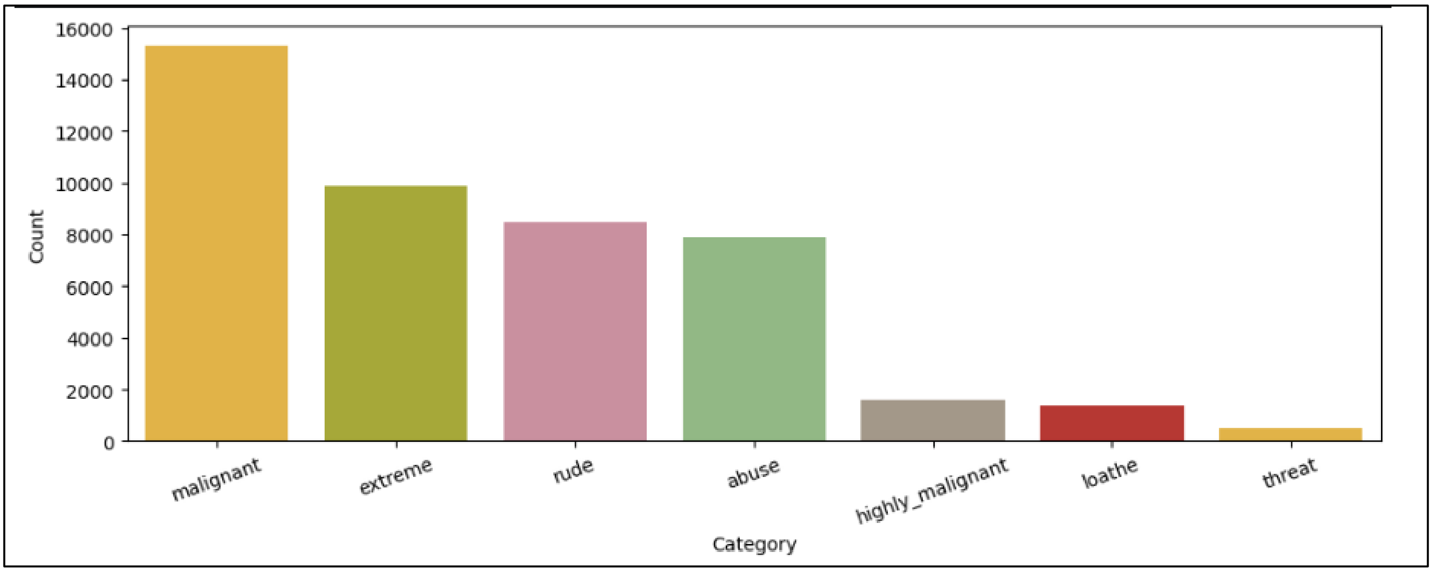
i. K Neighbours Classifier
ii. Multinomial Naïve Bayes **(BEST SCORE)**
iii. Linear Support Vector Classifier
iv. Logistic Regression
v. Decision Tree Classifier

III. Key Metrics for success in solving problem under consideration

The measures utilized to assess the model's effectiveness included the F1 score and accuracy.

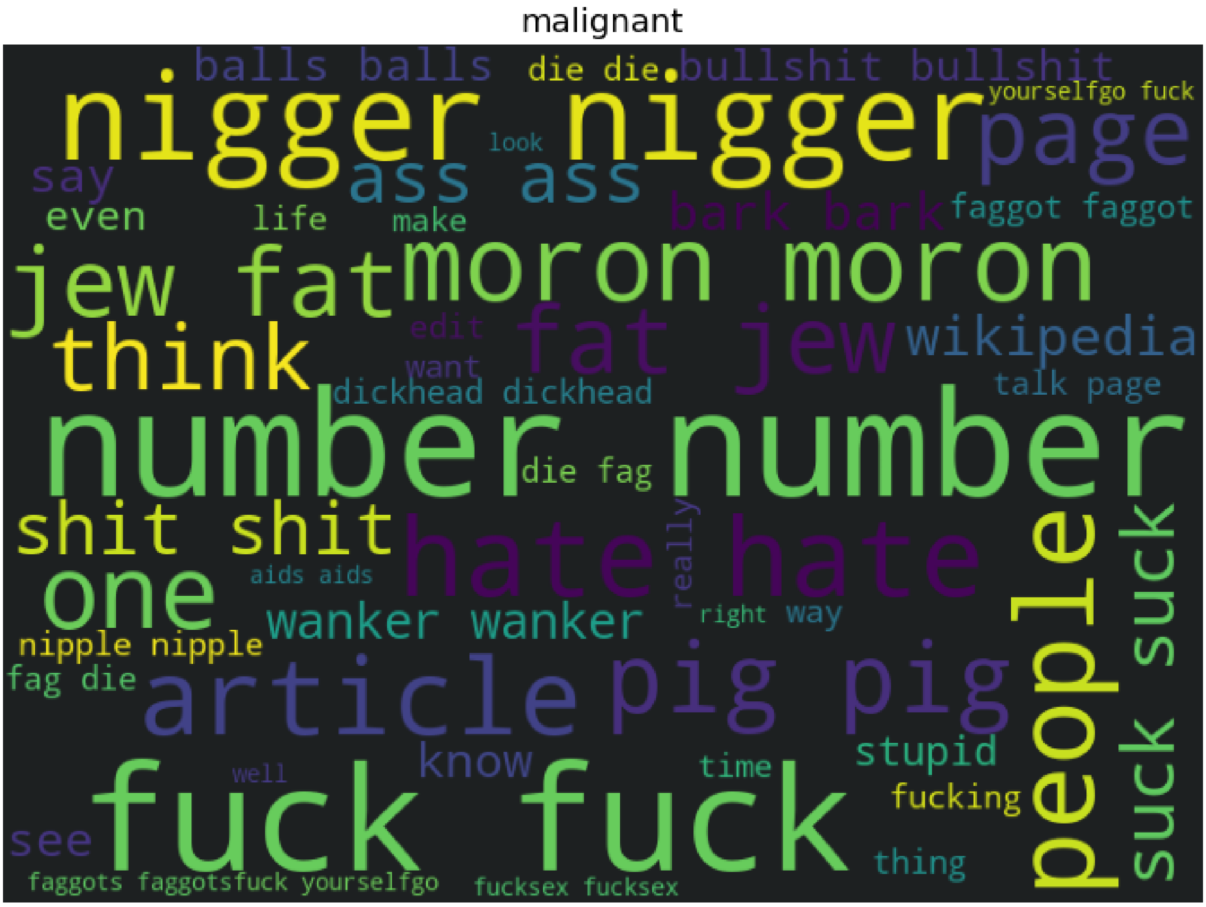
IV. Visualizations

i. Plotting the categories and the count

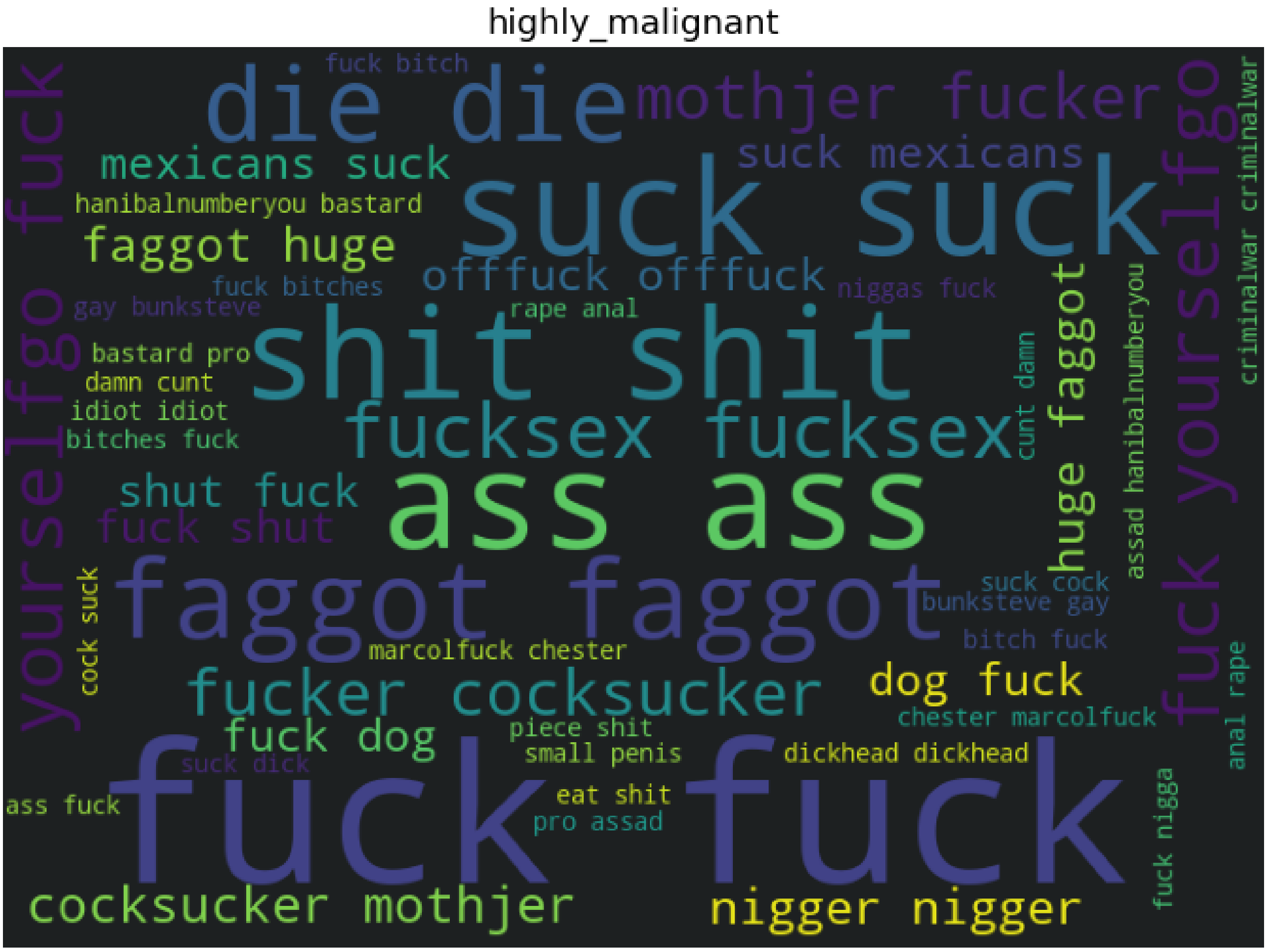


ii. Plotted the word clouds of the comments that matched the categories

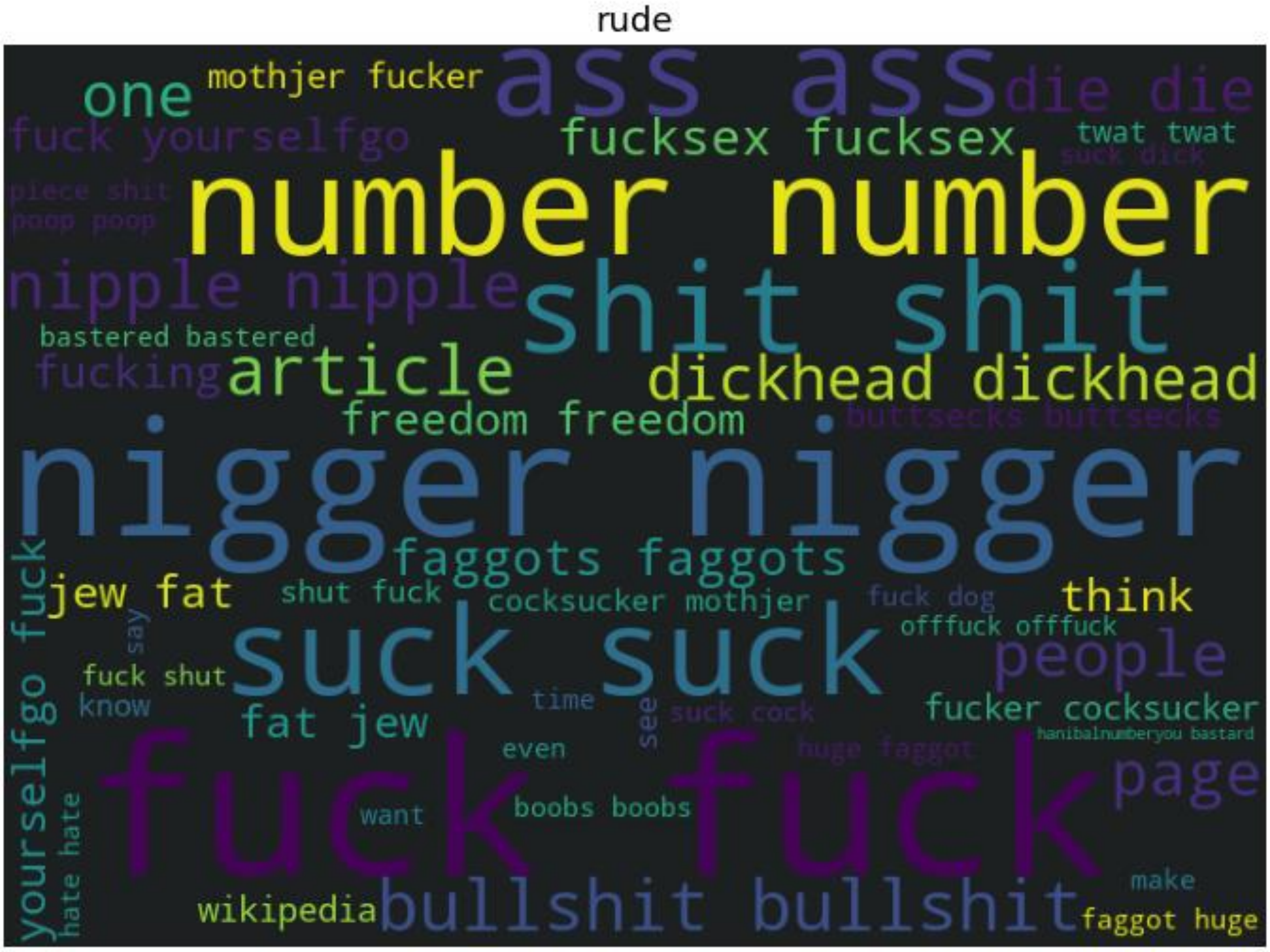
a. malignant



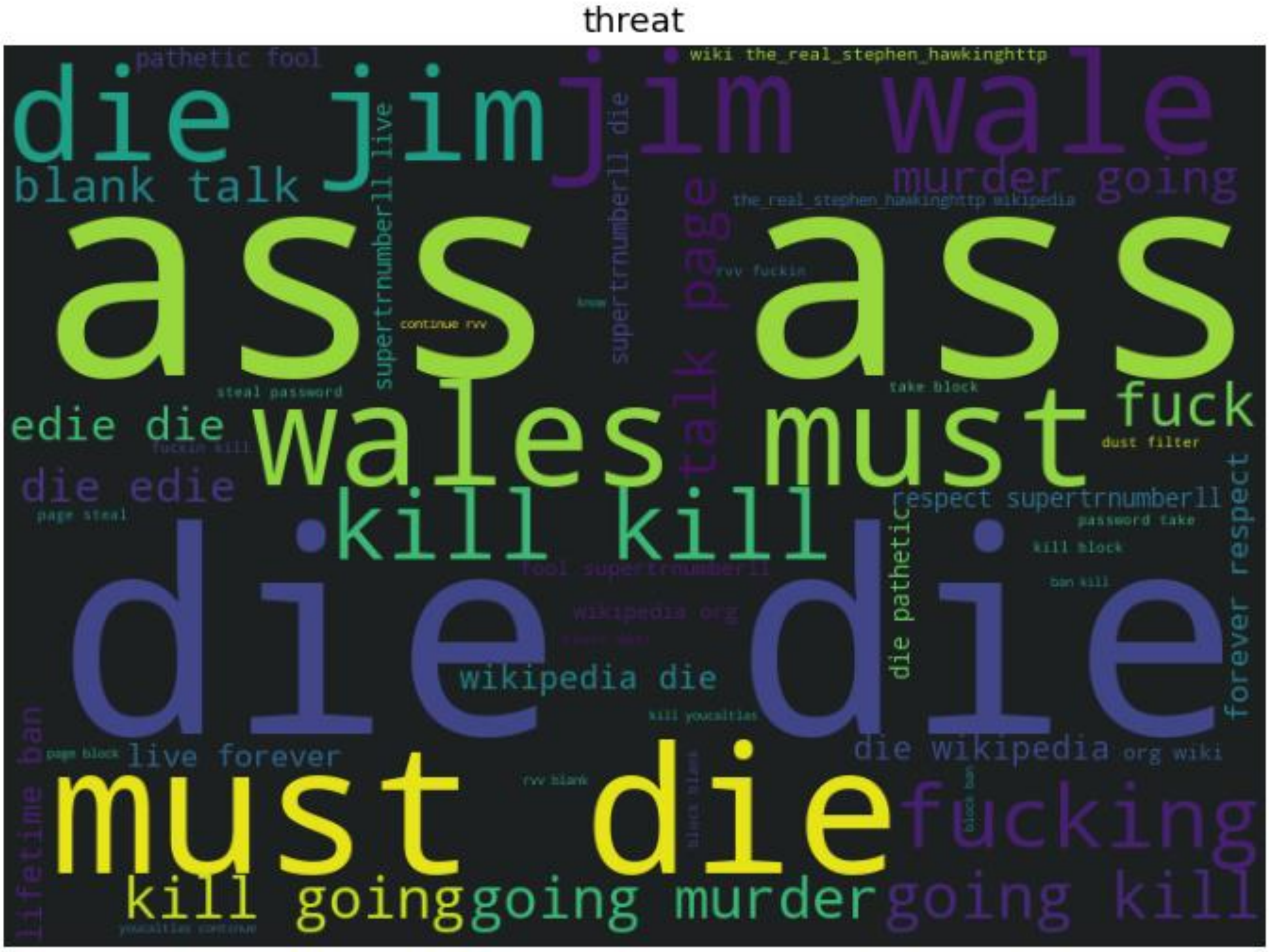
b. Highly malignant


highly_malignant
fuck bitch
die die
mothjer fucker
fuck yourselfgo
criminalwar criminalwar
mexicans suck
suck mexicans
hanibalnumberyou bastard
suck suck
faggot huge
fuck bitches
offfuck offfuck
niggas fuck
gay bunksteve
rape anal
bastard pro
damn cunt
idiot idiot
bitches fuck
shit shit
cunt damn
huge faggot
assad hanibalnumberyou
fucksex fucksex
shut fuck
fuck shut
ass ass
fuck yourselfgo
cock suck
faggot faggot
suck cock
bunksteve gay
bitch fuck
marcolfuck chester
anal rape
fucker cocksucker
dog fuck
chester marcolfuck
fuck dog
piece shit
small penis
suck dick
dickhead dickhead
ass fuck
fuck fuck
eat shit
pro assad
fuck nigga
cocksucker mothjer
nigger nigger

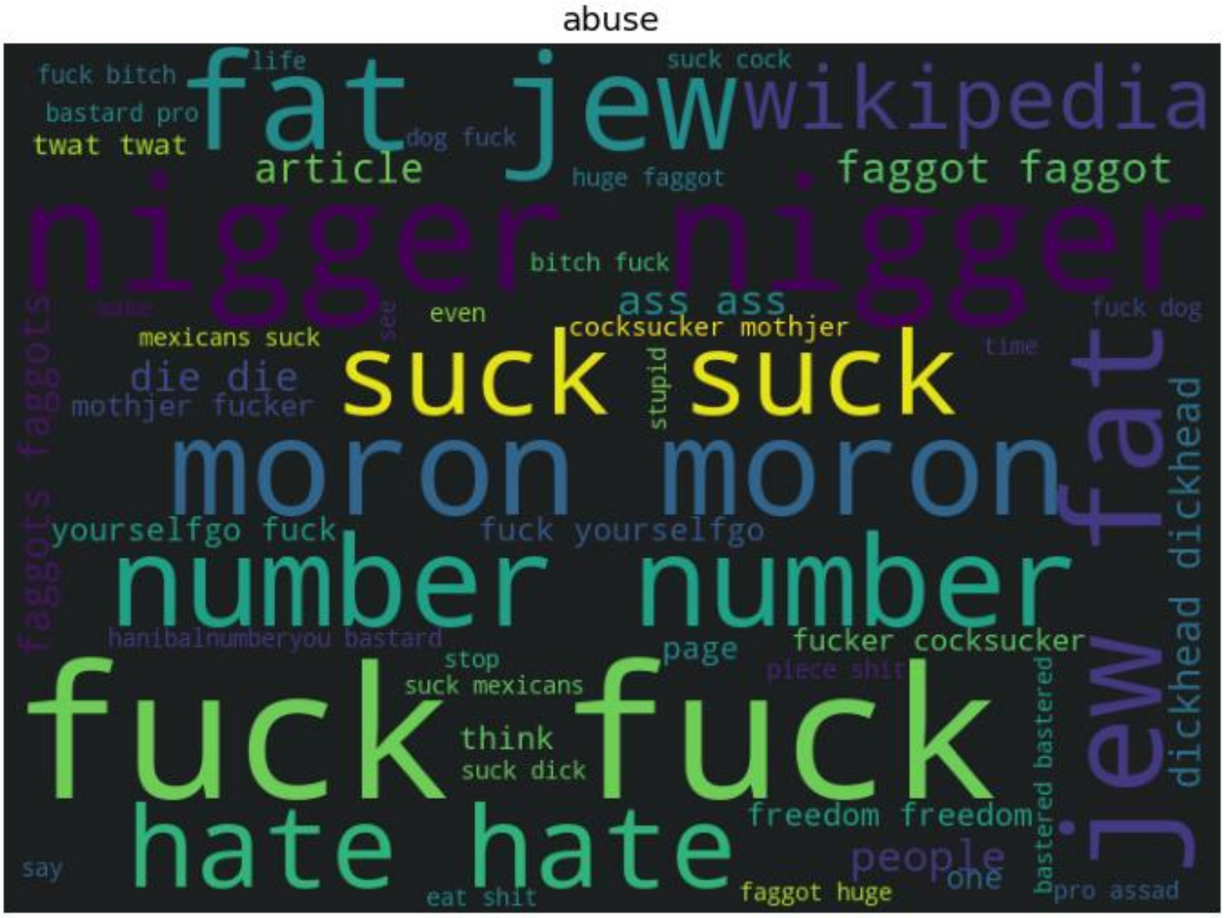
c. rude


rude
one
mothjer fucker
ass ass
die die
fuck yourselfgo
fucksex fucksex
twat twat
suck dick
piece shit
poop poop
number number
nipple nipple
shit shit
bastered bastered
fucking
article
dickhead dickhead
freedom freedom
buttsecks buttsecks
nigger nigger
faggots faggots
fuck yourselfgo
jew fat
shut fuck
cocksucker mothjer
fuck dog
think
say
offfuck offfuck
suck suck
people
fuck shut
know
time
see
suck cock
fucker cocksucker
hanibalnumberyou bastard
fat jew
even
huge faggot
hate hate
fuck fuck
want
boobs boobs
page
make
wikipedia
bullshit bullshit
faggot huge

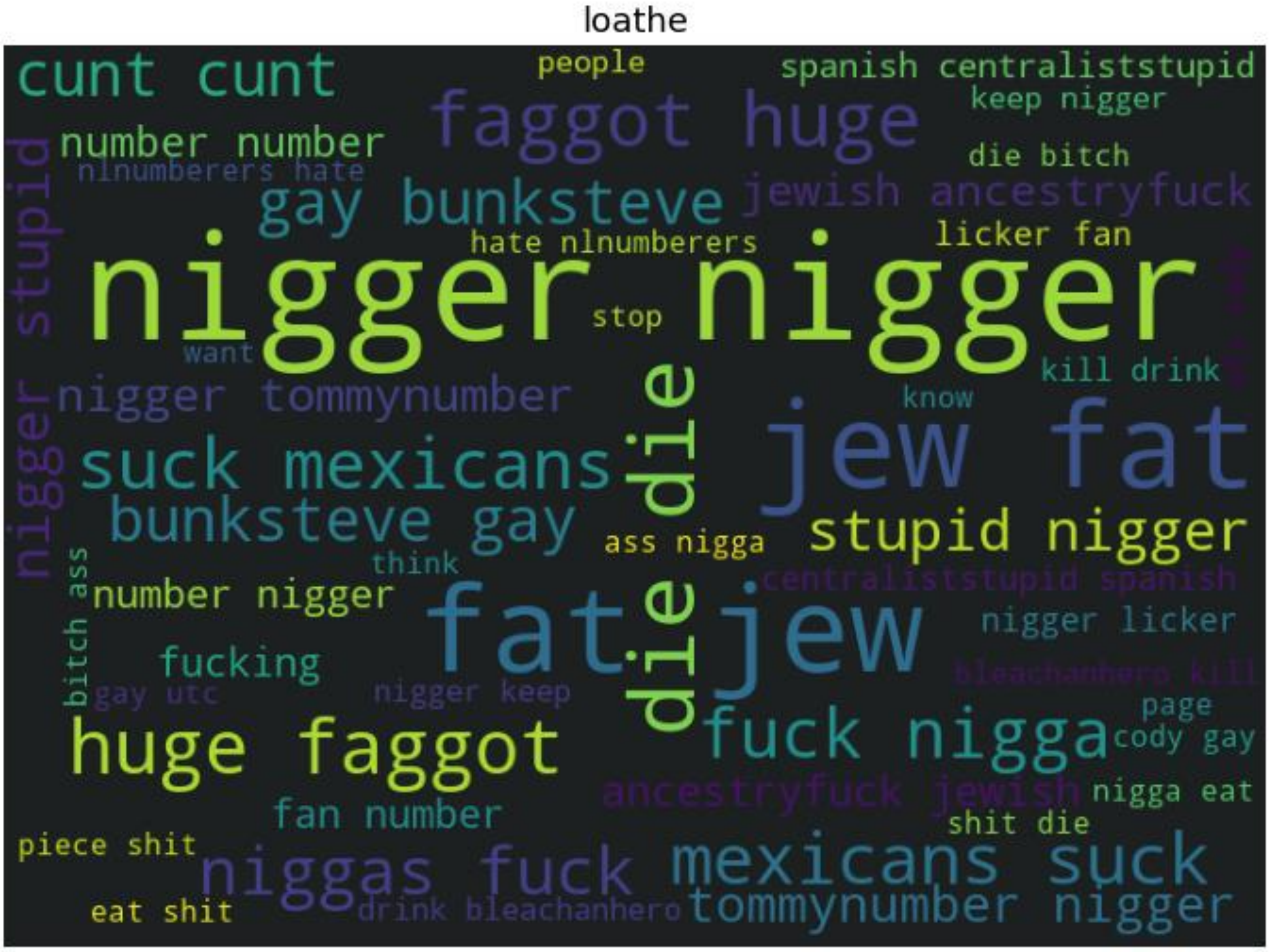
d. Threat



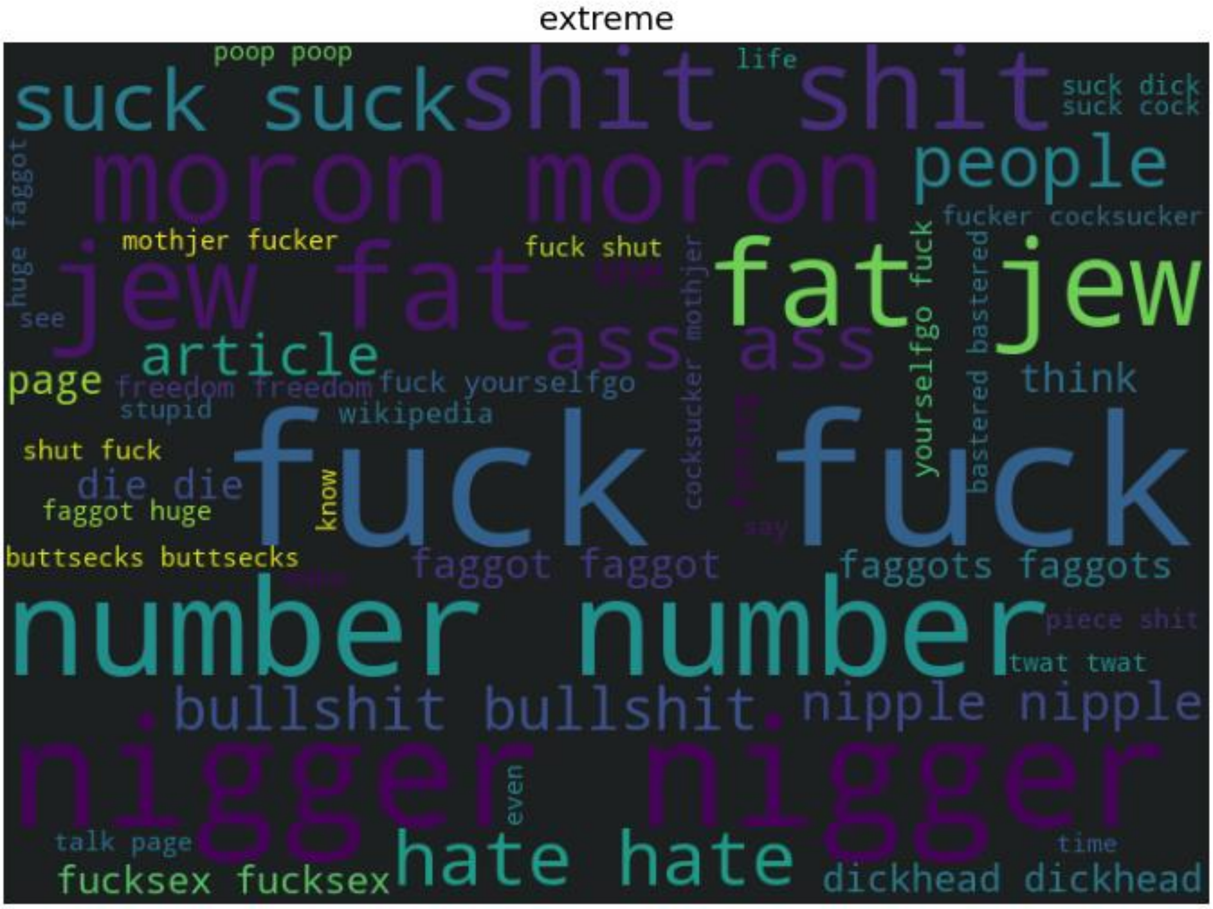
e. abuse



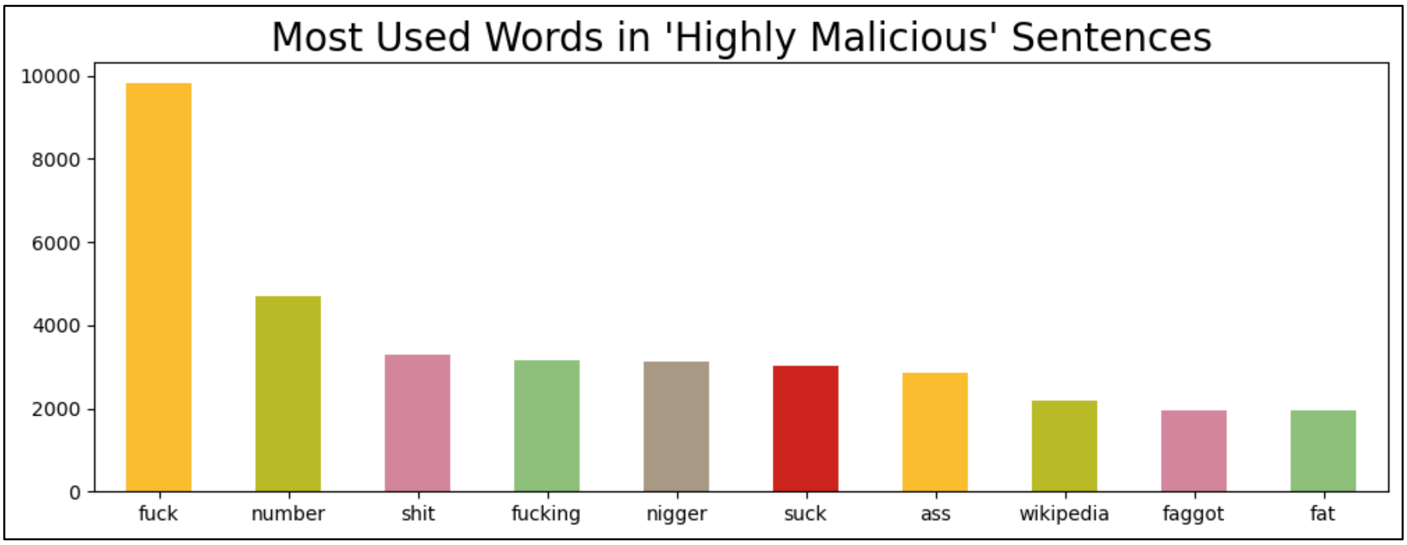
f. Loathe



g. extreme



iii. Plotting the most frequently used highly malicious words



V. Interpretation of the Results

a. “fuck” was seen to be the most frequently used word, followed by some kind of numbers and then the word “shit”

CONCLUSION

I. Learning Outcomes of the Study in respect of Data Science and Limitations of this work and Scope for Future Work

a. SMOTE was used to over sample the data however, it took a lot of time to run the best models on them due to the hardware restrictions.

b. An ROC_AUC curve could have been plotted but due to errors arising with multinomial naïve bayes, that was not possible.