
# MALIGNANT COMMENT CLASSIFICATION

Submitted by-

Anurag Kumar

# ACKNOWLEDGMENT

This project needed data provided by Flip Robo Technology. Necessary technical help. Some websites such as stackoverflow.com, Datatrained.com, guru99.com helped a lot for doubt clearing. Mentor at Flip Robo guided me through all the speed brakers I faced during this project completion.

# 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 a problem 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.

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-

We all are social media users and came across all types of comment. There are many articles published regarding online hate and abusive language used. So, there is need to classify such comments under different headings.

## Motivation for the problem undertaken

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.

# Analytical Problem Framing

## Data Sources and their formats-

Data was provided in csv format by flip robo which contains two csv file one for training the model and another one for testing. Training data set have 8 columns of independent variable and two columns in test data set.

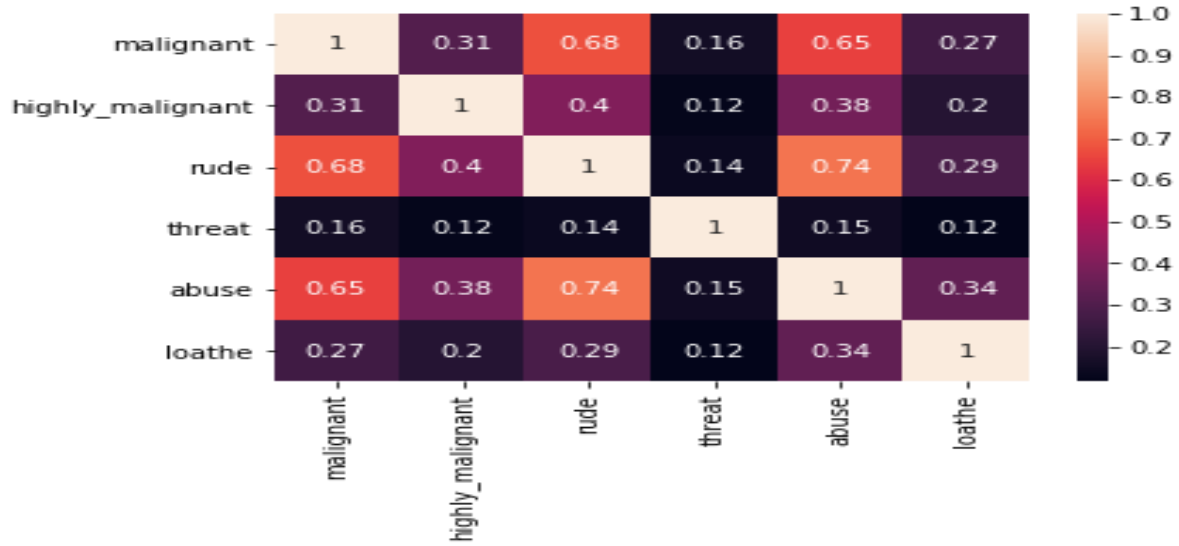
## Data Pre-processing Done-

First hurdle in making ML model is textual data values. For solving this problem, we first convert into lower case and remove all punctuations. Used SMOTE technique for imbalance data.

- ✓ Convert all messages to lower case
- ✓ Replace email addresses with 'email'
- ✓ Replace URLs with 'webaddress'
- ✓ Replace money symbols with 'moneysymb'
- ✓ Replace 10 digit phone numbers (formats include parenthesis, spaces, no spaces, dashes) with 'phonenumber'
- ✓ Replace numbers with 'numbr'
- ✓ Remove punctuations
- ✓ Remove stop words
- ✓ Word Lemmatize.

## Data Inputs- Logic- Output Relationships-

Some features are positively correlated, and some are negative. Which is visualize in following figure.




Library Used- Numpy, Pandas, Matplotlib, Seaborn, NLTK.

# Model/s Development and Evaluation

## Identification of possible problem-solving approaches-

Convert textual data into numerical using tf-idf approach. Which convert data into numerical. Then passing integer input into different models.

## Testing of Identified Approaches (Algorithms)-

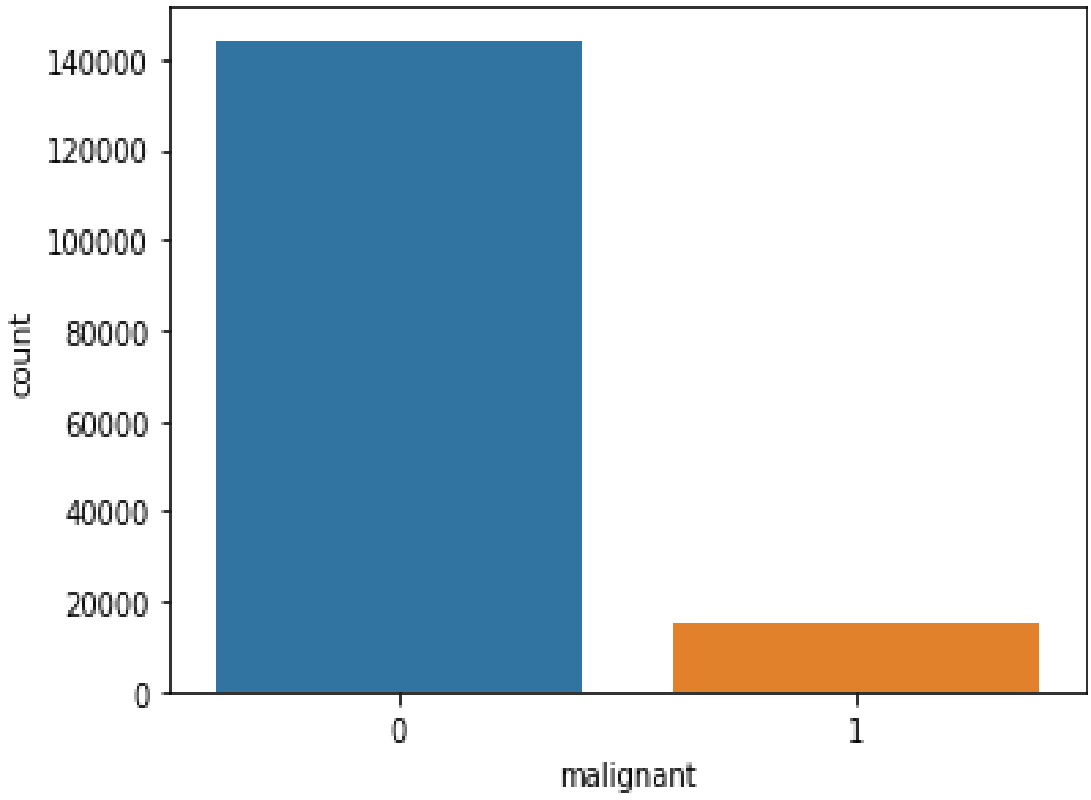
Split data set into train and test. Further feed it into different classification model. Merge all output in in column and convert it into 0 & 1(malignant and non-malignant).

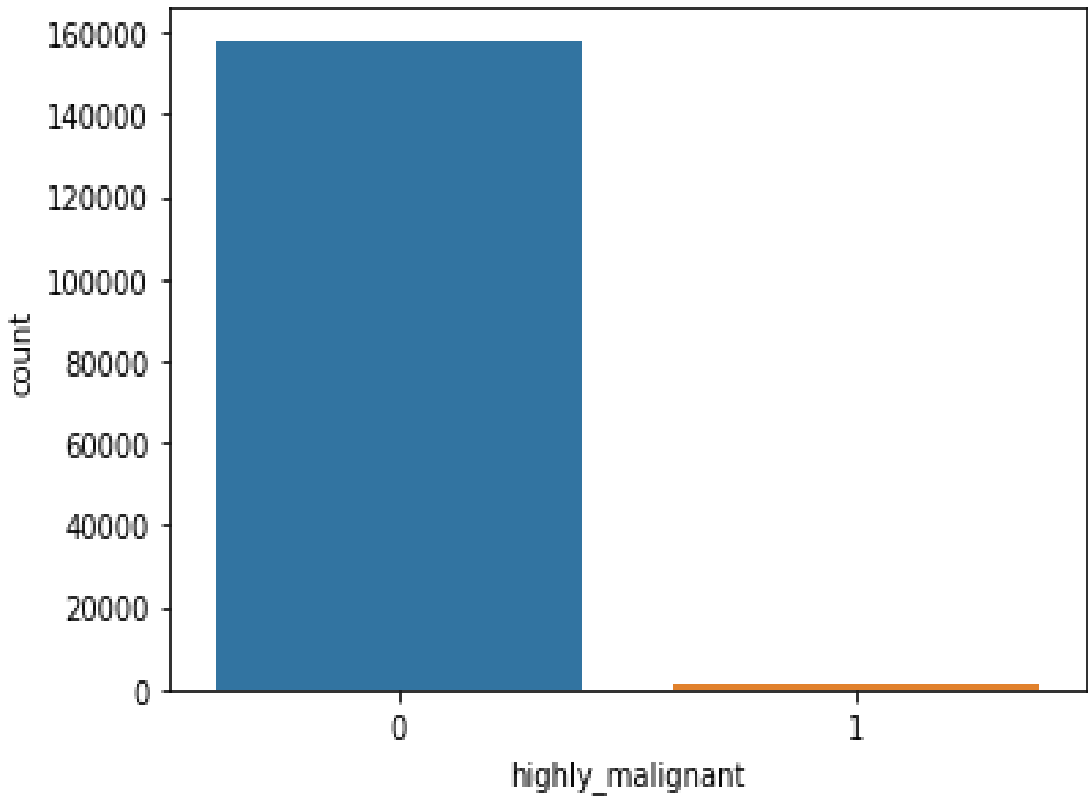
.

## Run and evaluate selected models –

Different approaches here for model building are Logistic Regression, Decision Tree & Random Forest Classifiers. And evaluation is done through Roc-Auc curve.

## Visualization-

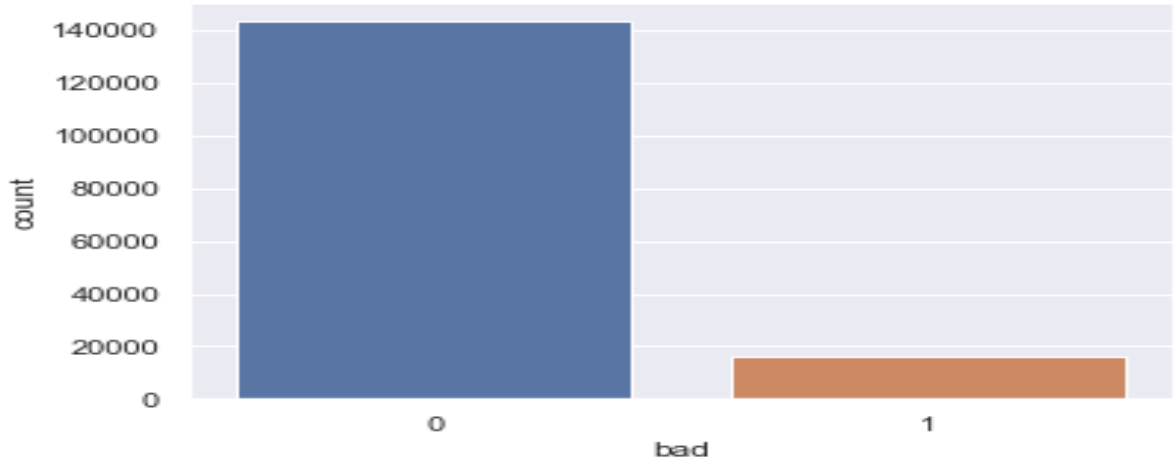



160000
140000
120000
100000
80000
60000
40000
20000
0
count
0
1
highly_malignant

Word Cloud-


even
page
article
know
go fuck
fuck fuck
nipple nipple
hate hate
numbr numbr
say
wikipedia
hi moron
fat jew
suck suck
faggot faggot
people
one
bark bark
moron hi
block
think
see
bullshit bullshit
jew fat
pig pig
nigger
nigger
stupid
wanker wanker
shit shit
ball ball

Characterization of Malignant (0) and non-Malignant (1)




Different Classification Report output-

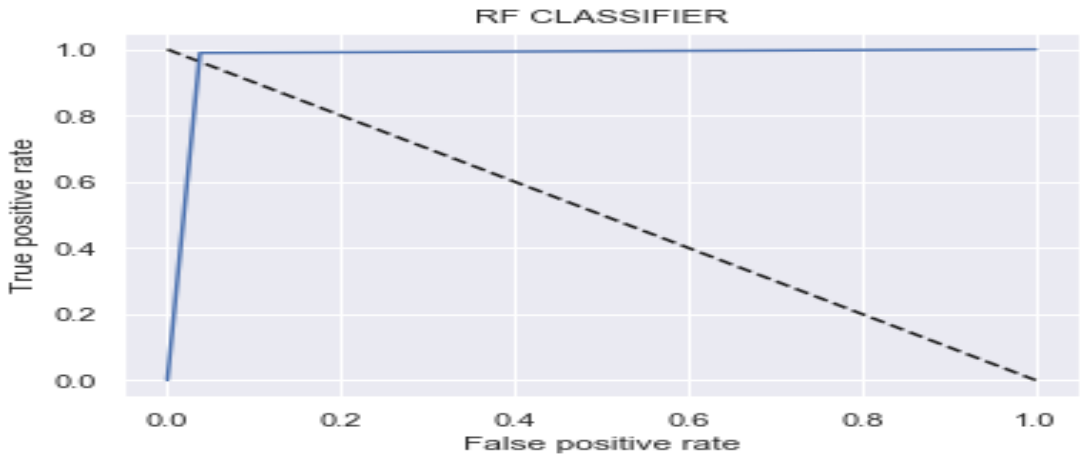
| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.91 | 0.94 | 43173 |
| 1 | 0.91 | 0.97 | 0.94 | 42835 |
| accuracy | | | 0.94 | 86008 |
| macro avg | 0.94 | 0.94 | 0.94 | 86008 |
| weighted avg | 0.94 | 0.94 | 0.94 | 86008 |

Different Model Score-

| Model | Training score | Testing Score |
|---|---|---|
| **Logistic Regression** | 93.76 | 92.866 |
| **Decision Tree** | 99.79 | 93.83 |
| **Random Forest** | 99.79 | 97.58 |

Random Forest Classifier turned out to be better model.

Roc-Auc curve-




**Conclusion-**

- Attained 97% accuracy in test dataset.
- There were imbalanced data which was rectified using SMOTE technique.
- Roc curve shows maximum area are under the curve.
- Model predicted following results- 1- 77850, 0-75314