

**MALIGNANT COMMENT**

**CLASSIFICATION**

Submitted by:

Preeti Singh

# ACKNOWLEDGMENT

I would like to thank Flip Robo Technologies for providing me with the opportunity to work on this project from which I have learned a lot. I am also grateful to Ms. Gulshana Chaudhary for her constant guidance and support. Reference sources are: -



* Google
* Stackoverflow.com
* Analytics vidhya
* Notes and repository from DataTrained

Research papers – <https://www.nltk.org/book/ch05.html>

[https://www.analyticsvidhya.com/blog/2017/01/ultimate-guide-tounderstand-implement-natural-language-processing-codes-in-python/](https://www.analyticsvidhya.com/blog/2017/01/ultimate-guide-to-understand-implement-natural-language-processing-codes-in-python/)

# 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.

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.

## • Conceptual Background of the Domain Problem

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 unoffensive, 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.

# Analytical Problem Framing

## • Mathematical/ Analytical Modelling of the Problem



This was an NLP Project and, in this project, we deal with the textual data and for understanding the data we used some methods like removing punctuations, numbers, stop words and using the lemmatization process convert the complex words into their simpler forms. These processes helped in cleaning the unwanted words form the comments and we were left with only those words which were going to help in our model building. After cleaning the data, we used TF-IDF Vectorizer technique to convert textual data into vector form. This technique works on the basis of the frequency of words present in the document. After training with train dataset, we use this technique into test dataset.

## • Data Sources and their formats

This data was provided to me by FlipRobo Technologies into a csv file format. This file contains training and testing dataset. On training dataset, we build a model and using this model we have to predict the outcomes from testing dataset.

The data set contains the training set, which has approximately 1,59,000 samples and the test set which contains nearly 1,53,000 samples. 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 0 denotes 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.

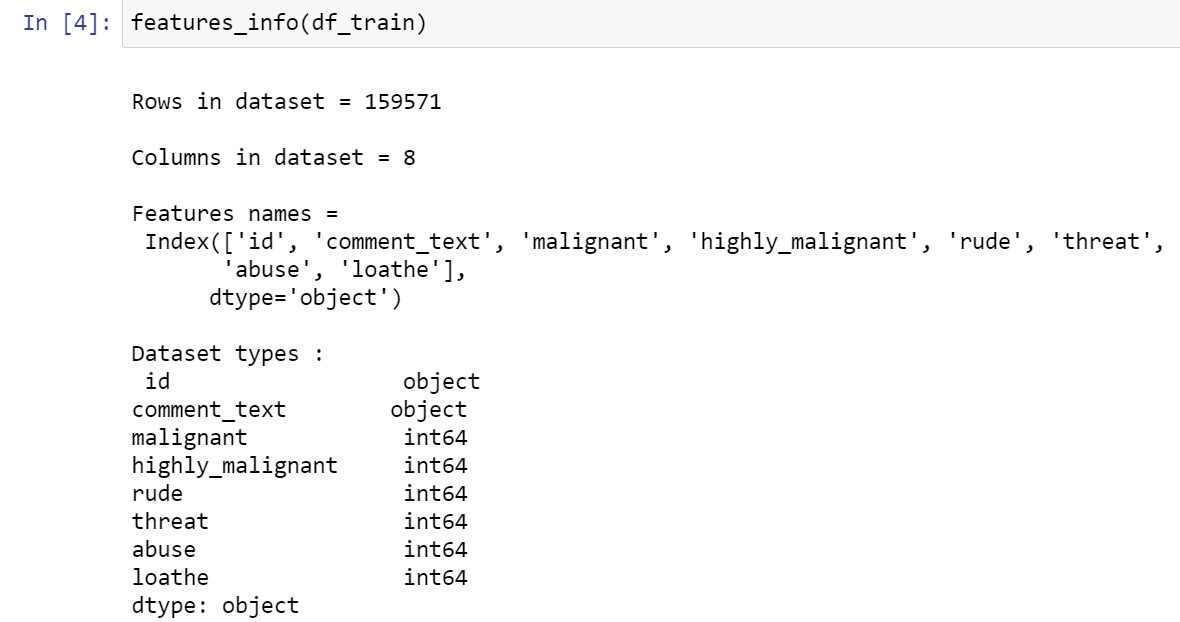
-Abuse: It is for comments that are abusive in nature.

-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.

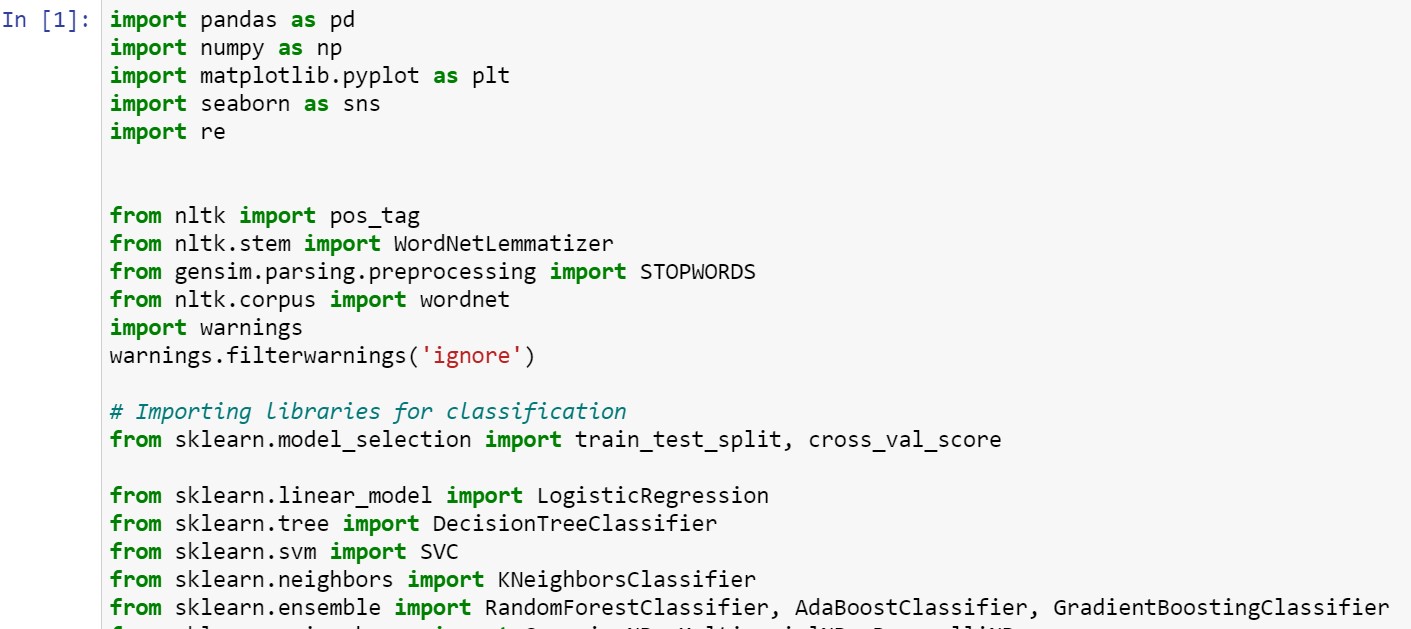
First of all, we upload the training dataset into df\_train dataframe and check the datatypes present in it.

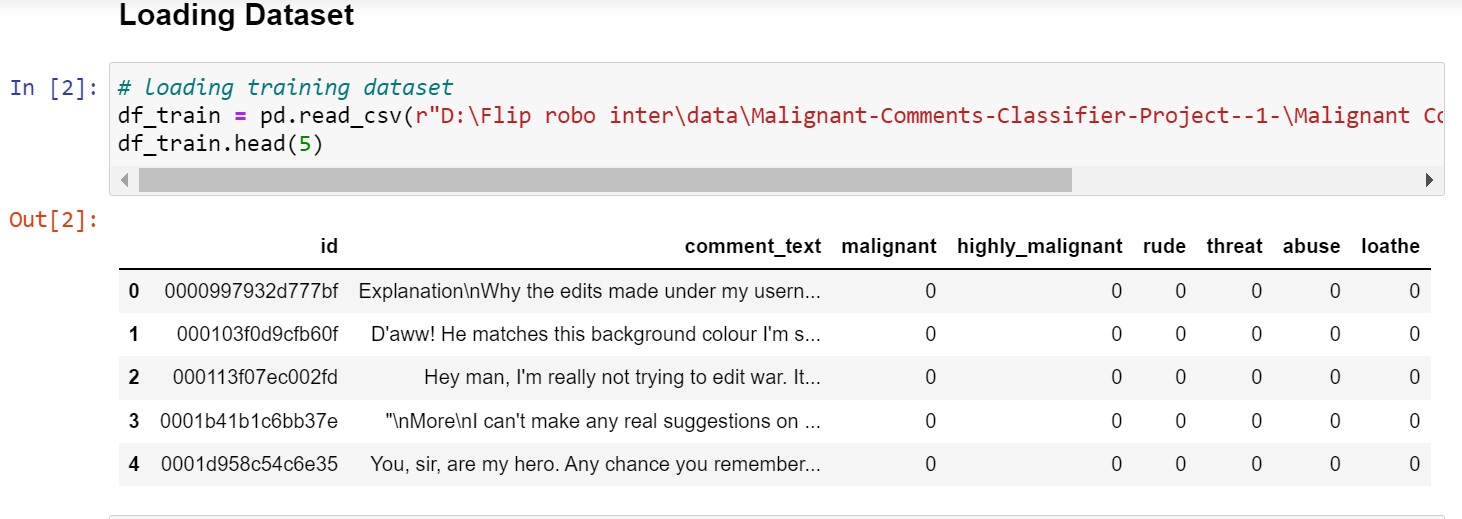


So, the data present in training data is both object and integer in nature. 2 columns are object dtypes and 6 columns are integer in dtypes.

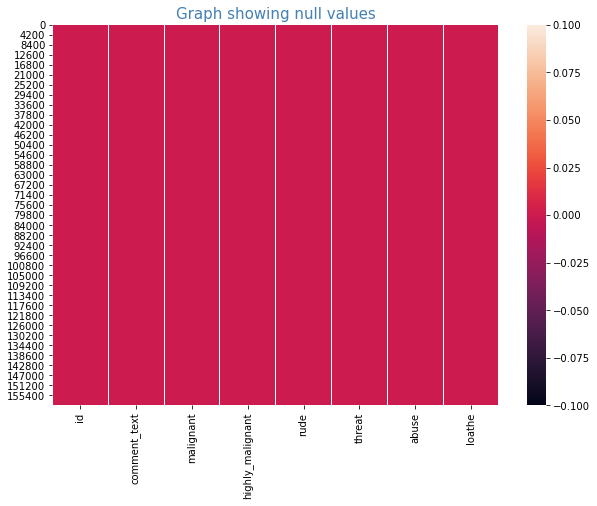
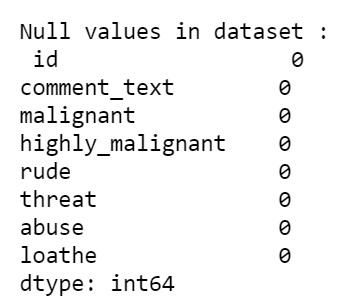
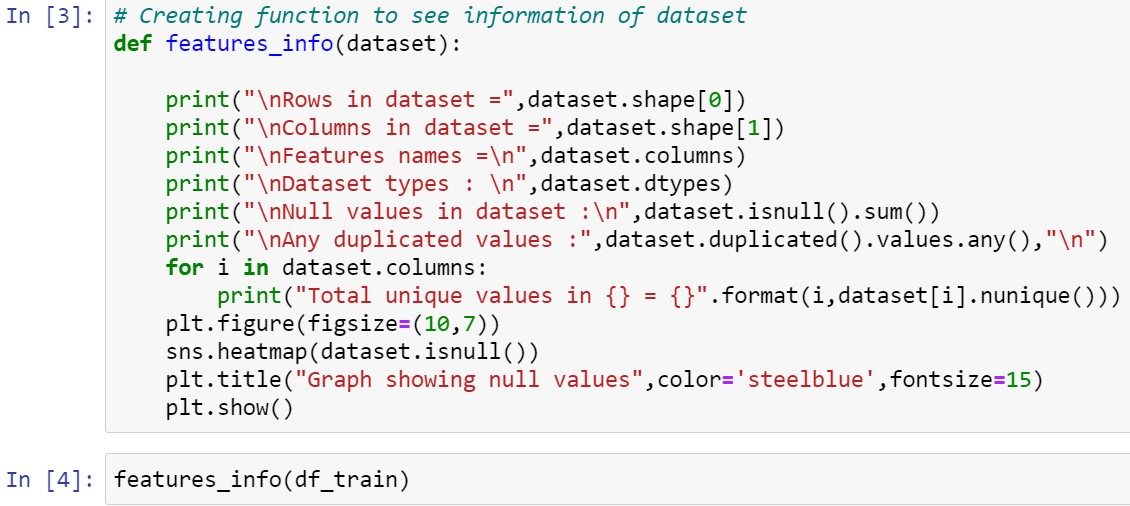
## • Data Pre-processing Done

Data pre-processing is the data mining technique that involves transforming raw data into an understandable data format. So, in data pre-processing technique first step is import data and the libraries to be used in model building.



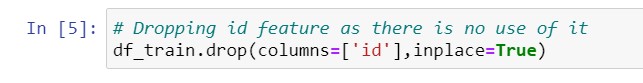


Now after loading the dataset, we check the missing values –



So, there were no null or missing values in our dataset we move to next step of data cleaning –

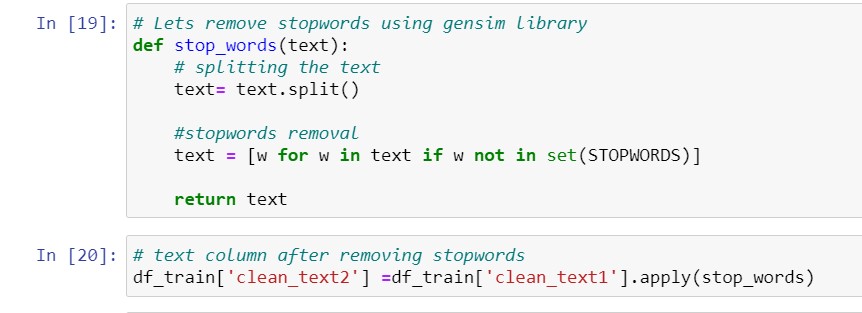
In data cleaning we dop the ID column as it gives no information. After dropping it we created another feature “negative\_cmnts” which shows the labelled data of positive and negative comments.



Now in cleaning the textual data we created a function to remove unwanted space, punctuation, numbers, emails, phone numbers etc and converted upper case letters into lower case and append the result into a new column.



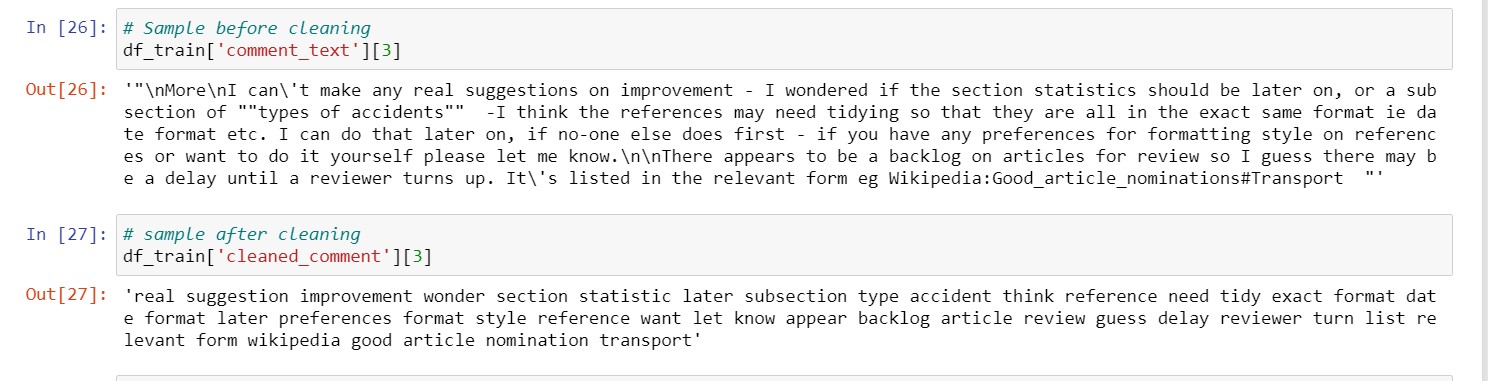
After the removal of unwanted notations, we moved to remove stopwords from our dataset. For removing them we created another function named stop\_words and append the text into another new column.



After removing all the unnecessary words or numbers we converted the words into their simpler form using the Lemmatization process. In this process we defined two functions, first one will tag the words into proper format and the other function will convert them into simpler form using the tagged alphabet.



Now we took a random sample from our dataset and compared the text of before and after the treatment.



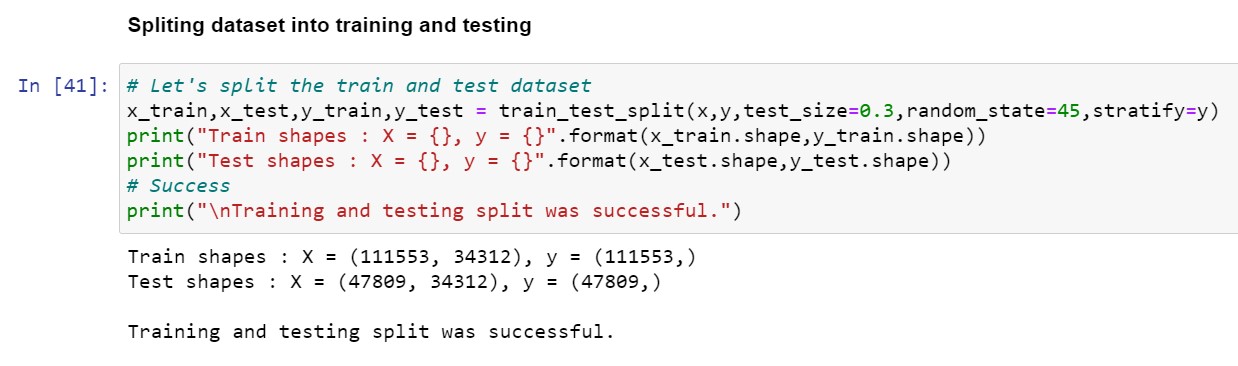
Encoding the categorical data (Feature Extraction and scaling)

As of now we had cleaned the data, now another major step towards building a model is to convert the textual data into numerical form because our algorithms understand only the numerical data. So, for converting into numerical or vector form we used Tf-Idf Vectorizer technique which converted the textual data into the vectors using the terms frequency method.

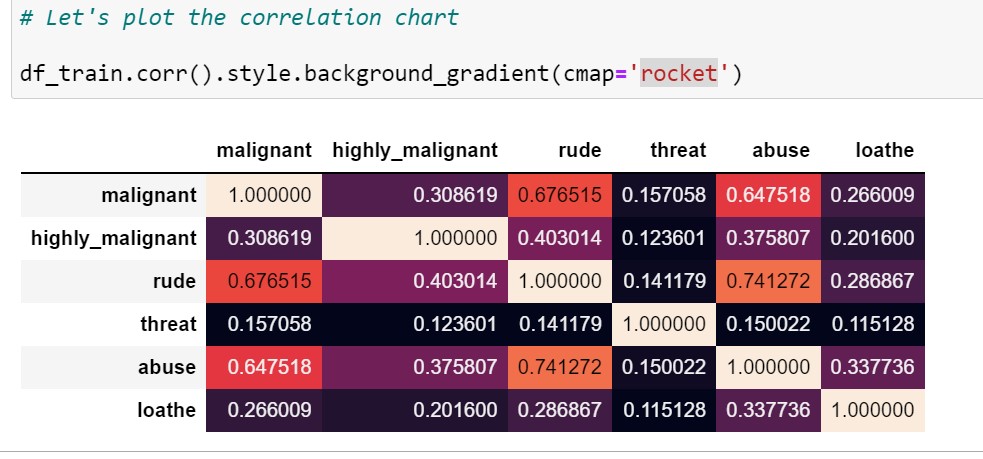


Splitting the dataset **–**

The last step for data pre-processing is splitting the dataset into training and testing dataset Using the train\_test\_split model selection method we converted the dataset into training and testing.



## • Data Inputs- Logic- Output Relationships



All the harmful comments present in dataset are having positive correlation with each other’s. Rude and abuse are highly correlated.

## • Hardware and Software Requirements and Tools Used

There is no such requirement for hardware, but I have used intel i5 8th generation processor with 6 GB Ram.

Software: Jupyter Notebook (Anaconda 3)

Language: Python 3.9 Libraries used in project:

1. Pandas
2. Numpy
3. Matplotlib
4. Seaborn
5. Sklearn

# Model/s Development and Evaluation

## • Identification of possible problem-solving approaches (methods)



In this project there were 5-6 features which defines the type of comment like malignant, hate, abuse, threat, loathe but we created another feature named as “negative\_cmnts” which is combined of all the above features and contains the labelled data into the format of 0 and 1 where 0 represents “NO” and 1 represents “Yes”.

As we have labelled data into our target feature which is the case of classification method. So, we are going to use algorithms based on classification.

## • Testing of Identified Approaches (Algorithms)

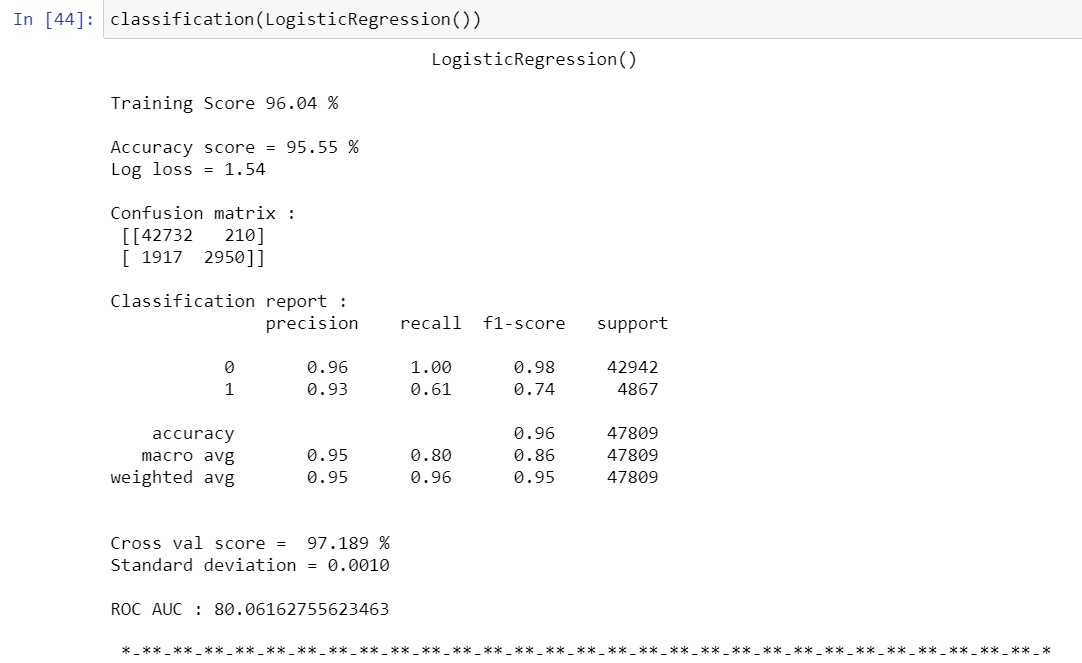
Based on the classification approach we are going to use following approaches:

1. Logistic Regression
2. Decision Tree Classifier
3. RandomForest Classifier
4. AdaBoost Classifier
5. GradientBoosting Classifier

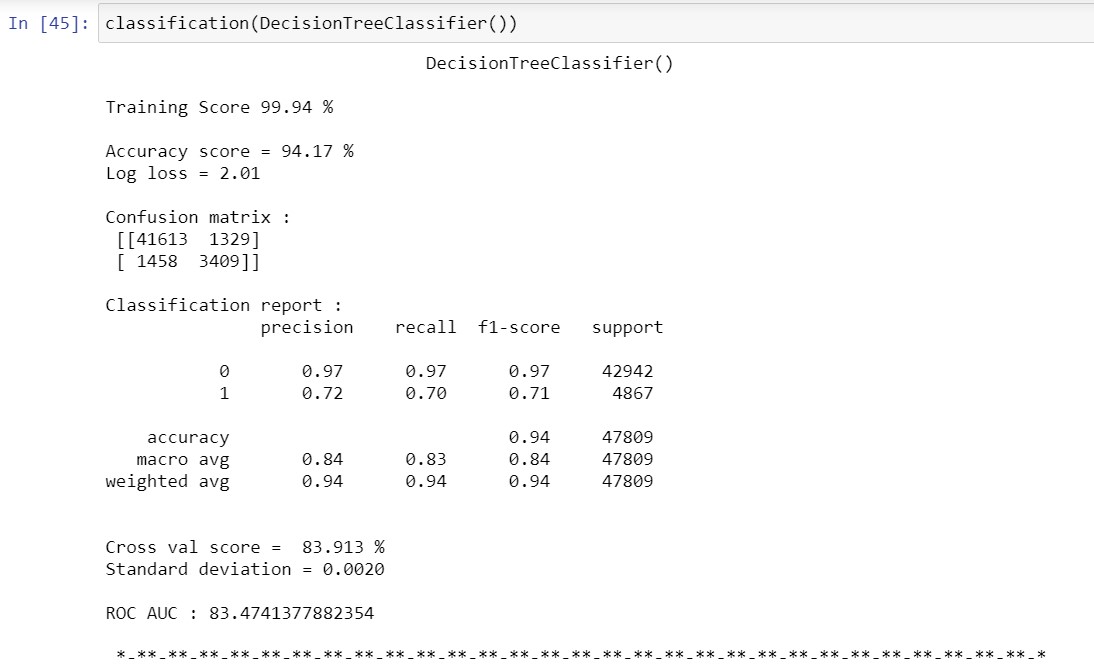
In NLP we use Naïve Bayes algorithms mostly but due to our systems we faced memory error to deal with them.

## • Run and Evaluate selected models

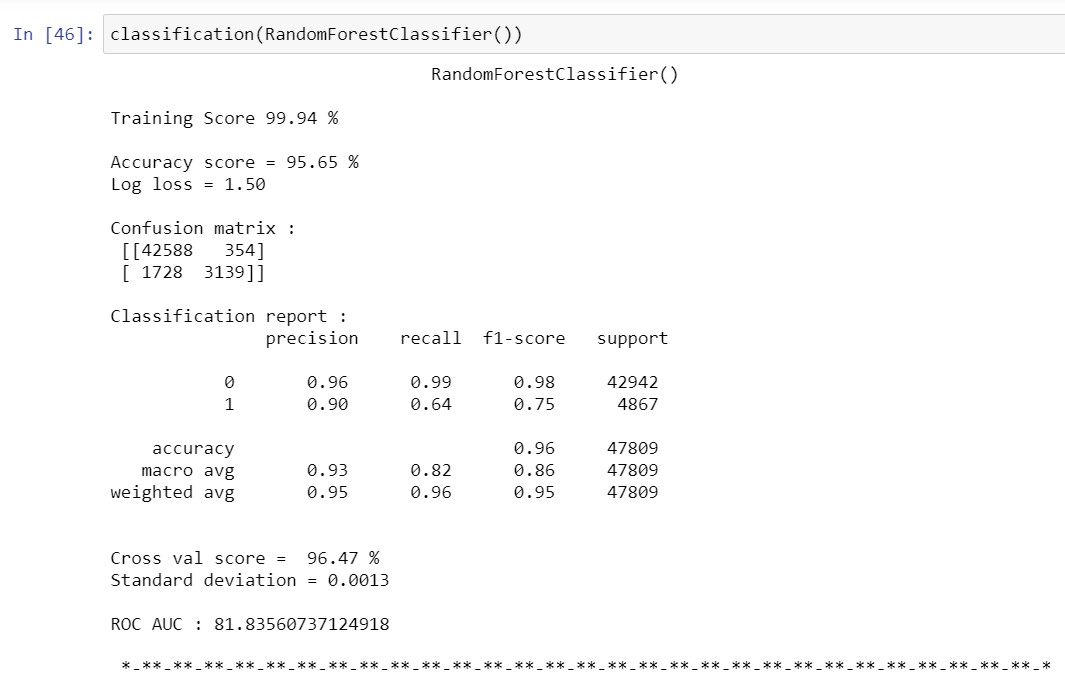
1. Logistic Regresison -



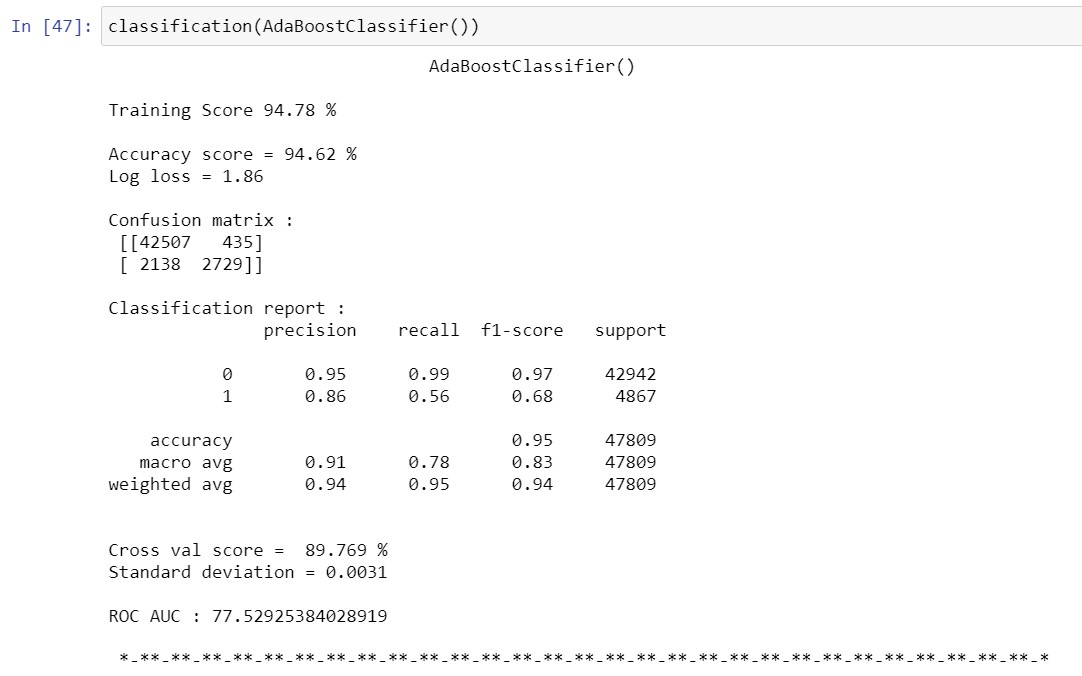
1. DecisonTree Classifier –



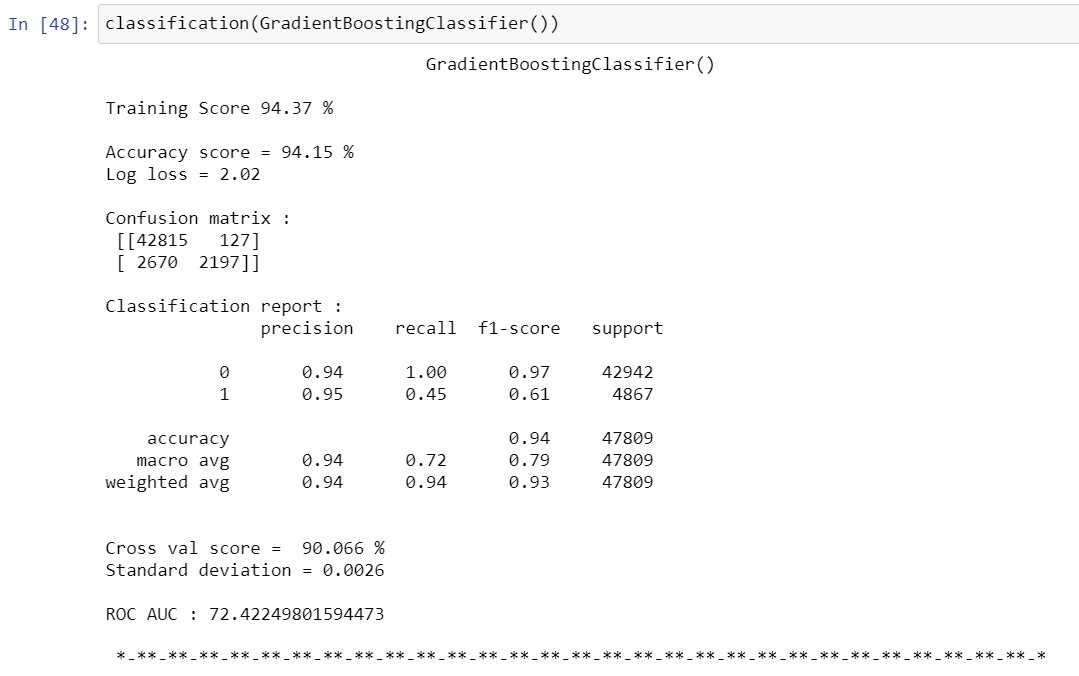
1. RandomForest Classifier –

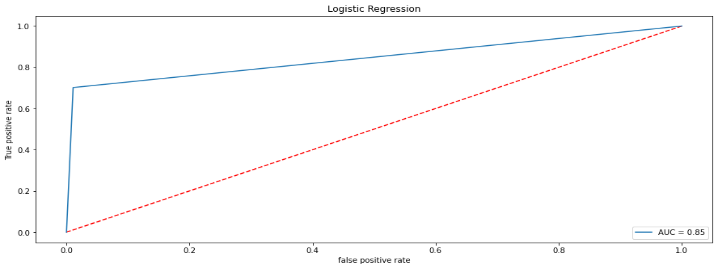
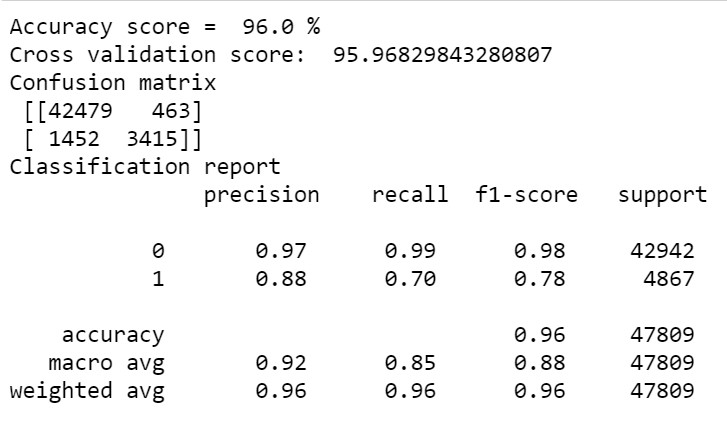
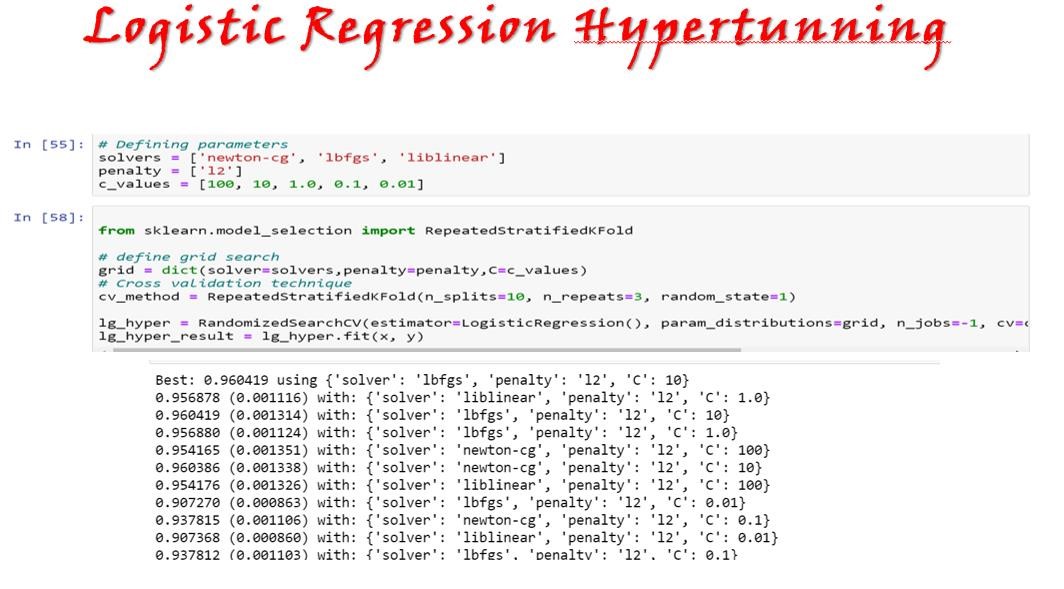


1. AdaBoost Classifier –



1. GradientBoosting Classifier –





## • Key Metrics for success in solving problem under consideration

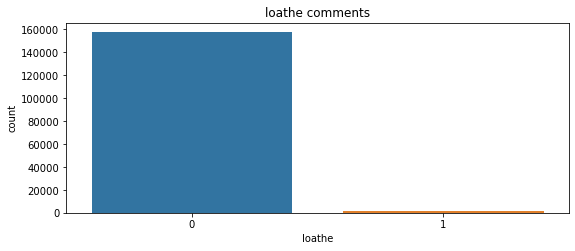
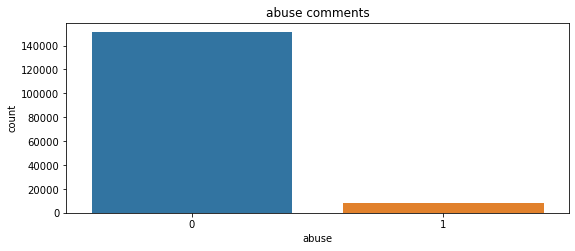
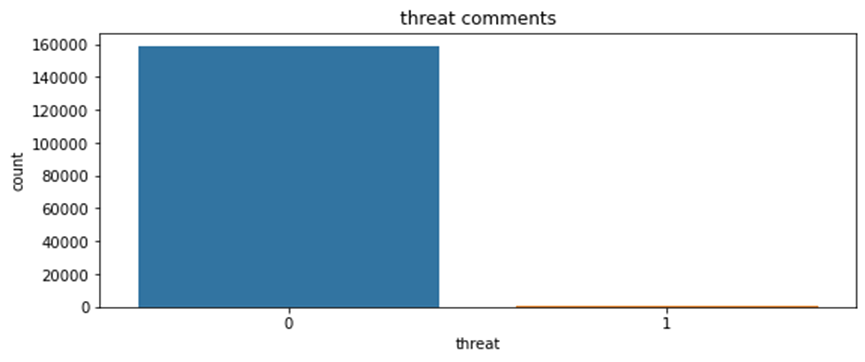
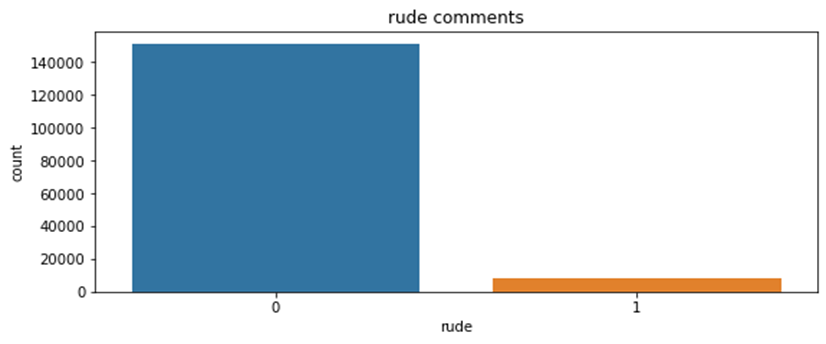
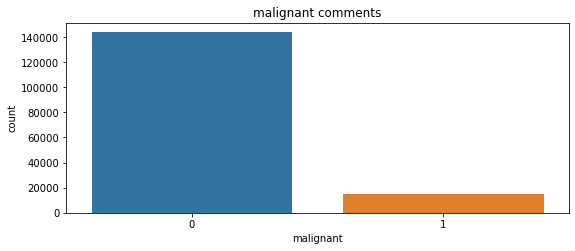
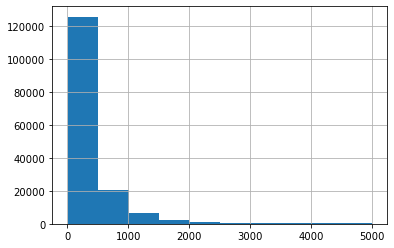
For solving the problems and understanding the result of each algorithm we have used a lot of metrics:

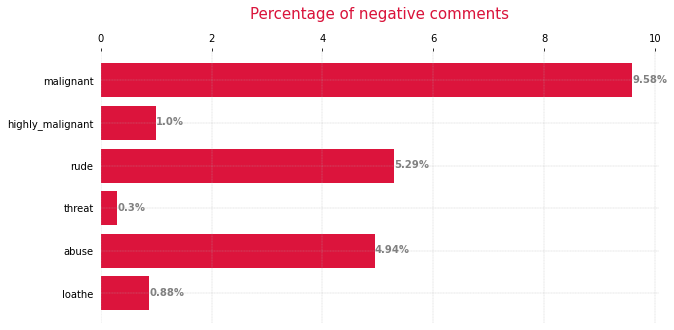
1. Accuracy Score
2. Classification Report
3. Confusion Matrix
4. Log Loss
5. Roc-Auc

## • Visualizations

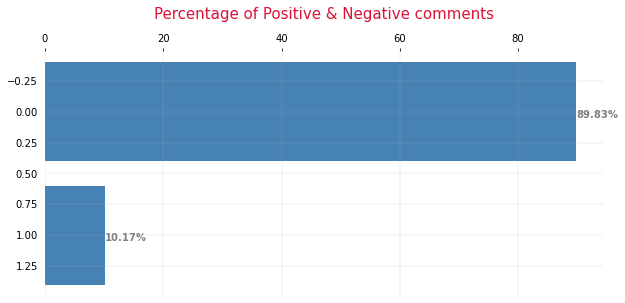
For the Visualization we have used Matplotlib and

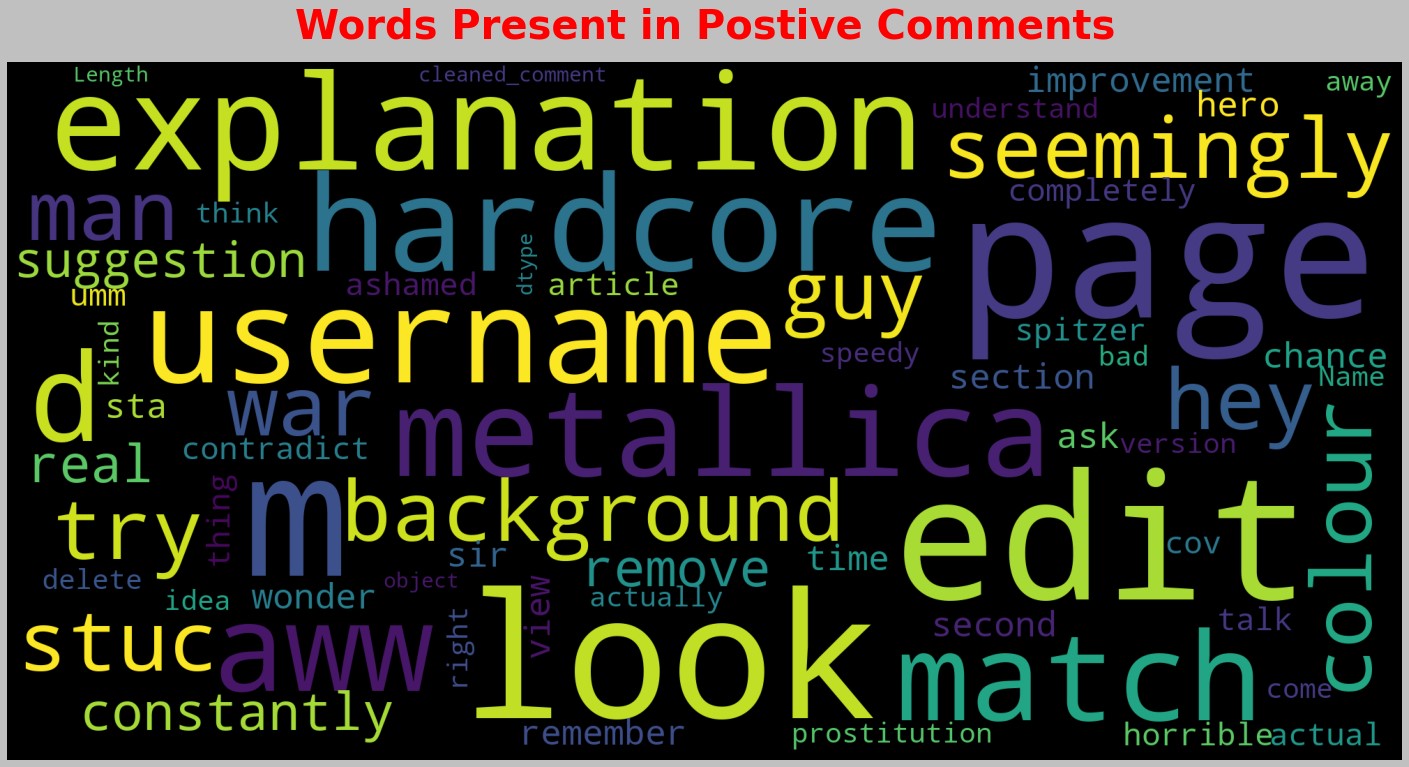
Seaborn library to plot the numerical data into graphs –

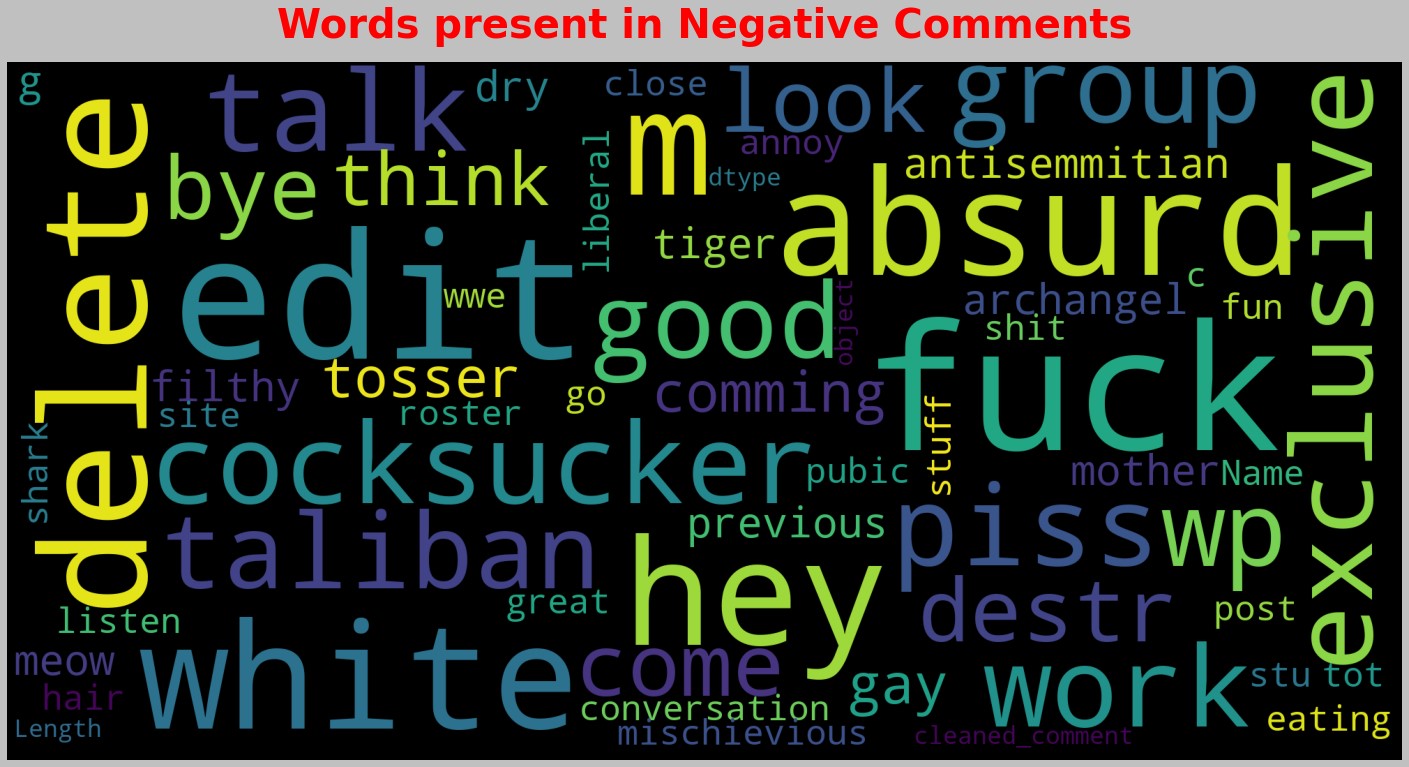






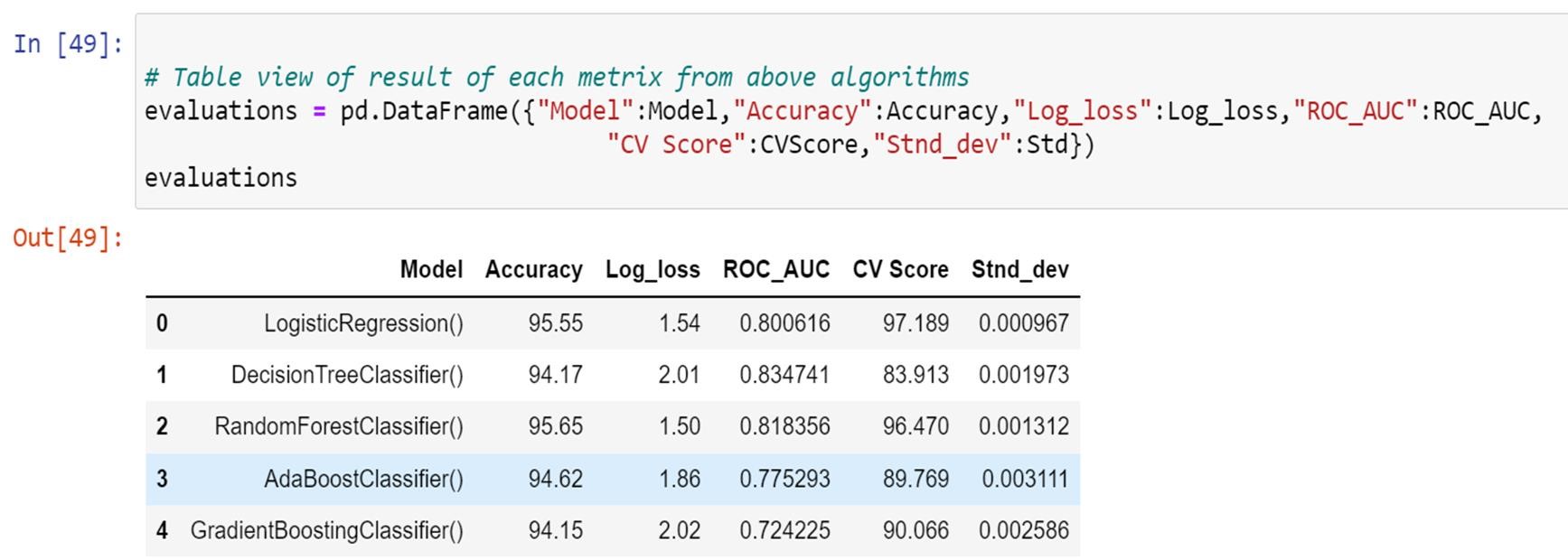








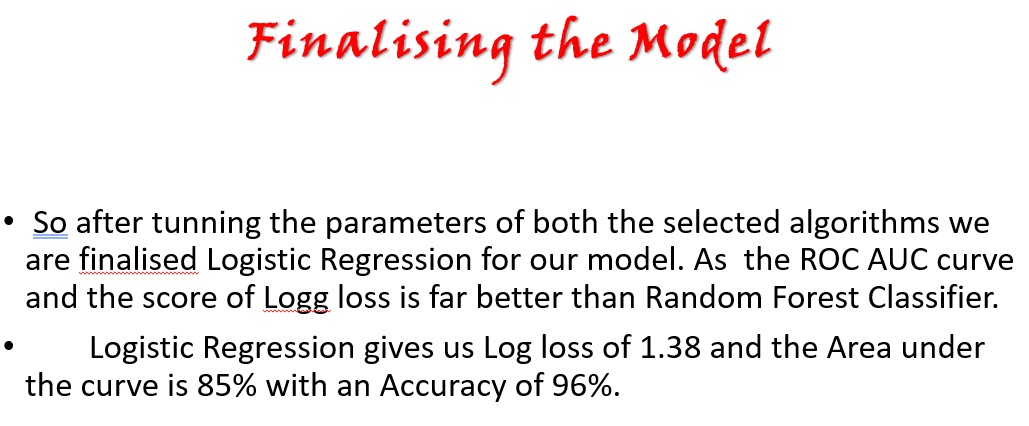
## • Interpretation of the Results



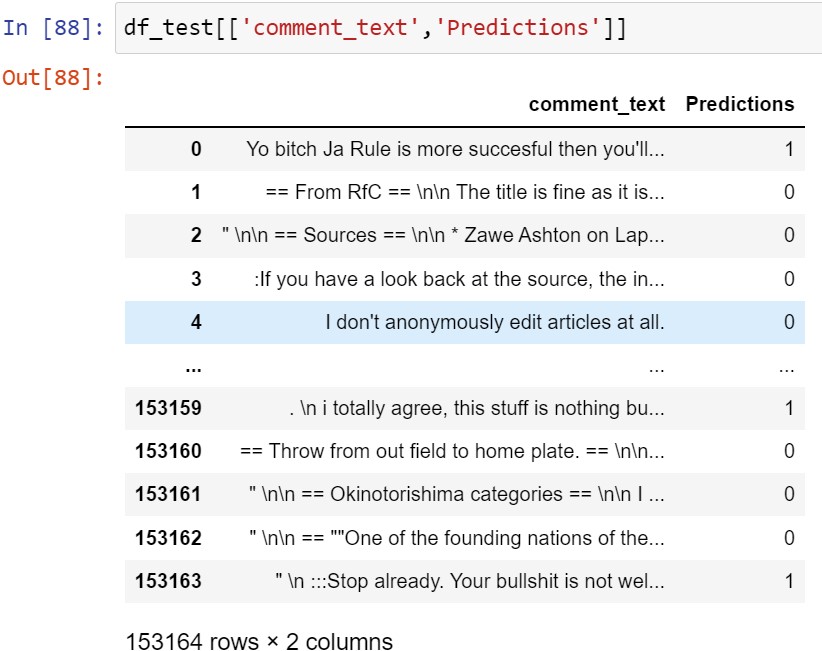
As our target feature is imbalanced which means that alone accuracy score will not give best results but instead accuracy, we are using the classification report log loss score, Roc-Auc score and confusion matrix to find the best algorthim which is above all.

So, we selected Logistic Regression and RandomForest Classifier based on above condition and use RandomizedSearchCV for hyper tunning the parameters of these 2 selected algorithms. After hyper tunning the parameters we selected the Logistic Regression algorithm as it gives upto 85% score of Roc-Auc and losgg loss of 1.38 far better than RandomForest Classifier whose log loss was above 2.0 and again the classification report also tells that it is better than RandomForest

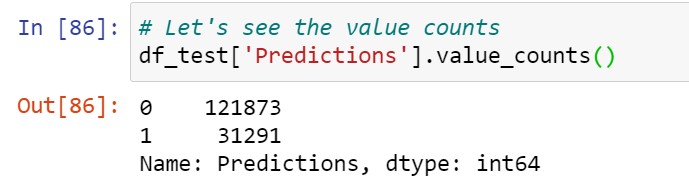
Classifier.



## • Predictions



These are the predictions from our testing dataset.



# CONCLUSION

## • Key Findings and Conclusions of the Study



The conclusion for our study: -

1. In training dataset, we have only 10% of data which is spreading hate on social media.
2. In this 10% data most of the comments are malignant, rude or abuse.
3. After using the word cloud, we find that there are so many abusive words present in the negative comments. While in positive comments there is no use of such comments.
4. Some of the comments are very long while some are very short.

## • Learning Outcomes of the Study in respect of Data Science

From this project we learned a lot. Gains new techniques and ways to deal with uncleaned data. Find a solution to deal with multiple target features. Tools used for visualizations gives a better understanding of dataset. We have used a lot of algorithms and find that in the classification problem where we have only two labels, Logistic Regression gives better results compared to others.

But due to our system we could not use algorithms which gives much better results in NLP project like GaussinaNB, MultinomailNM. We also used googlecolab and some pipelines techniques but none of them worked here and also it was too much time consuming.

## • Limitations of this work and Scope for Future Work

This project was amazing to work on, it creates new ideas to think about but there were some limitations in this project like unbalanced dataset, multiple target features. To overcome these limitations, we have to use balanced dataset so that during the training of dataset our algorithm will not give biased result.

## Thankyou