

MALIGNANT COMMENTS CLASSIFIER PROJECT

Submitted by:

POOJA MISHRA

**ACKNOWLEDGMENT**

I would also like to thank Flip Robo Technologies and Datatrained team who has given me such wonderful opportunity to learn more about Machine Learning and their mentors Shubham Yadav, Sajid Chaudhary for their constant encouragement, valuable suggestions, and cooperation.

This acknowledgement will remain incomplete if I fail to express my deep sense of obligation to my family members and God for their consistent blessings and believe in me.

Thank You.

**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 must 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.

Thus, it became necessary for an organization to have an automated system which effectively identify and keep a track record and can take strict actions, such reporting or blocking same to prevent such comments in future.

* Conceptual Background of the Domain Problem

The background for the problem originates from the multitude of online forums, where-in people participate actively and make comments. As the comments sometimes may be abusive, insulting or even hate-based, it becomes the responsibility of the hosting organizations to ensure that these conversations are not of negative type.

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.

* Review of Literature

Online forums and social media platforms have provided individuals with the means to put forward their thoughts and freely express their opinion on various issues and incidents. In some cases, these online comments contain explicit/ malignant language which may hurt the readers. Comments containing explicit language can be classified into myriad categories such as ‘Malignant’, ‘Highly malignant’, ‘Rude’, ‘Threat’, ‘Abuse’ and ‘Loathe’. The threat of abuse and harassment means that many people stop expressing themselves and give up on seeking different opinions.

To protect users from being exposed to offensive language on online forums or social media sites, companies have started flagging comments and blocking users who are found guilty of using unpleasant language. Several Machine Learning algorithms have been developed and deployed to filter out the unruly language and protect internet users from becoming victims of online harassment and cyberbullying.

* Motivation for the Problem Undertaken

This is a huge concern, as in world there are 7.9 billion people, and out of 7.9 billion, more than 4.6 billion are on social media, which means that everyone in three people uses social media platforms. 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.

Thus, it’s very necessary to deal with this problem that can be only possible by machine learning techniques. With help of Natural Language Processing, we try to recognize the intention of speaker by building models that can detect the words of toxicity like threats, obscenity, insults, and identity-based hates. Moreover, its is crucial to handle any such kind of nuisance to make a more user-friendly experience only after people can enjoy in participating in online discussions and conversations.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

As the task was to figure out whether the data belongs to zero, one, or more than one category out of the six listed labels: ‘Malignant’, ‘Highly malignant’, ‘Rude’, ‘Threat’, ‘Abuse’ and ‘Loathe’. Current problem is of classification in which the label can be either 0 or 1, where 0 denotes a NO while 1 denotes a YES.

In multi-class classification, we have one basic assumption that our data can belong to only one label out of all the labels we have. For example, a given picture of a fruit may be an apple, orange or guava only and not a combination of these.

In multi-label classification, data can belong to more than one label simultaneously. For example, in our case a comment may be toxic, obscene and insulting at the same time. It may also happen that the comment is non-toxic and hence does not belong to any of the six labels.

* Data Sources and their formats

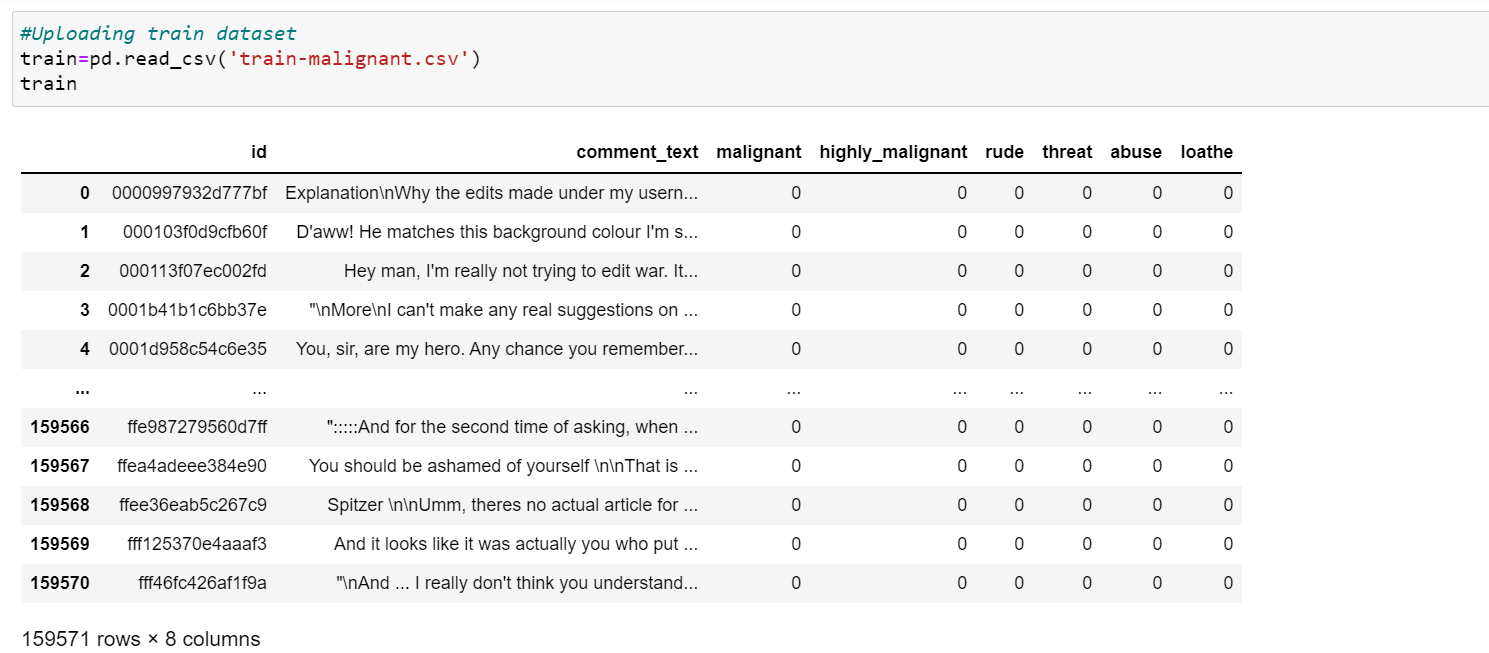
Given a group of sentences or paragraphs, used as a comment by a user in an online platform, classify it to belong to one or more of the following categories — , ‘Malignant’, ‘Highly malignant’, ‘Rude’, ‘Threat’, ‘Abuse’ and ‘Loathe’ with discrete values (0/1).

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.

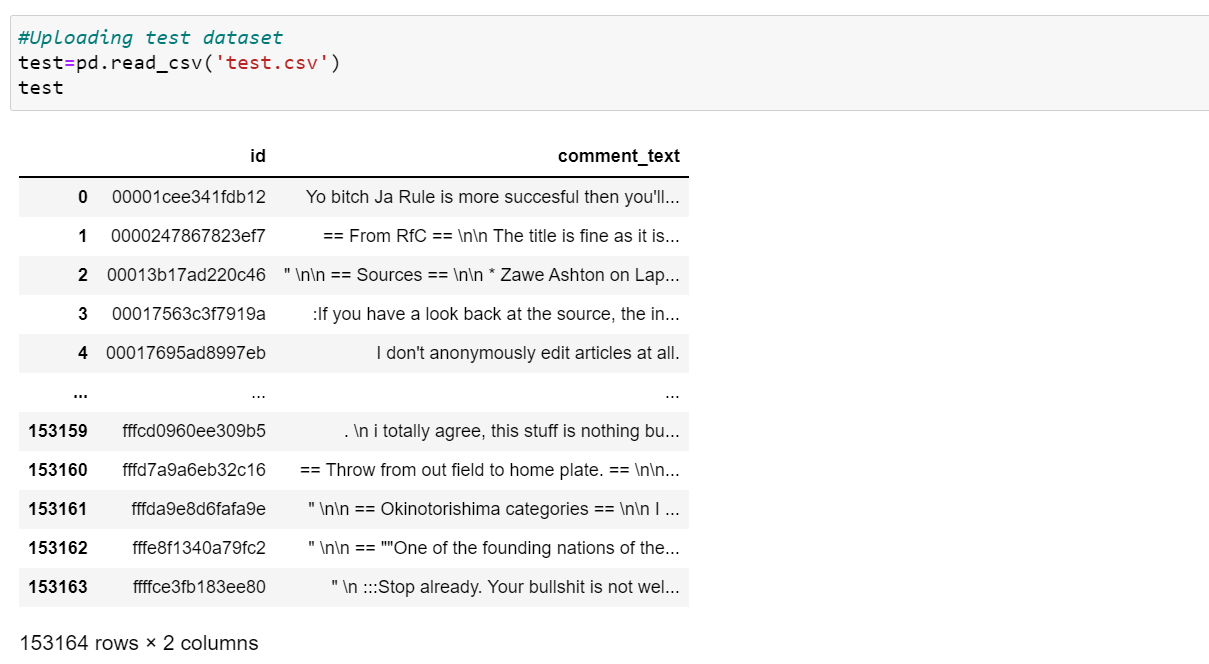
The Train data set includes:

* **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.



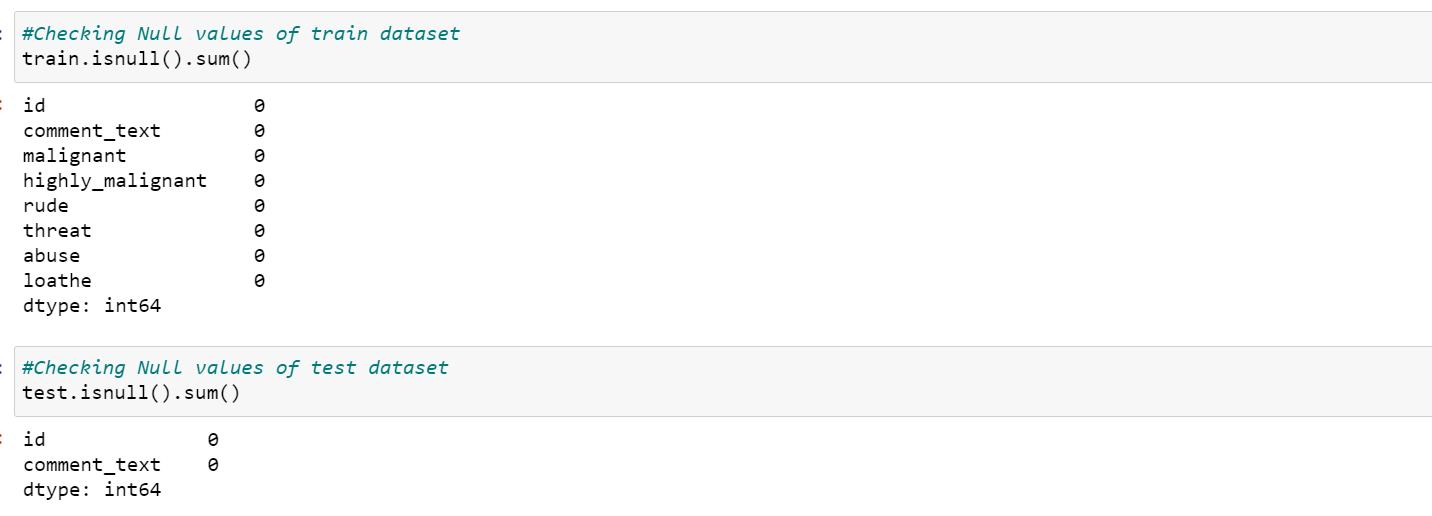
The Test data set includes:

* **ID:** It includes unique Ids associated with each comment text given.
* **Comment text:** This column contains the comments extracted from various social media platforms.



* Data Pre-processing Done
* Step 1: Checking for missing values.

First and foremost, after importing the training and test data into the panda’s data frame, I decided to check for missing values in the downloaded data. Using the “isnull” function on both the training and test data, I discovered that there were no missing records and therefore, I moved on to the next step of my project.



* Step 2 Text Normalization

The text normalization steps that I performed are listed below:-

* Removing Characters in between Text.
* Removing Repeated Characters.
* Converting data to lower-case.
* Removing Punctuation.
* Removing unnecessary white spaces in between words.
* Removing “\n”.
* Removing Non-English characters.



* Step 3 Lemmatising & Stemming

Stemming is the process of converting inflected/derived words to their word stem or the root form.

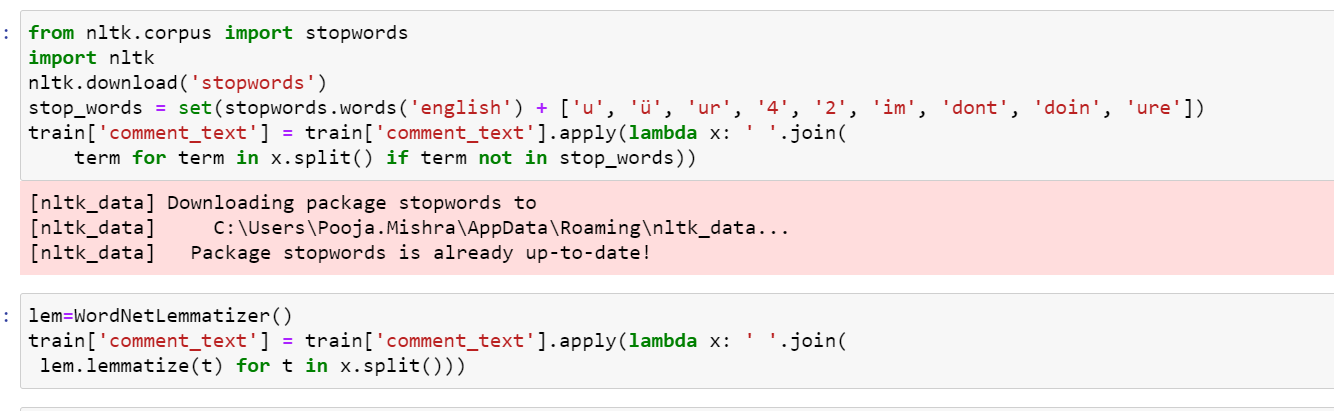
Lemmatisation is the process of grouping together the inflected forms of a word so they can be analyzed as a single item. This is quite like stemming in its working but not exactly same.

I used the *word-net library* in nltk for this purpose. Stemmer and Lemmatizer were imported from nltk.

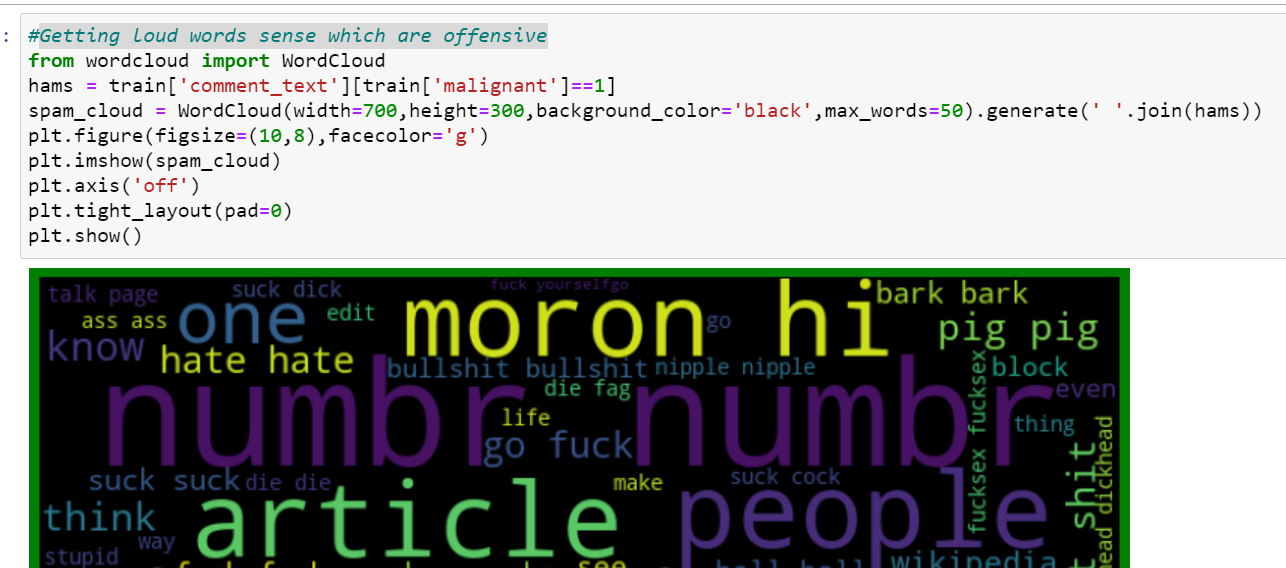


* Step 4 Stopwords Removal

Stopwords Removal, as we all know, is one of the most critical steps in text pre-processing for use-cases that involve text classification. Removing stopwords ensures that more focus is on those words that define the meaning of the text.



* Step 5 Getting loud words sense which are offensive



* Step 7 Converting Texts into Vectors



* Data Inputs- Logic- Output Relationships

Malignant Comment Classifier is a proliferation of social media. The data set for building the classification model was acquired from the social media sites and it included the training set as well as the test set. The current problem is of classification thus 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.

* State the set of assumptions (if any) related to the problem under consideration

This project has assumptions to prepare the best model to prepare the output dataset.

All labels assumed to be negative comments.

* Hardware and Software Requirements and Tools Used

Machine learning comes with an extensive collection of ML tools, platforms, and software.

Well, in this problem solution we have used following tools that helps to make this project

successful as per my possibility.

Jupyter notebook is one of the most widely used machine learning tools among all. It is a very fast processing as well as an efficient platform. Moreover, for this problem solution I have used python programming.

Scikit-Learn is built on top of the three main Python libraries viz. NumPy, Matplotlib, and SciPy. Along with this, it will also help you with testing as well as training your models.

The Libraries are as listed:

For Data loading and Visualisation:

Numpy - NumPy is very useful for handling linear algebra, Fourier transforms, and random

numbers.

Pandas - Pandas are turning up to be the most popular Python library that is used for data analysis with support for fast, flexible, and expressive data structures designed to work on both “relational” or “labeled” data.

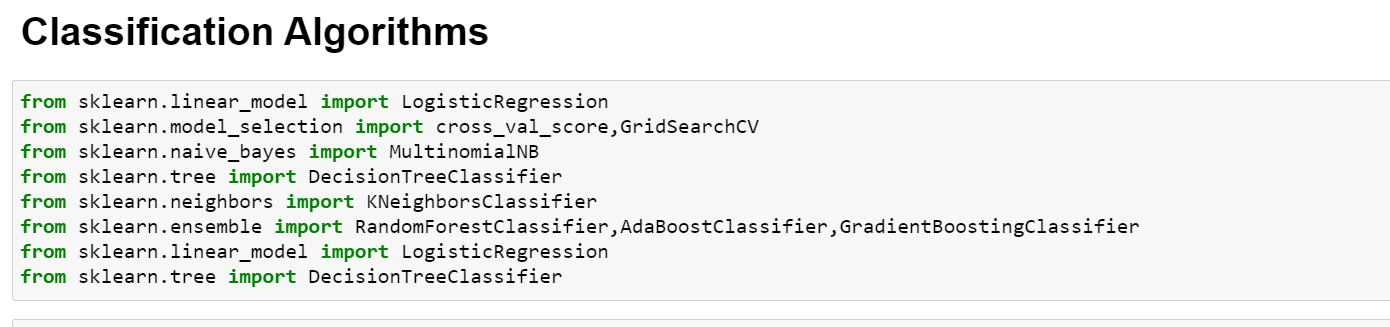
Matplotlib - The library helps to generate histograms, plots, error charts, scatter plots, bar charts with just a few lines of code.

Seaborn – Used for visualization.

train\_test\_split: We use it to perform test train spliting.

Algorithm Libraries:

1. Classifications Algorithm Libraries:
2. Logistic Regression Classifier
3. Random Forest Classifier
4. Decision Tree Classifier
5. Ada Boost Classifier



**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

Classifying comments is a difficult task because there is little structure and a large variety of words used. Considering the challenges of multi-label classification and working with a large quantity of messy, unbalanced data, we are satisfied with our testing results.

Thus we’ll be using classification algorithms to predict the output dataset.

Split Training Data into Train-Set and Validation-Set.

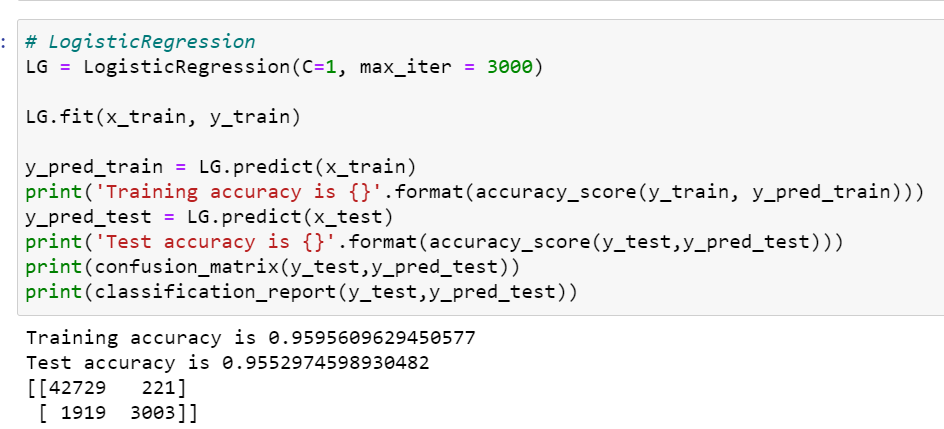
Since we have completed the data pre-processing and feature engineering part of our project, we move on to the model creation and model assessment part of the project.

* Testing of Identified Approaches (Algorithms)

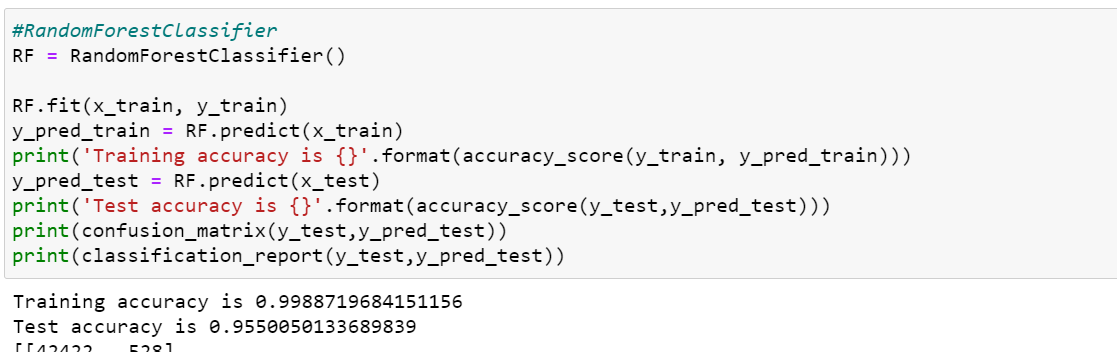
After performing all pre-processing we’ll proceed to classification algorithm.

Following are the algorithms implemented to get the best model:

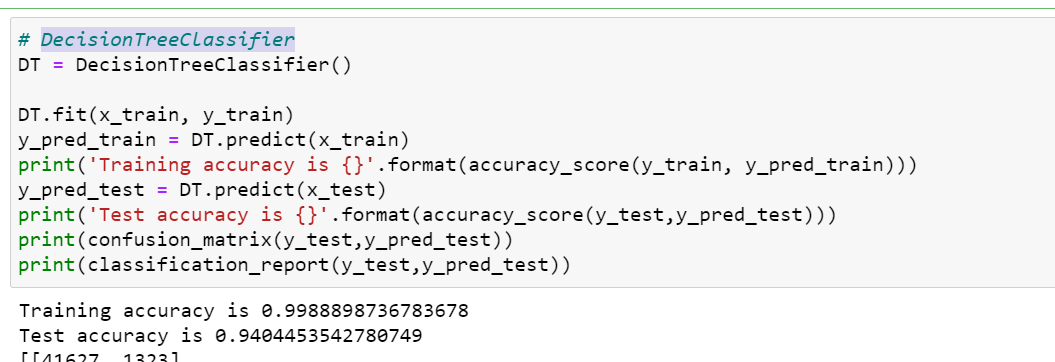
* Logistic Regression Classifier:



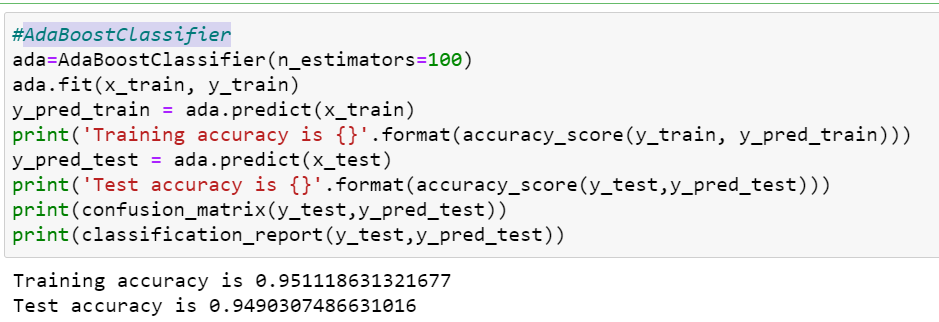
* Random Forest Classifier



* Decision Tree Classifier



* Ada Boost Classifier



* Run and Evaluate selected models

After performing all classifier algorithms we came to analyse that Random Forest Classifier is giving best accuracy as model. Thus, to train the train set and get the predicted output we’ll be using Random Forest Classifier.

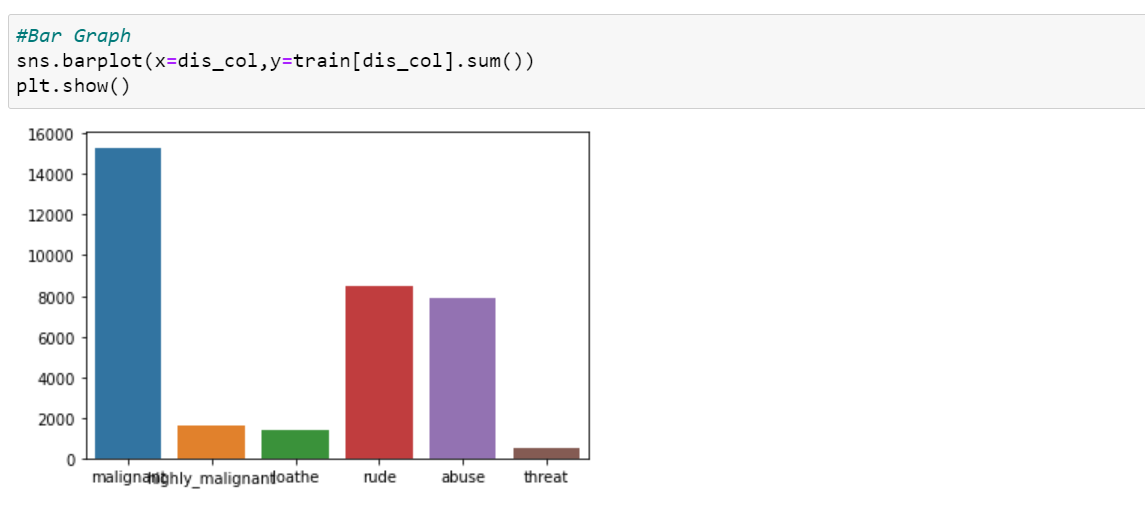
* Key Metrics for success in solving problem under consideration

For each one of the Classification techniques mentioned in the previous section (Linear Regression Classifier, Random Forest Classifier, Decision Tree Classifier, Ada Boost Classifier etc.), we will follow these steps to build a model:

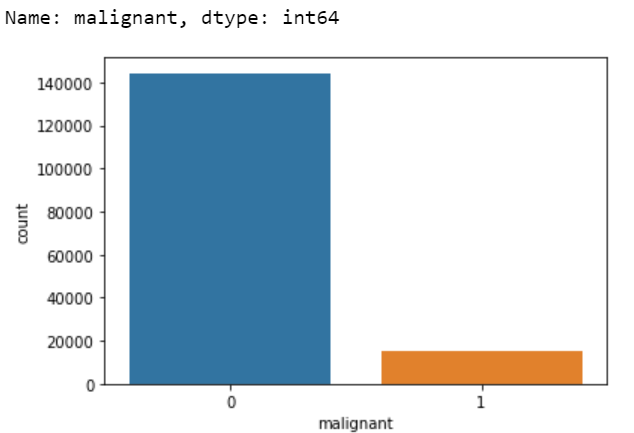
* Choose an algorithm that implements the corresponding technique
* Search for an effective parameter combination for the chosen algorithm
* Create a model using the found parameters
* Train (fit) the model on the training dataset
* Test the model on the test dataset and get the results

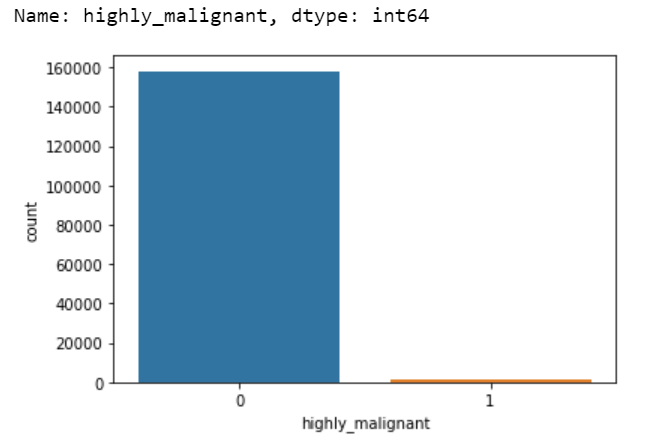
Visualizations

From the first visualisation we can observe that all comments count between 2000 to 16000. The highest number of comments are of ‘malignant’ label.



The second visualisation plots the number of comments belonging to various categories. Toxic comments were highest in number, followed ‘Malignant’, ‘Highly malignant’, ‘Rude’, ‘Threat’, ‘Abuse’ and ‘Loathe’ in decreasing order.

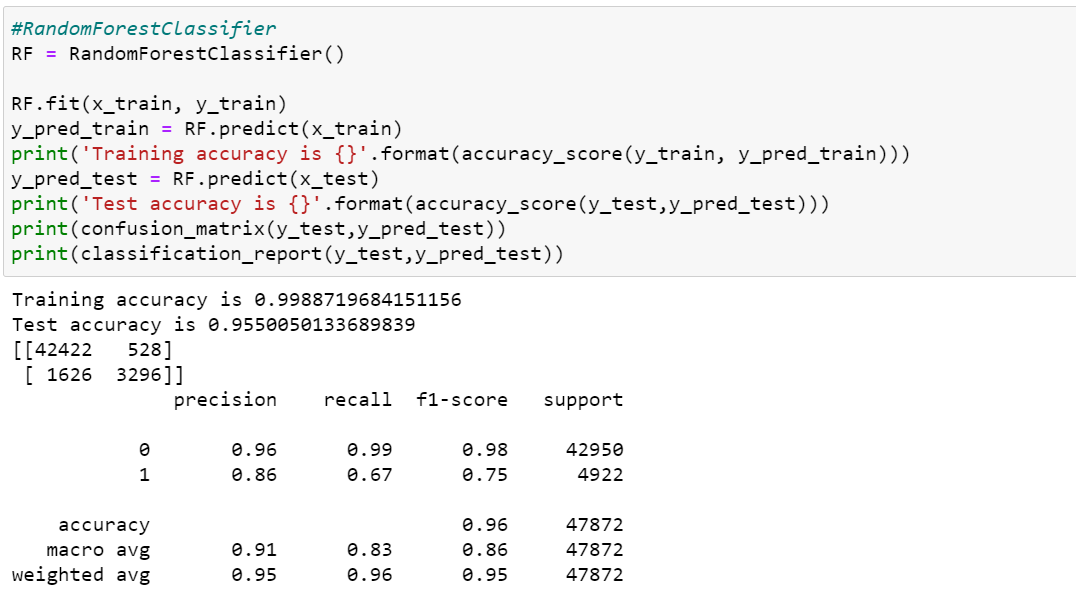




This analysis gave me some really good insights about the distribution of my data. The next step was to perform pre-processing of the data. The volume of the data available was fair enough for good analysis but not easy enough to deal with.

* Interpretation of the Results

After performing data pre-processing, Data Visualization and Modelling we have chosen Random Forest Classifier as a best model giving accuracy of to predict the test dataset.



**CONCLUSION**

* Key Findings and Conclusions of the Study

After performing all models on this malignant comments we found out that from all performed classifier models Random Forest Classifier seems to be best model. This gives us a fairly decent idea about the quality of our deep-learning model, and whether it has been appropriately trained.

* Learning Outcomes of the Study in respect of Data Science

This project allowed me to work with two different deep learning models and additionally, I was able to implement them on a Natural Language Processing use-case. The various data pre-processing and feature engineering steps in the project made me cognizant of the efficient methods that can be used to clean textual data. Along with this we came to know about the malignant comments prediction rate and its outcomes that can affect the audience. Resulting in ‘Random Forest Classifier’ as best model with accuracy of 96%.

* Limitations of this work and Scope for Future Work

Our model also provides beneficence for the platform hosts as it replaces the need to manually moderate discussions, saving time and resources. Employing a machine learning model to filter comments promotes justice, because all comments will be processed on an equal footing.

The motivating principle behind our project is promoting nonmaleficence within online communities by identifying harmful comments and acting against them. This is primarily experienced by those who prefer a safe and productive environment without negative distractions.