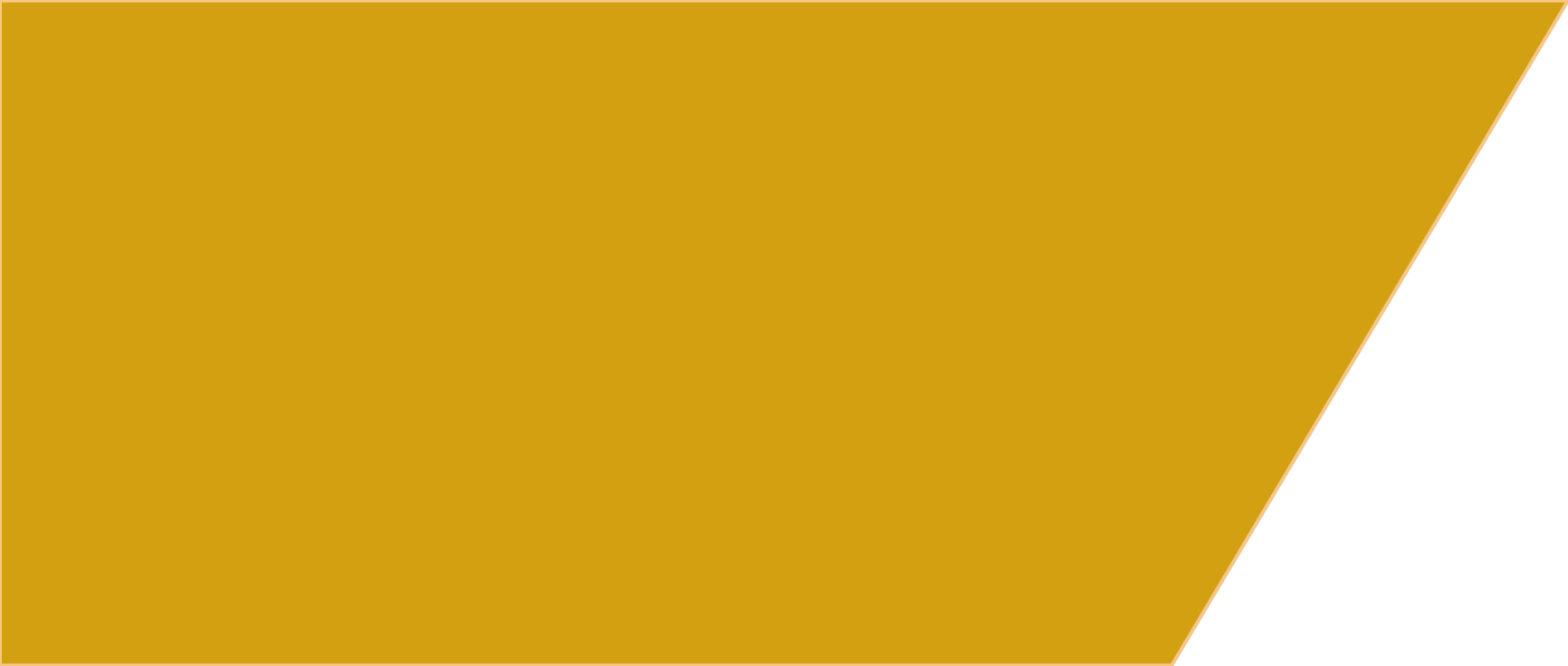
|  |
| --- |
| MALIGNANT COMMENTS CLASSIFICATION |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| SAURABH UPADHYAY |  |





MALIGNANT COMMENTS

CLASSIFICATION

SUBMITTED BY

SAURABH UPADHYAY

## ACKNOWLEDGEMENT

First of all, I extol the Almighty for pouring his blessings on me and giving me potentiality and opportunity to carry the work to its end with a success.

“It’s impossible to prepare a project report without the help and fillip of some people and certainly this project report is of no exception.”

At the commencement of this project report I would like to evince my deepest sense of gratitude to Ms. Astha Mishra, my honored mentor. Without her guidance, insightful decision, valuable comments and correction it would not have possible to reach up to this mark.

I would like to draw my gratitude to Flip Robo and Data Trained for providing me a suitable environment and guidance to complete my work. Last but not least thanks to the brilliant authors from where I have got the idea to carry out the project.

References were taken from various articles from Medium, KDnuggets, Towards Data Science, Machine Learning Mastery, Analytics Vidya, American Statistical Association, Research Gate and documentations of Python and Sklearn.

Saurabh Upadhyay

CONTENTS

|  |  |
| --- | --- |
| ACKNOWLEDGEMENT | 02 |
| LIST OF FIGURES & TABLES | 04-05 |
| CHAPTER 1: INTRODUCTION | 07-09 |
| CHAPTER 2: ANALYTICAL PROBLEM FRAMING | 10-22 |
| CHAPTER 3: MODEL DEVELOPMENT | 23-31 |
| CHAPTER 4: CONCLUSION | 32 |

LIST OF FIGURES AND TABLES

|  |  |
| --- | --- |
| FIGURES | PAGE NO |
| Table.1 METADATA | 10 |
| Fig.1 TRAINING DATASET | 11 |
| Fig.2 SHAPE AND COLUMNS | 12 |
| Fig.3 DUPLICATED VAKUES | 12 |
| Fig.4 NULL VALUES | 12 |
| Fig.5 HEATMAP OF NULL VALUES | 13 |
| Fig.6 DROPPING COLUMNS | 14 |
| Fig.7 CREATING NEW LABEL | 14 |
| Fig.8 RAW STRING LENGTH LABEL COLUMN | 15 |
| Fig.9 VALUE COUNT | 15 |
| Fig.10 COUNT OF LABEL WITH YES CONDITION | 16 |
| Fig 11: COMMENTS WITH MORE THAN ONE CATEGORIES | 16 |
| Fig.12 INFO | 17 |
| Fig.13 STOPWORDS | 18 |
| Fig.14 FUNCTION FOR PROCESSING | 18 |
| Fig.15 SAMPLE TESTING | 18 |
| Fig.16 TEST RESULT | 19 |
| Fig.17 FUNCTION IMPLEMENTTION ON TRAINING DATA | 19 |
| Fig.18: DATASET AFTER PROCESSING | 20 |
| Fig.19 REDUCTION DUE TO FUNCTION | 20 |
| Fig.20: FUNCTION ON TEST DATASET | 21 |

|  |  |
| --- | --- |
| Fig.21 X Y SPLIT | 22 |
| Fig.22 TFIDF VECTORIZER | 22 |
| Table 2. HARDWARE & TOOLS USED | 22 |
| Fig 23. TRAIN TEST SPLIT | 23 |
| Fig 24. LOGISTIC REGRESSION PIPELINE | 25 |
| Fig 25. LOGISTIC REGRESSION WITH MALIGNANT LABEL | 26 |
| Fig 26. RESULTS | 27-28 |
| Fig 27. PIE PLOT | 29 |
| Fig 28. WORDCLOUDS | 29-30 |
| Fig 29. MODEL SAVING | 30 |
| Fig 30. TEST DATASET | 31 |
| Fig 31. MODELLING OF TEST DATASET | 31 |

CHAPTER-1

INTRODUCTION

* 1. BUSINESS PROBLEM

Social media has given a lot of things to people which beyond imagination. In this era of technology, it has become the hub of information. The numbers of contents on social media are vast and rich and everything has found a place on social media that may be anything. It has given wings to its users to fly high and express their feelings. It has become a boon for the mankind but we all know that if there is good there must be some bad. Likewise, social media has also got the dark side.

I would like to quote Tarana Burke who once told that “Social media is not a safe space.” It is absolutely true even though it has given a lot of things to the mankind it has also taken it toll. Now a days it is becoming a weapon to create disturbance in the society and personal life of people. Everyday the count of incidents of Online hate is increasing. So to face this problem effectively a machine learning model will be created. 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.

1.2 BACKGROUND

In the past few years its seen that the cases related to social media hatred have increased exponentially. The social media is turning into a dark venomous pit for people now a days. Online hate is the result of difference in opinion, race, religion, occupation, nationality etc.

In social media the people spreading or involved in such kind of activities uses filthy languages, aggression, images etc. to offend and gravely hurt the person on the other side. This is one of the major concerns now.

The result of such activities can be dangerous. It gives mental trauma to the victims making their lives miserable. People who are not well aware of mental health online hate or cyber bullying become life threatening for them. Such cases are also at rise. It is also taking its toll on religions. Each and every day we can see an incident of fighting between people of different communities or religions due to offensive social media posts.

Online hate, described as abusive language, aggression, cyberbullying, hatefulness, insults, personal attacks, provocation, racism, sexism, threats, or toxicity has been identified as a major threat on online social media platforms. These kinds of activities must be checked for a better future.

1.3 MOTIVATION

The project was the first provided to me by FlipRobo as a part of the internship programme. The exposure to real world data and the opportunity to deploy my skillset in solving a real time problem has been the primary objective. However, the motivation for taking this project was that it is relatively a new field of research. Here we have many options but less concrete solutions. The main motivation was to classify the news in order to bring awareness and reduce unwanted chaos and make a good model which will help us to know such kind of miscreants.

CHAPTER-2

ANALYTICAL PROBLEM FRAMING

2.1 ANALYTICAL MODELING OF PROBLEM

Anyone can be a victim of online hate or cyberbully. The social media has become a dangerous place to dwell in. The use of abusive language, aggression, cyberbullying, hatefulness, insults, personal attacks, provocation, racism, sexism, threats, or toxicity have significantly high negative impact on individual. We can use Machine Learning and NLP technologies to deal with such toxic comments.

We were provided with two different datasets. One for training and another to test the efficiency of the model created using the training dataset. The training dataset provided here has a shape of 159571 rows and 8 columns. As it is a multiclass problem it has 6 dependent / target column. Here the target or the dependent variables named “malignant, highly\_malignant, rude, threat, abuse, loathe” have two distinct values 0 and 1. Where 1 represents yes and 0 represents no for each class. As the target columns are giving binary outputs and all the independent variables has text so it is clear that it is a supervised machine learning problem where we can use the techniques of NLP and classification-based algorithms of Machine learning.

Here we will use NLP techniques like word tokenization, lemmatization, stemming and tfidf vectorizer then those processed data will be used to create best model using various classification based supervised machine learning algorithms like Logistic Regression, Passive Aggressive Classifier, Multinomial NB, Complement NB with the help of OneVsRestClassifier which is helpful to deal with multilabel classification problems.

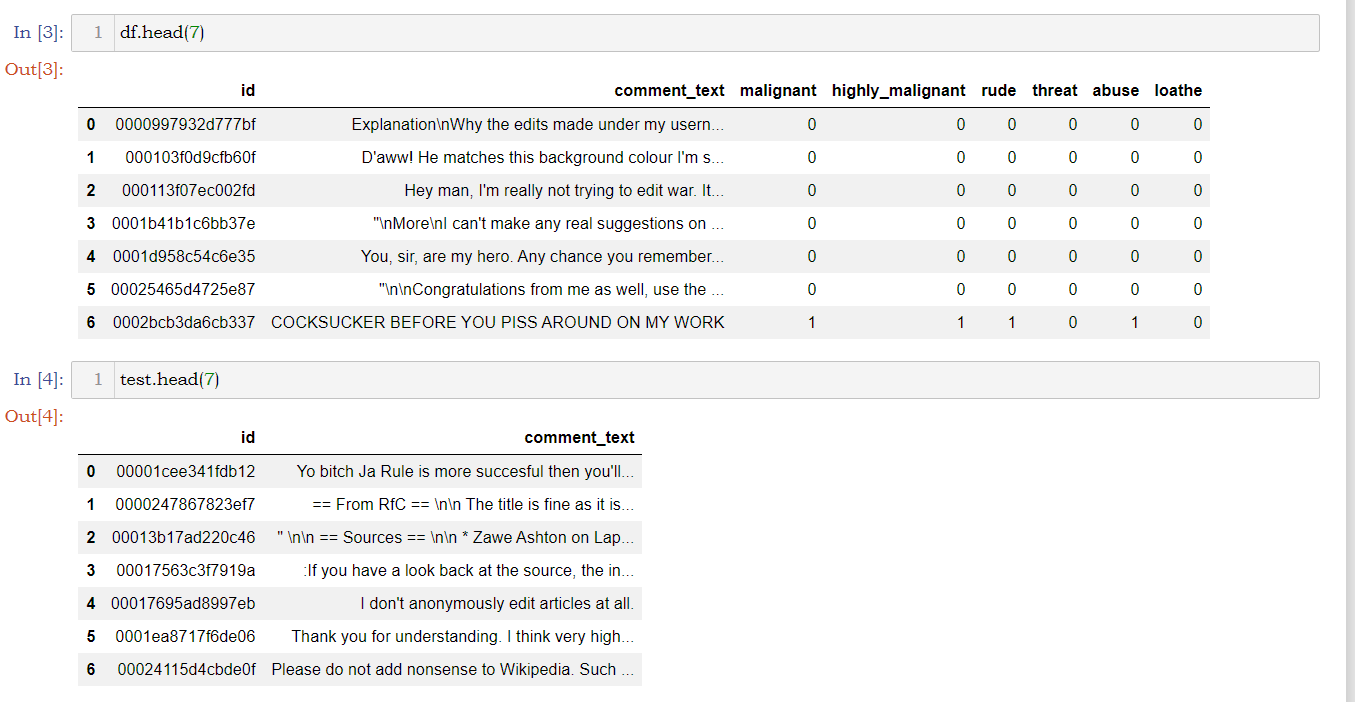
The passive Aggressive Classifier belongs to the family of online machine learning algorithms and it is very much helpful in processing large scale data. It remains passive for a correct classification and turns aggressive in case of a misclassification. Its aim is to make updates that corrects the loss causing very little change in the weight vector.

2.2 DATA SOURECE AND FORMATS

The data was provided by FlipRobo in CSV format. After loading the training dataset into Jupyter Notebook using Pandas and using df.head() **[Fig. 1]** it can be seen that there are eight columns named as “ *id, comment\_text,* “malignant, highly\_malignant, rude, threat, abuse, loathe”. Similarly the test file can be load using pandas and the first five rows of the dataset can be seen using df.head() method [**Fig. 1**].The metadata table is provided below for better understanding of the data given [**Table 1**].

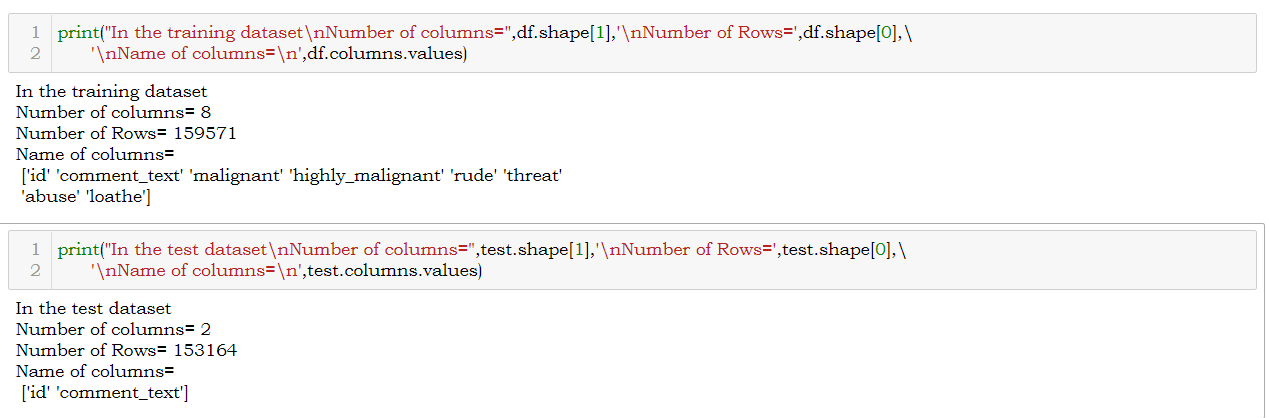
|  |  |  |
| --- | --- | --- |
| INDEPENDENT | ID | DENOTES THE UNIQUE IDS ASSOCIATED WITH EACH COMMENT GIVEN. |
| COMMENT\_TEXT | DENOTES COMMENTS EXTRACTED FROM VARIOUS SOCIAL MEDIA PLATFORMS. |
| DEPENDENT | MALIGNANT | IT DENOTES IF THE COMMENT ARE MALIGNANT OR NOT. |
| HIGHLY\_MALIGNANT | IT DENOTES COMMENTS THAT ARE HIGHLY MALIGNANT. |
| RUDE | IT DENOTES COMMENTS THAT ARE VERY RUDE AND OFFENSIVE. |
| THREAT | IT CONTAINS THE COMMENTS THAT POSE A TO SOMEONE. |
| ABUSE | IT IS FOR COMMENTS THAT ARE ABUSIVE IN NATURE. |
| LOATHE | IT DESCRIBES THE COMMENTS WHICH ARE HATEFUL AND LOATHING IN NATURE. |

**(Table 1: METADATA)**

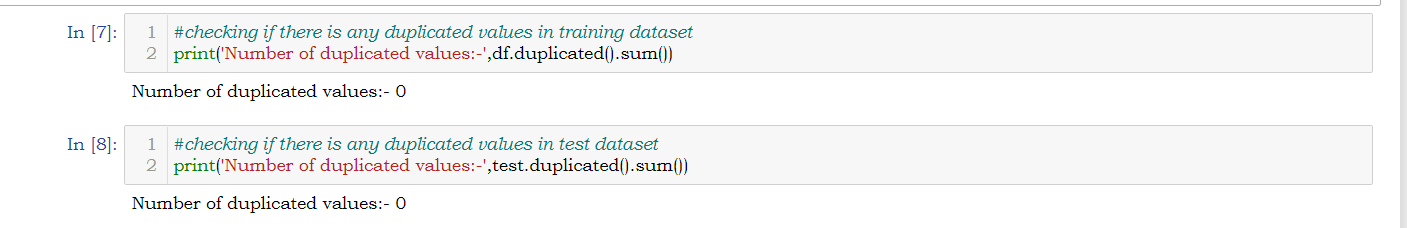


**(Fig 1. TRAINING DATASET)**

As mentioned earlier the shape of the training dataset is (159571, 8) and the shape of test dataset is (153164,2). The shape of the datasets in form of a tuple can be accessed using df.shape(). The column names of the datasets in form of a list can be seen using df.columns.values() [Fig 2]. The datasets have no duplicated values or null values. Both the datset have no trace of any null or duplicated values. The number of duplicated values of a dataset can be seen using df.duplicated().sum() [Fig 3] and the null values can be seen using df.isnull().sum() [Fig 4]. The null values can also be visualized with help of seaborn and matplotlib library [Fig 5]. Visulization gives a better idea .



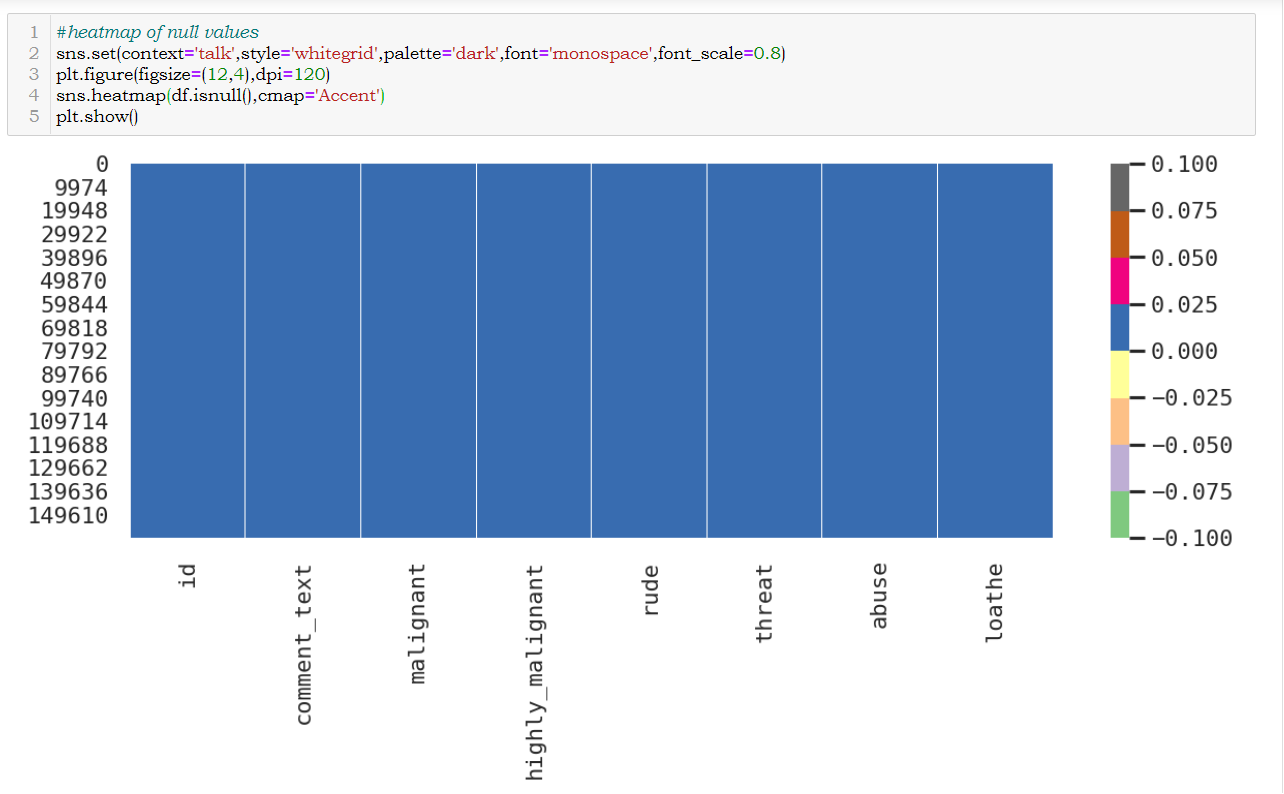
(Fig 2. SHAPE AND COLUMNS)



(Fig 3. DUPLICATED VALUES)



(Fig 4. NULL VALUES)



(Fig 5. HEATMAP OF NULL VALUES)

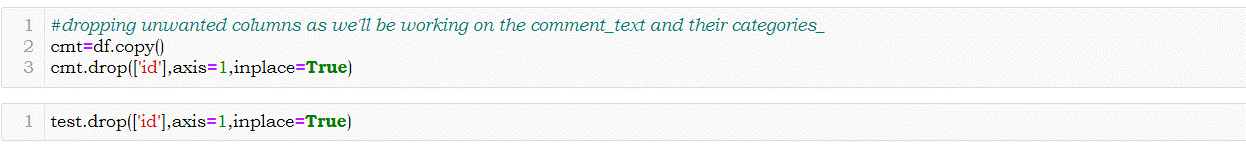
2.3 DATA PREPROCESSING

After the dataset is loaded and the shape, null values and duplicated values were checked then the data- set is further treated where the unwanted column “*id”* isremoved from the training dataset as we will work on the columns like “comment\_text, “malignant, highly\_malignant, rude, threat, abuse, loathe*”.* So a copy of the training dataset was made using df.copy() and the column was dropped from the new dataset using df.drop(). Similarly, the ‘id’ column is also dropped from the test dataset. [Fig 6].

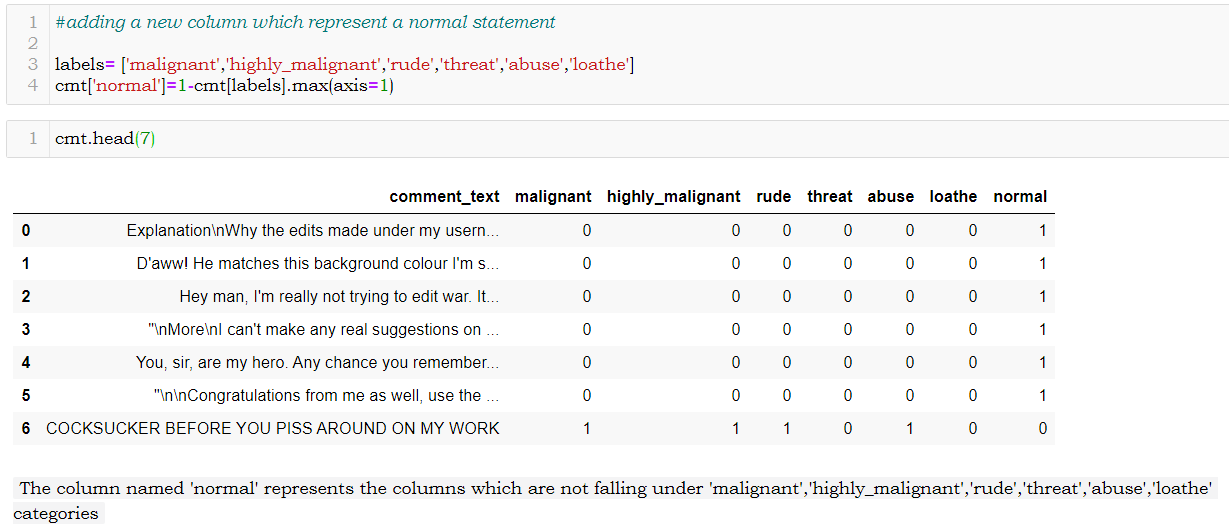
After removing the unwanted column, a new column named ‘normal’ was created in the training dataset which represents the statements not falling under malignant, highly\_malignant, rude,

threat, abuse, loathe category or statements where values of malignant, highly\_malignant, rude, threat, abuse, loathe are 0 [Fig 7].

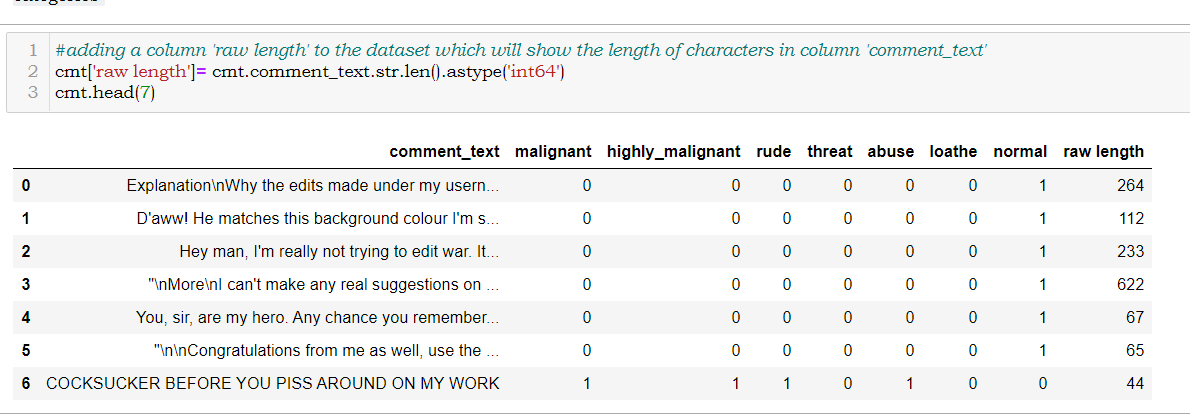
After the new column ‘normal’ was added and unwanted column ‘id’ was dropped a new column named “raw length” representing the string length of the ‘comment\_text’ column is added to the dataset [Fig 8]. It’ll help to know the length of the strings in ‘comment\_text’ columns before preprocessing and later a new column will be created to compare the length of strings before and after preprocessing



(Fig 6. DROPPING COLUMN)

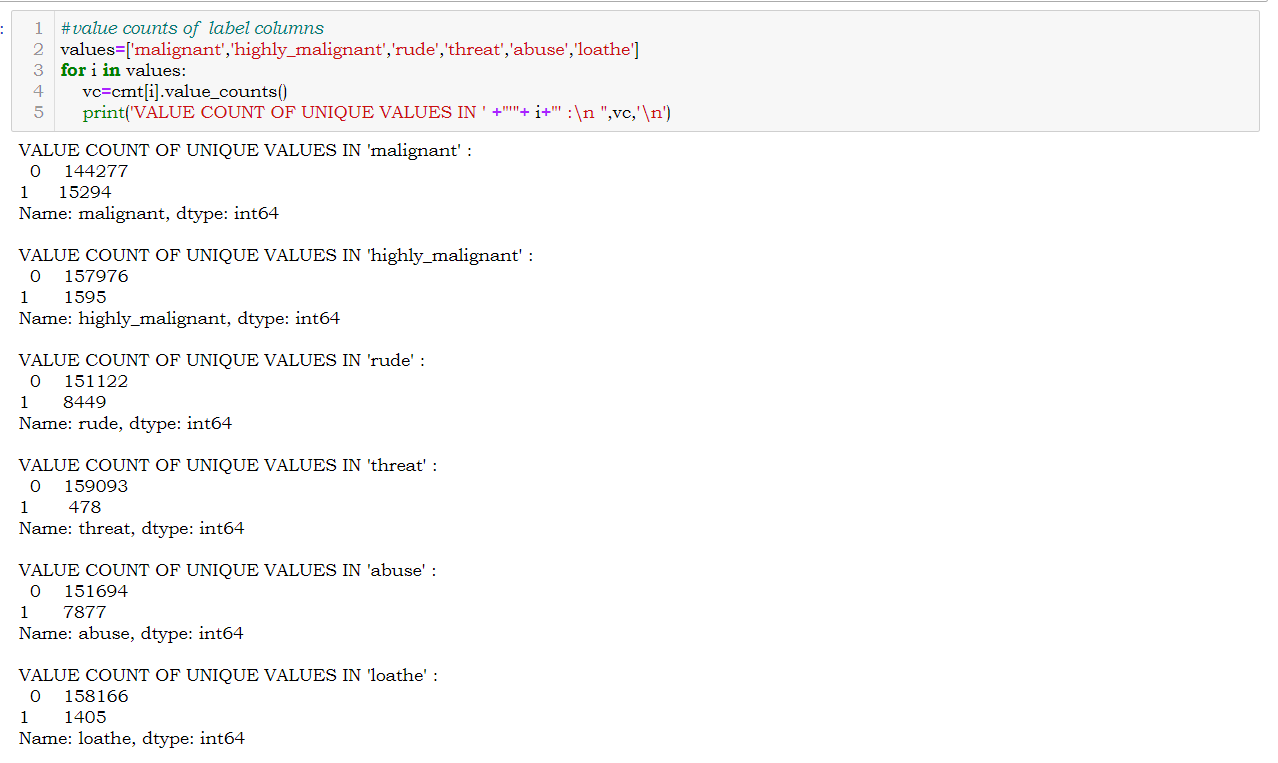


(Fig 7. CREATING NEW LABEL)



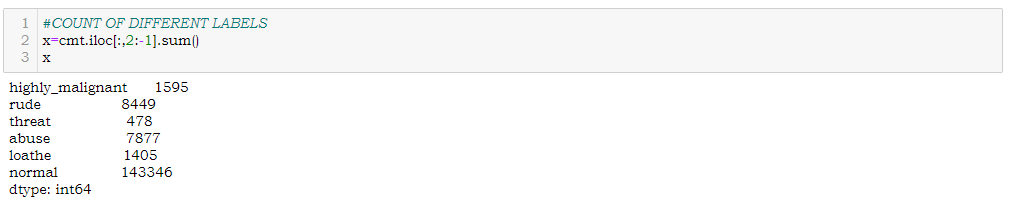
(Fig 8. RAW STRING LENGTH)

After that we can check the which label carries how many comments using df.value\_count() method [Fig 9]. It’ll briefly show the count of numbers of 0 and 1 of all dependent columns / target columns.



(Fig 9. VALUE COUNT)

We can also check the count of 1 (yes case) for each label which will show the number of malignant, highly\_malignant, rude, threat, abuse, loathe, normal comments [Fig 10]. It can be seen that there are comments which represents more than 1 category of labels this can also be checked and it’ll be helpful for more understanding [Fig 11].

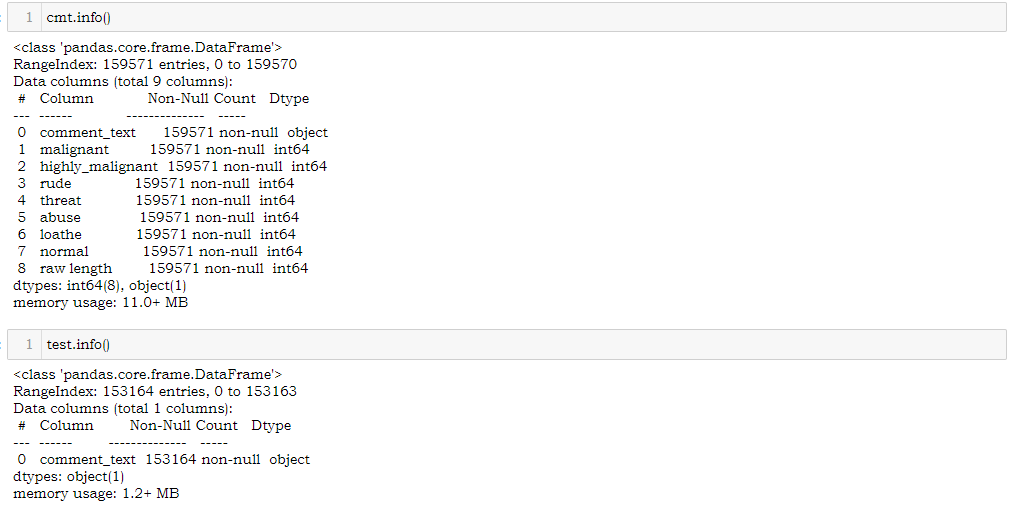


(Fig 10. COUNT OF LABELS WITH 1)



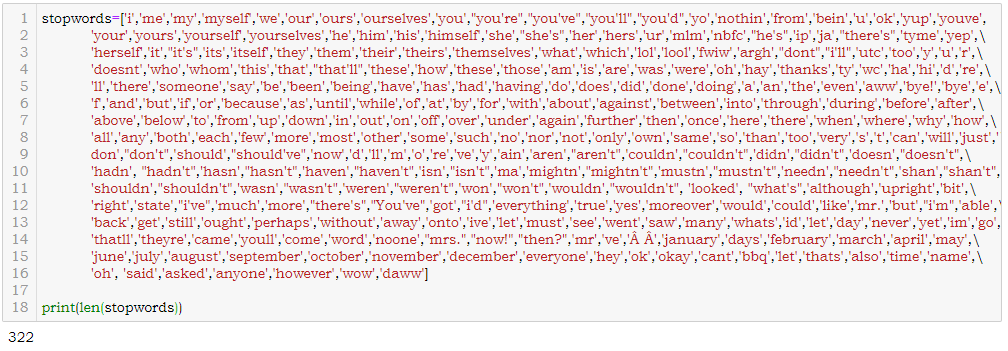
(Fig 11. COUNT OF COMMENTS FALLING UNDER MORE THEN ONE CATEGORIES)

After the columns to show anew category and to show the raw length of strings were created and unnecessary column ‘id’ was dropped the df.info() was used to get the detailed summary of the training and test dataset [Fig 12].

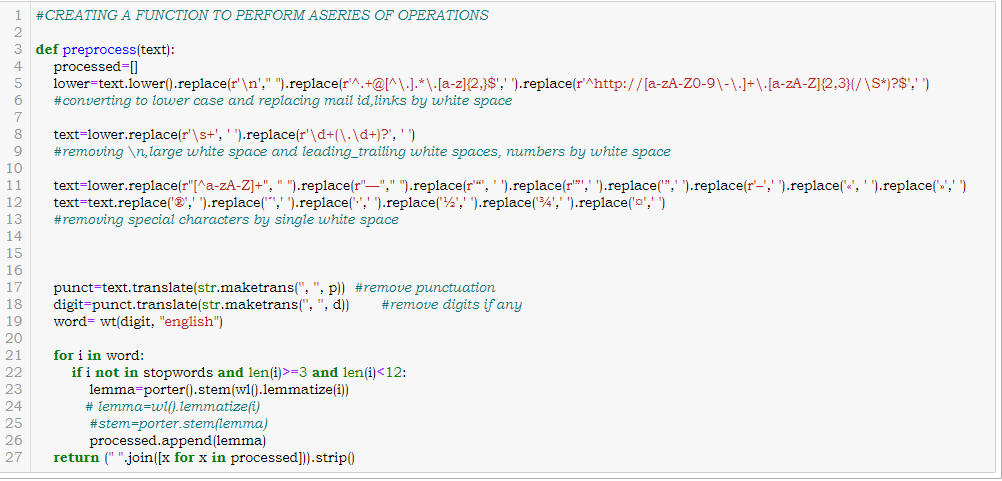


(Fig 12. INFO OF TEST & TRAIN DATASET)

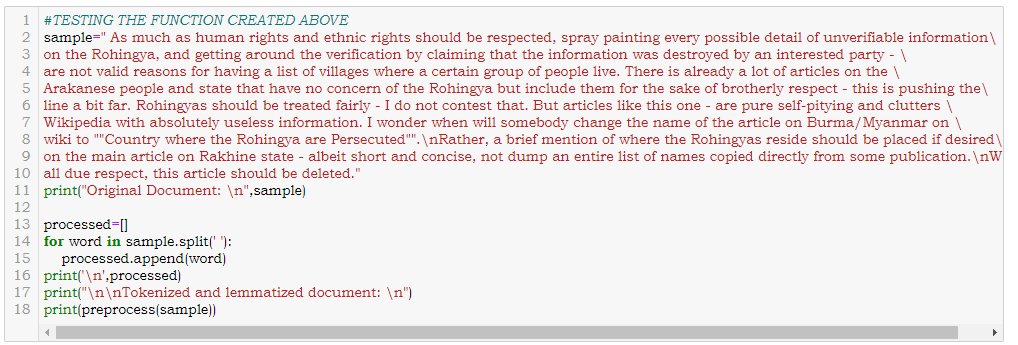
After the basic EDA is done the NLP techniques were implemented for processing the texts in the ‘comment\_text’ columns. For this process a list of stopwords were created manually [Fig 13]. In the preprocessing the string converted to lower case as it is easier to understand for the machine then from the strings the stopwords, special characters, digits were dropped using proper techniques. After those unnecessary characters were removed the string is tokenized using word\_tokenization() function of NLTK library then those tokenized words were checked for stopwords and token length of 3. If both the condition were satisfied the tokenized words were lemmatized and stemmed using wordnetLemmatizer() and PorterStemmer() which brings back all words to their root form. Then again, those tokenized words were joined to form a string. All these operations were compiled inside a function [Fig 14]. After the function was created a test run was done on a sample text to check the effectiveness of the function so created [Fig 15-16]. After successful testing the entire ‘comment\_text’ column was processed using the function created to get a clear and pure form of data for further operations [Fig 17].



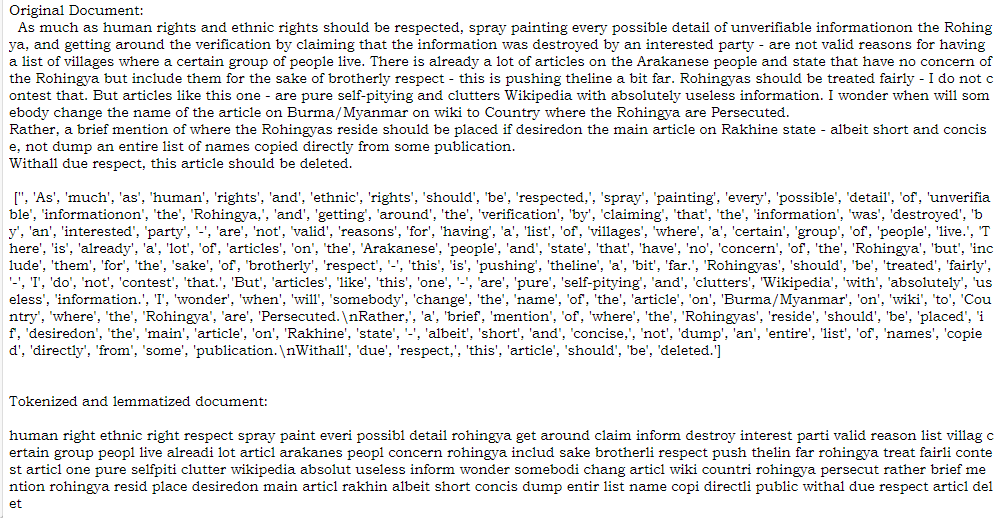
(Fig 13. STOPWORDS)



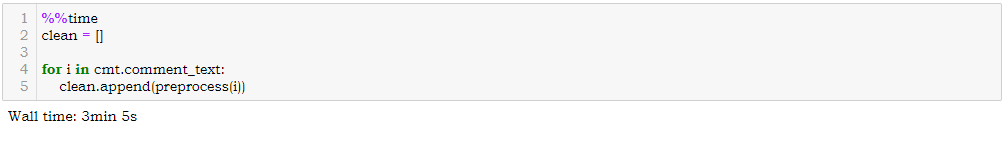
(Fig 14. FUNCTION FOR PROCESSING DATA)



(Fig 15. SAMPLE TESTING)

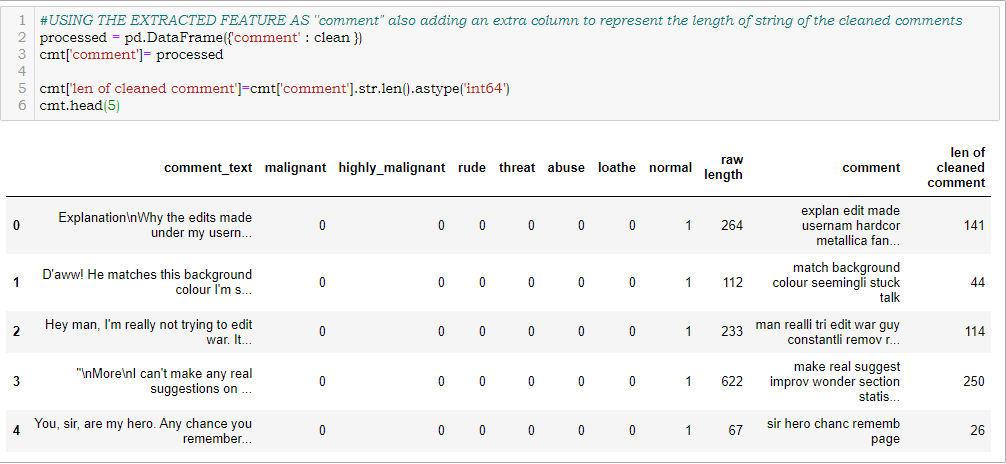


(Fig 16. TEST RESULTS)



(Fig 17. FUNCTION IMPLEMENTATION)

After the procedure was completed a list of cleaned data were obtained which was added to the dataset by column name ‘comment’ and another column name ‘len of clean comment’ was added showing the length of words in the ‘comment’ column **[Fig 18]**. Further calculation revelled that there were total of 62893130 words were present in the raw ‘comment\_text’ column and after processing it became 29977753 the pre-processing led to a reduction of 32915377 strings [**Fig 19**].



**(Fig 18. DATASET AFTER FUNCTION)**



(Fig 19. REDUCTION DUE TO FUNCTION)

Similar step was used on test data. The dataset was processed using the function created and after the processing is done the processed list added to the test dataset as a new column named ‘comment’ [Fig 20].



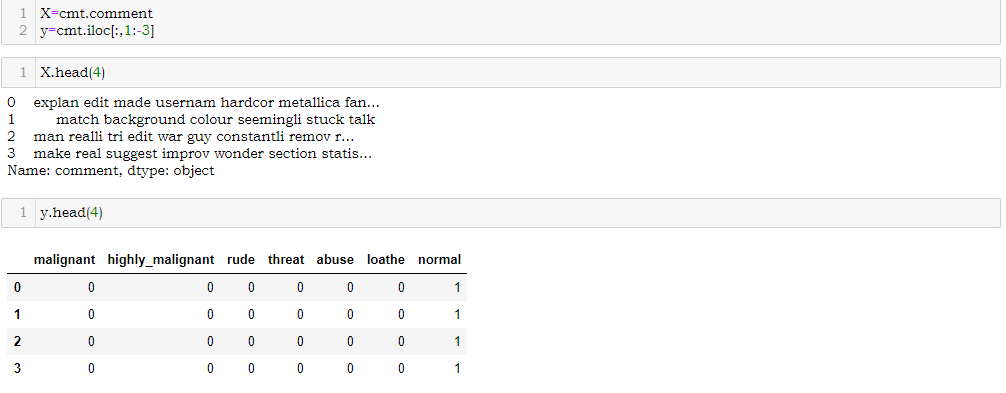
(Fig 20. FUNCTION ON TEST DATASET)

After getting a cleaned data TF-IDF vectorizer will be used. It’ll help to transform the text data to feature vector which can be used as input in our modelling. The TFIDF stands for Term Frequency Inverse Document Frequency. It is a common algorithm to transform text into vectors or numbers. It measures the originality of a word by comparing the frequency of appearance of a word in a document with the number of documents the words appear in.

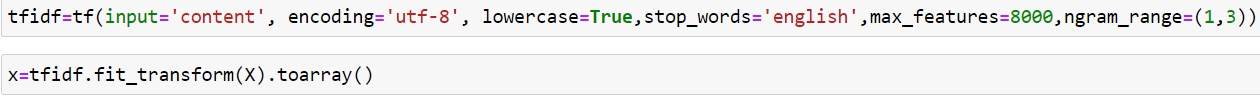
Mathematically,

TF-IDF =TF(t\*d) \* IDF (t,d)

So here the dataset is divided into two parts X and Y. X represents the column ‘comment’ which carries the cleaned text and Y represents the labels like ‘malignant, highly\_malignant, rude, threat, abuse, loathe, normal’ [Fig 21]. After the splitting the tfidf vectorizer was initialized and X is fitted into it and converted into an array [Fig 22].



(Fig 21. X Y SPLITING)



(Fig 22. TFIDF VECTORIZER)

2.4 HARDWARE & TOOL USED

In this project the below mentioned Hardware, IDE, Language, Packages were used.

|  |  |
| --- | --- |
| HARDWARE | LAPTOP: ASUS TUF A17  OS: WIN 10 HOME BASIC  PROCESSOR: AMD RYZEN 7 4800H  RAM: 16GB  VRAM: 6GB NVIDIA GTX 1660Ti |
| LANGUAGE | Python 3.8 |
| IDE | JUPYTER NOTEBOOK 6.0.3 |
| PACKAGES | PANDAS, NLTK, SKLEARN, MATPLOTLIB, SEABORN |

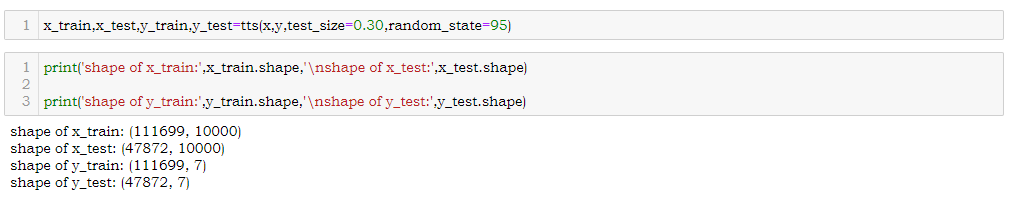
**(Table 2: HARDWARE & TOOLS)**

CHAPTER-3

DEVELOPMENT AND EVALUTION

3.1 IDENTIFACTION OF POSSIBLE PROBLEM-SOLVING APPROACHES

After TF-IDF implementation array conversion we have x and y for modelling. Then x and y were split for training and testing using train\_test\_split in a ratio of 70:30 respectively. After split the shape of x\_train and x\_test found to be (111699,10000) and (47872, 10000) and y\_train and y\_test found to be (111699,7) and (47872,7) respectively [Fig 23].



(Fig 23. TRAIN TEST SPLIT)

3.2 TESTING OF IDENTIFIED ALGORITHMS

As it is a multi-label classification problem, we will use OneVsRestClassifier from sklearn with other classification algorithms like;

* Logistic Regression()
* Passive Aggressive Classifier()
* Multinomial NB()
* Complement NB()

During modelling various metrices like f1\_score, confusion matrix, accuracy score, classification report, roc curve, roc auc score, mean squared error, precision score, recall score, log loss will be used to determine the performance of the model. At each step at the end of a model a data frame will be generated which will show the performance of the model per class. This will be executed with the help of pipe line.

3.3 TESTING OF IDENTIFIED ALGORITHMS

Here for each model I have created a number of list named as F1, ACCURACY, PRECISION, RECALL, RMSE, MSE, AUC, LOG\_LOSS to hold the values of matrices like f1 scores, accuracy scores, precision values, recall values, root mean squared error values, mean squared error values, auc scores, tpr values, fpr values, cross validation with f1 values, log loss values respectively.

Here a pipeline has been created where the algorithm will run under OneVsRestClassifier. It’ll also show the confusion matrix, accuracy score, classification report, roc curve, auc, roc auc score, mean squared error, precision score, recall score, tpr, fpr, f1 score, log loss value along with AUC Curves and Heatmap of confusion matrix for each label. The values obtained will be added to their respective lists. Below are few images of function performed on algorithms. Showing the metrics, heatmap of confusion matrix, AUC ROC curve. Below is the image of the pipeline created to achieve the required metrics and graphs.

|  |
| --- |
|  |
|  |

(Fig 24. LOGISTIC REGRESSION PIPELINE)

|  |  |
| --- | --- |
| RESULT AT MALIGNANT LABEL | AUC CURVE AT MALIGNANT LABEL  HEATMAP OF CONFUSION MARIX AT MALGNANT |

(Fig 25. LOGISTIC REGRESSION RESULT WITH MALIGNANT LABEL)

3.4 METRICE OF EVALUATION

In the modeling I have chosen metrices like F1 score, Accuracy score, Precision, Recall, Mean Squared Error, Root Mean Square Error as my evaluation criteria. All the values were stored in a list and later they were saved in form of a DataFrame for proper evaluation and visualization of the values. Below are the result obtained using various algorithms with OneVsRestClassifier().

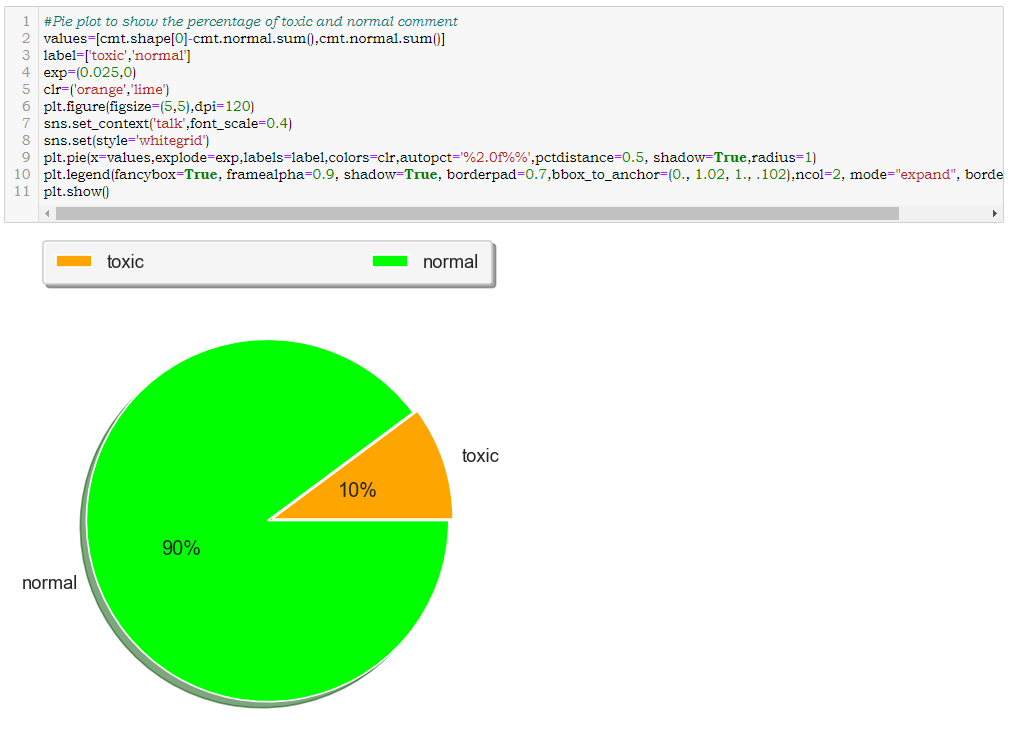
|  |
| --- |
| RESULTS USING LOGISTIC REGRESSION |
| RESULTS USING PASSIVE AGGRESSIVE CLASSIFIER |

|  |
| --- |
| RESULTS USING MULTINOMIAL NB |
| RESULTS USING COMPLEMENT NB |
| RESULTS USING LINER SVC |

(Fig 26. RESULTS)

3.5 VISUALIZATION

Visualization plays a crucial role in EDA as well as during modelling. It gives a better idea about the things going on beautifully. Below are the few visualizations used during this project to understand the dataset and performance of the algorithms.



(Fig 27. PIE PLOT OF NORMAL & HATE COMMENTS)

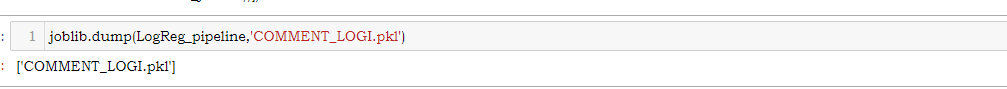
|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

(Fig 28. WORD CLOUDS)

3.6 INTERPRETATION

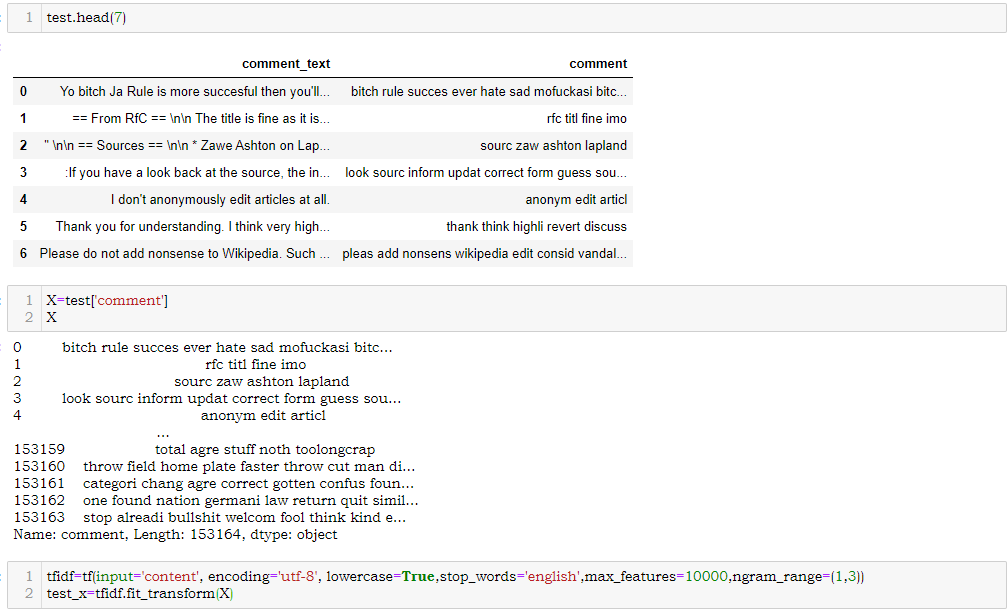
Basing on the result obtained ‘Logistic Regression’ have performed well and has given better result as compared to other models so it has been selected as final model and it will be saved using joblib library.



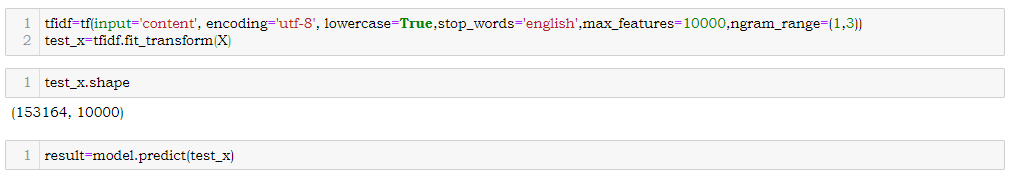
(Fig 29. SAVING MODEL)

3.7 TESTING

After the model is saved it was again load into the system by joblib.load() method along with a variable name. From the test dataset the processed column “comment” was then transformed into vectors using tfidf vectorizer and then the result was predicted for possible classes using the model load.



(Fig 30. CLEANED TEST DATASET)



(Fig 31. MODELLING OF TEST DATASET)

CHAPTER-4

CONCLUSION

4.1 KEY FINDINGS

From the above analysis the below mentioned results were achieved which depicts the

chances and conditions of a comment being a hateful comment or a normal comment;

* With the increasing popularity of social media, more and more people consume feeds from social media and due differences they spread hate comments to instead of love and harmony. It has strong negative impacts on individual users and broader society.

4.2 LEARNING OUTCOMES

* It is possible to classify the comments content into the required categories of authentic and However, using this kind of project an awareness can be created to know what is fake and authentic.

4.3 LIMITATION AND SCOPE OF WORK

Every effort has been put on it for perfection but nothing is perfect and this project is of no exception. There are certain areas which can be enhanced.Comment detection is an emerging research area with few public datasets. So, a lot of works need to be done on this field.