FLIP ROBO

Malignant comment classifier Project

Submitted by:
FENNY DENNY

ACKNOWLEDGMENT

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped me and guided me in completion of the project.

I wish to express my sincere gratitude to Miss. Khushboo Garg, SME for providing me an opportunity to do my internship and project work in “FLIP ROBO”.

It gives me immense pleasure in presenting this project report on “Micro Credit Defaulter Model”. It has been my privilege to have a team of project guide who have assisted me from the commencement of this project. The success of this project is a result of sheer hard work, and determination put in by me with the help of You Tube videos, references taken from Kaggle.com, skikit-learn.org.. To know more about micro finance, I read

https://www.geeksforgeeks.org/

https://github.com/

https://www.mckinsey.com/

https://www.counterpointresearch.com/

I hereby take this opportunity to add a special note of thanks for to Miss. Khushboo Garg, who undertook to act as my mentor despite his many other professional commitments. Her wisdom, knowledge and commitment to the highest standards inspired and motivated me. Without his insight, support this project wouldn’t have reached fruitfulness.

The project is dedicated to all those people of Fliprobo, Datatrained who helped me while doing this project.

INTRODUCTION

- Business Problem Framing:

We are required to model the comments classification malignant/highly-malignant with the available independent variables. This model will then be used by the management to understand to classify the comments based on input.

- Conceptual Background of the Domain Problem:

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.

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.

- Review of Literature: Now a days balancing an environment on social media platform is extremely important. Describing as “highly malignant and harmful passing through the medium of electronic text”, cyber bullying puts targets under attack from a barrage of degrading, threatening, and/or sexually explicit messages and images conveyed using web sites, instant messaging, blogs, chat rooms, cell phones, web sites, e-mail, and personal online profiles. Thus, the task of finding and removing toxic communication from social media forums is very crucial.

- Motivation for the Problem Undertaken:

This project helps me understand the toxic comments classification problem in social media platform, its customer comments. With the right set of datasets in hand I have built a model that helps the enterprise take the right decision that is whether to focus on a malignant/highly-malignant set of customers. This also motivate learn about text classification problem in social media platform in details. This project is more about exploration, feature engineering and classification that can be done on this data. Since the data set is huge and includes many categories of comments, we can do good amount of data exploration and derive some interesting features using the comments text column available.

We built a model that can differentiate between comments and its categories.

Analytical Problem Framing

- **Mathematical/ Analytical Modelling of the Problem**:
  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.

. I have used a Random forest classifier model to classify the comments in terms malignant/highly malignant and also used cross validation to remove overfitting problem while predicted the correct outcome and validate the model.

- **Data sources are provided internally by the enterprise.**

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.

- **Data Pre-processing**:

In the data pre-processing stage, I have found out if there is any missing data in dataset, for a particular column if there are any outliers present and how to handle the outliers. I have also found the total shape of the data set. I have also found out the dataset description using describe method. So, in this pre-processing process I have mainly cleansed the data and prepared the right set of data for further processing & for predicting the model.

- **Data Inputs- Logic- Output Relationships**:

  To find out the relationship between all the input variable I have used correlation function and find out whether there is a positive/negative relationship between a pair of variables. From this describe function that also known as Five-point summary analysis if there are any outliers are present for a particular column. Also five point summary analysis was done for the target variable to explore & understand the data in a better way.

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

  Since all the dataset provided and defined properly so in this dataset, I assume malignant/highly malignant as the target variable for this project. Rest of the parameters are used as input variables.

- **Hardware and Software Requirements and Tools Used:**

Data Science task should be done with sophisticated machine with high end machine configuration. The machine which I'm currently using is powered by intel core i5 processor with 8GB of RAM. With this above-mentioned configuration, I managed to work with the data set in Jupyter Notebook which help us to write Python codes. As I'm using low configuration machine so it took more time then usual to execute codes. The library used for the assignment are Numpy, Pandas, Matplotlib, Seaborn, Scikit learn

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods):

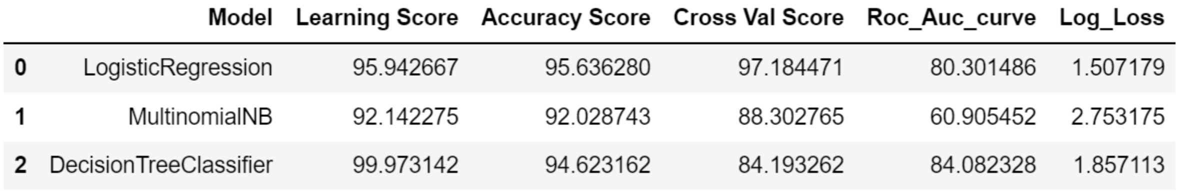
For this particular project I have used different classification models to predict the outcome of this dataset. After the model implementation Random forest classifier method predicted the best outcome out of all the models in terms of accuracy score and also I have used cross validation to flag the problem related overfitting or selection bias for the dataset and hence we can use this model for further evaluation.

Testing of Identified Approaches (Algorithms):

I have used mainly different classification methods to get the outcome of the house price prediction and 80% data used for training purpose and rest 20% are used for testing the prediction of the accuracy score for this machine learning model building process.

Run and Evaluate selected models:

To predict the result of this dataset below are machine learning models used for evaluations.



	Model	Learning Score	Accuracy Score	Cross Val Score	Roc_Auc_curve	Log_Loss
0	LogisticRegression	95.942667	95.636280	97.184471	80.301486	1.507179
1	MultinomialNB	92.142275	92.028743	88.302765	60.905452	2.753175
2	DecisionTreeClassifier	99.973142	94.623162	84.193262	84.082328	1.857113

Out of all the machine learning models used I have selected Logistic Regression model for further evaluation of this project.

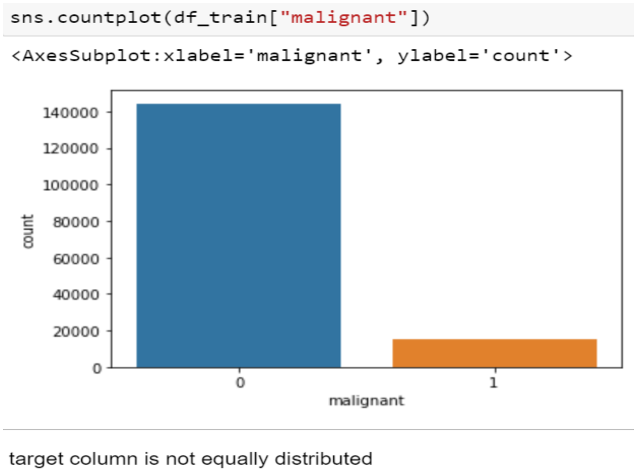
- **Key Metrics for success in solving problem under consideration**

The key metrics that were mainly taken into consideration were the followings:

- Comments_text
- malignant
- highly_malignant
- rude
- threat
- abuse
- loathe

These are all the prime metrics under consideration.

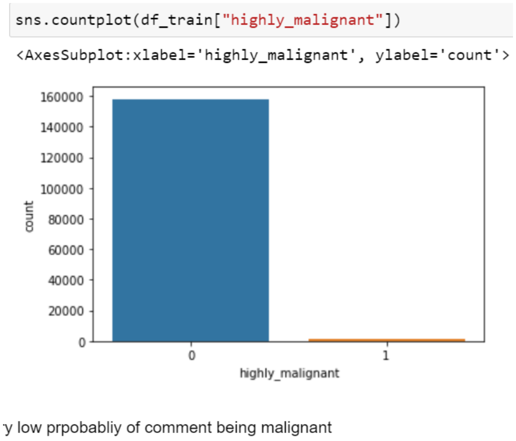
- **Visualizations:**



```
sns.countplot(df_train["malignant"])
```

```
<AxesSubplot:xlabel='malignant', ylabel='count'>
```

target column is not equally distributed



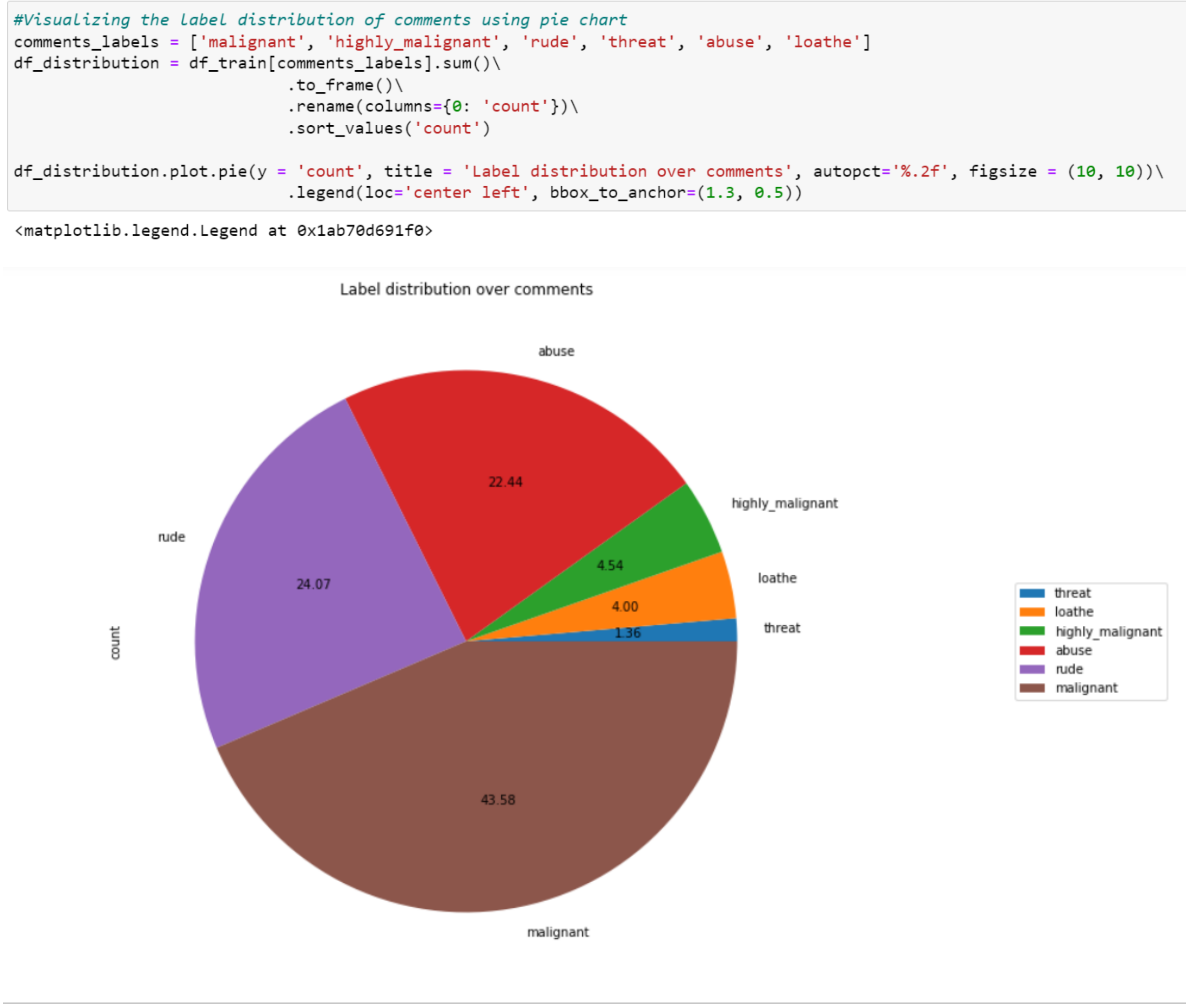
```
sns.countplot(df_train["highly_malignant"])
```

```
<AxesSubplot:xlabel='highly_malignant', ylabel='count'>
```

y low prpobabliy of comment being malignant

From the above graph we are getting the highly malignant count in the training dataset using the count plot under seaborn library.

Similarly, we have found out the count of **‘abuse’, ’threat’, ‘loathe’, ‘rude’** for this input parameters and visualize it using the count plot method under seaborn library.



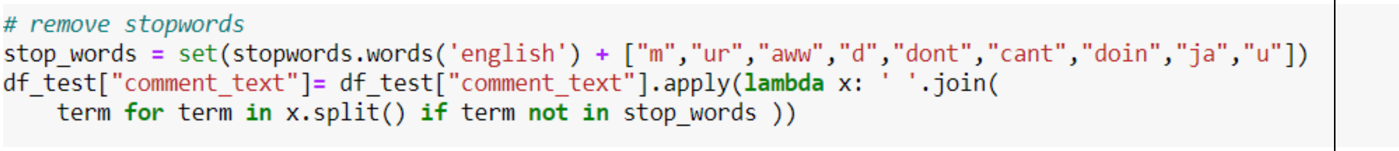
```
#Visualizing the label distribution of comments using pie chart
comments_labels = ['malignant', 'highly_malignant', 'rude', 'threat', 'abuse', 'loathe']
df_distribution = df_train[comments_labels].sum()\
                            .to_frame()\
                            .rename(columns={0: 'count'})\
                            .sort_values('count')

df_distribution.plot.pie(y = 'count', title = 'Label distribution over comments', autopct='%.2f', figsize = (10, 10))\
                            .legend(loc='center left', bbox_to_anchor=(1.3, 0.5))
```

```
<matplotlib.legend.Legend at 0x1ab70d691f0>
```

from sklearn.preprocessing import LabelEncoder le=LabelEncoder() df_train=df_train.apply(LabelEncoder().fit_transform)

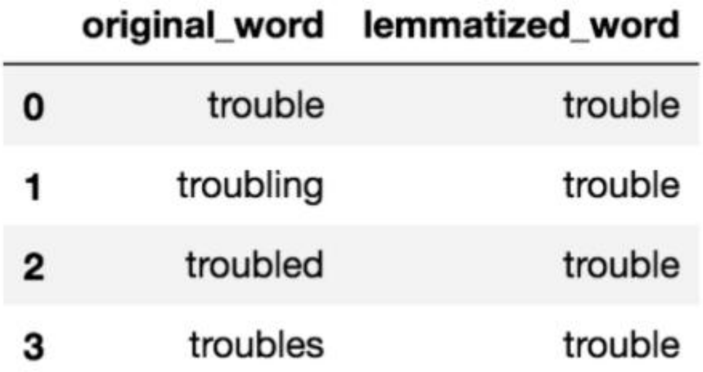
From the above distribution plot, we have found out how the input parameters like ‘**malignant’, ‘highly_malignant’, ‘rude’, ‘threat’, ‘abuse’, ‘loathe’** are distributed across the dataset using a pie chart

- **Stop word Removal:** *Stop words are those words that are frequently used in both written and verbal communication and thereby do not have either a positive/negative impact on our statement.* Stop words are a set of commonly used words in a language. Examples of stop words in English are “a”, “the”, “is”, “are” and etc. The intuition behind using stop words is that, by removing low information words from text, we can focus on the important words instead. For example, in the context of a search system, if your search query is “what is text processing?”, you want the search system to focus on surfacing documents that talk about text pre-processing over documents that talk about what is. This can be done by preventing all words from your stop word list from being analysed. Stop words are commonly applied in search systems, text classification applications, topic modelling, topic extraction and others.

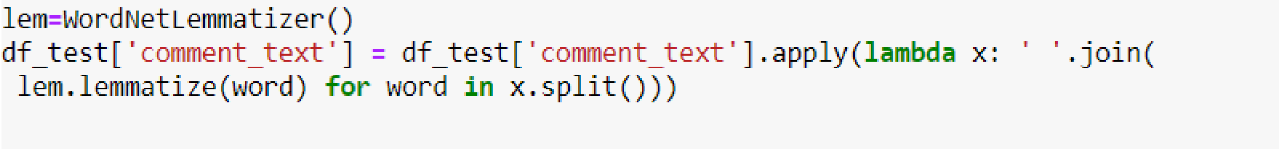


```
# remove stopwords
stop_words = set(stopwords.words('english') + ["m","ur","aww","d","dont","cant","doin","ja","u"])
df_test["comment_text"]= df_test["comment_text"].apply(lambda x: ' '.join(
    term for term in x.split() if term not in stop_words ))
```

- **Lemmatization:** Lemmatization on the surface is very similar to stemming, where the goal is to remove inflections and map a word to its root form. The only difference is that, lemmatization tries to do it the proper way. It doesn’t just chop things off, it actually transforms words to the actual root. For example, the word “better” would map too “good”. It may use a dictionary such as word net for mapping or some special rule-based approaches. Here is an example of lemmatization in action using a WordNet-based approach:

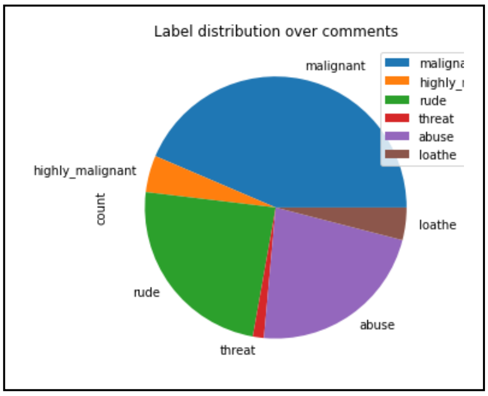


	original_word	lemmatized_word
0	trouble	trouble
1	troubling	trouble
2	troubled	trouble
3	troubles	trouble



```
lem=WordNetLemmatizer()
df_test['comment_text'] = df_test['comment_text'].apply(lambda x: ' '.join(
 lem.lemmatize(word) for word in x.split()))
```

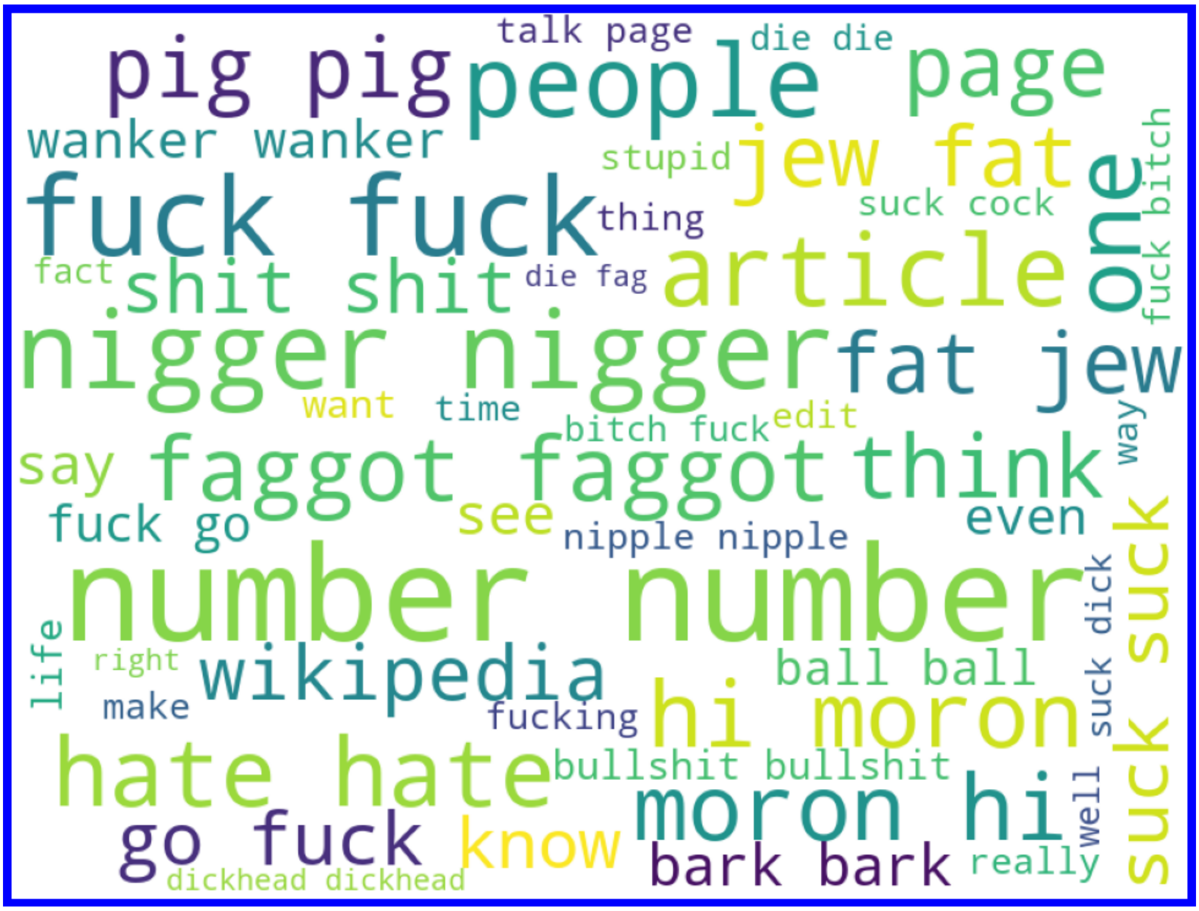
Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed.



This pie plot gives us the information about the no of comment which are classified as malignant, highly-malignant, abuse, loathe, threat. It gives us the information about the distribution of comments which are malignant into various sections.

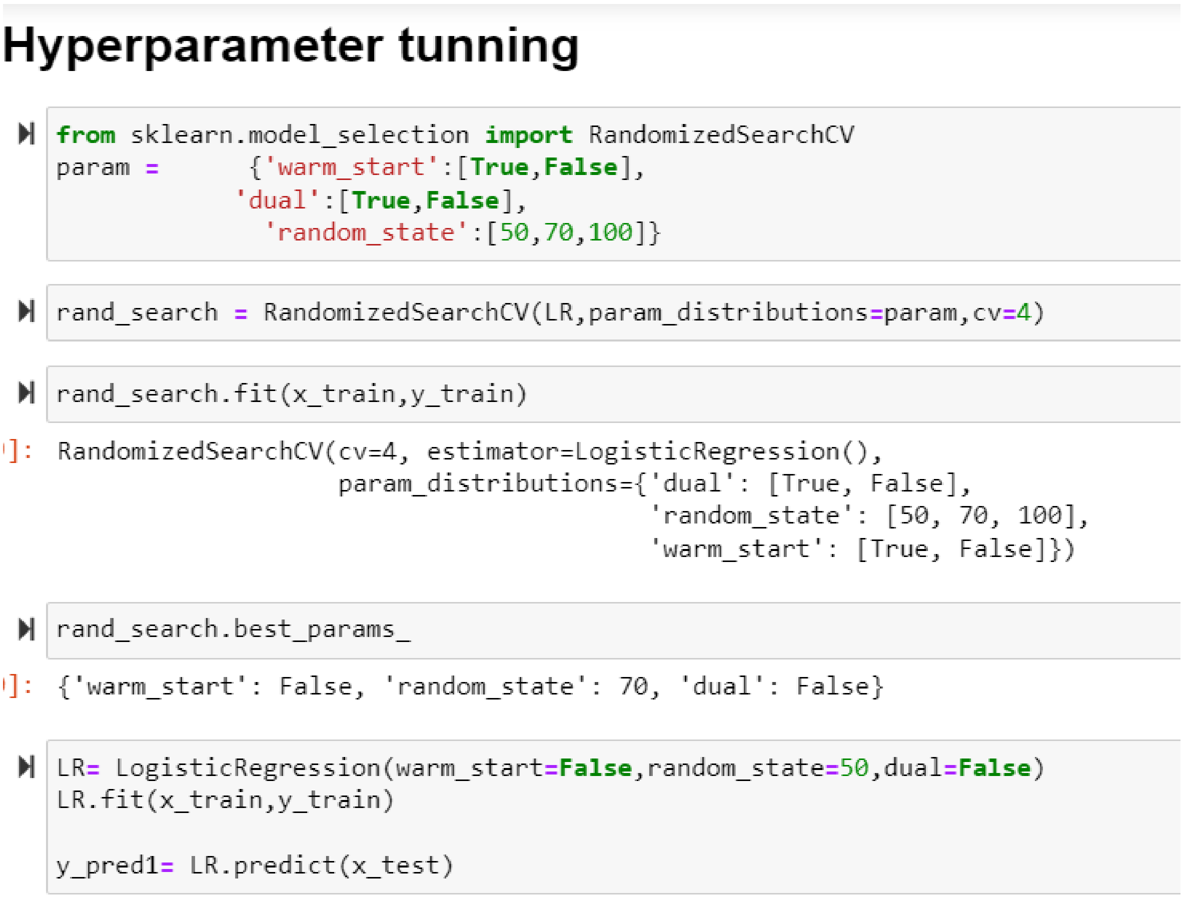
Word Cloud:

Word Cloud is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. Word clouds are widely used for analyzing data from social network websites.



- **Interpretation of the Results:**

Many machine learning algorithms are used to predict. However, the prediction accuracy of these algorithms depends heavily on the given data when training the model. If the data is in bad shape, the model will be overfitted and inefficient,which means that data pre-processing is an important part of this experiment and will affect the final results. Thus, multiple combinations of pre-processing methods need to be tested before getting the data ready to be used in training. After analyzing every model logistic regression shows good accuracy and cv with least difference and on doing hyper parameter tuning it accuracy reaches to 96%.



Hyperparameter tunning

```
from sklearn.model_selection import RandomizedSearchCV
param =      {'warm_start':[True,False],
            'dual':[True,False],
              'random_state':[50,70,100]}
```

```
rand_search = RandomizedSearchCV(LR,param_distributions=param,cv=4)
```

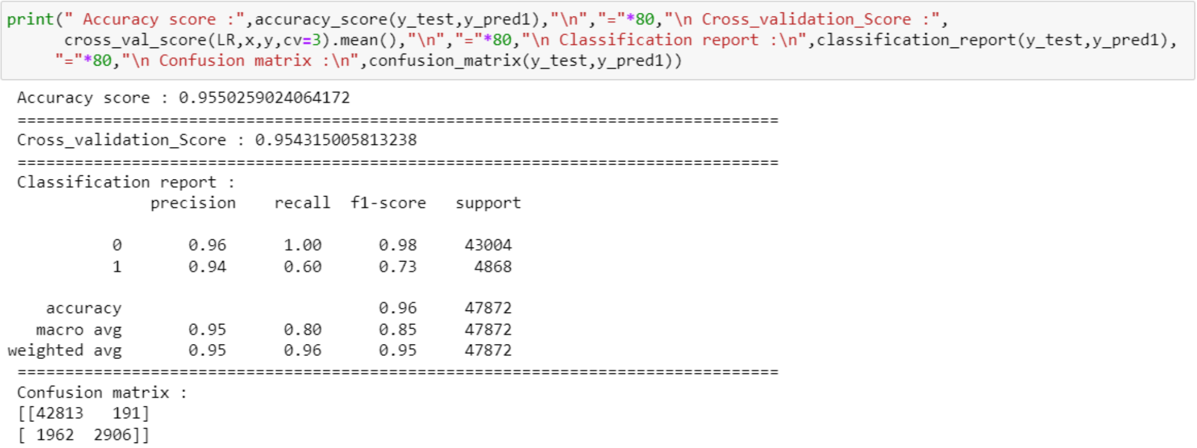
```
rand_search.fit(x_train,y_train)
```

```
]:  RandomizedSearchCV(cv=4, estimator=LogisticRegression(),
                       param_distributions={'dual': [True, False],
                                            'random_state': [50, 70, 100],
                                            'warm_start': [True, False]})
```

```
rand_search.best_params_
```

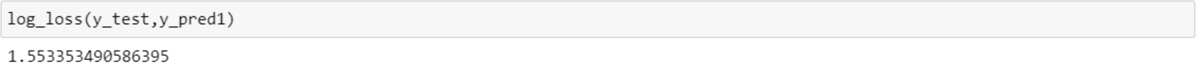
```
]:  {'warm_start': False, 'random_state': 70, 'dual': False}
```

```
LR= LogisticRegression(warm_start=False,random_state=50,dual=False)
LR.fit(x_train,y_train)

y_pred1= LR.predict(x_test)
```



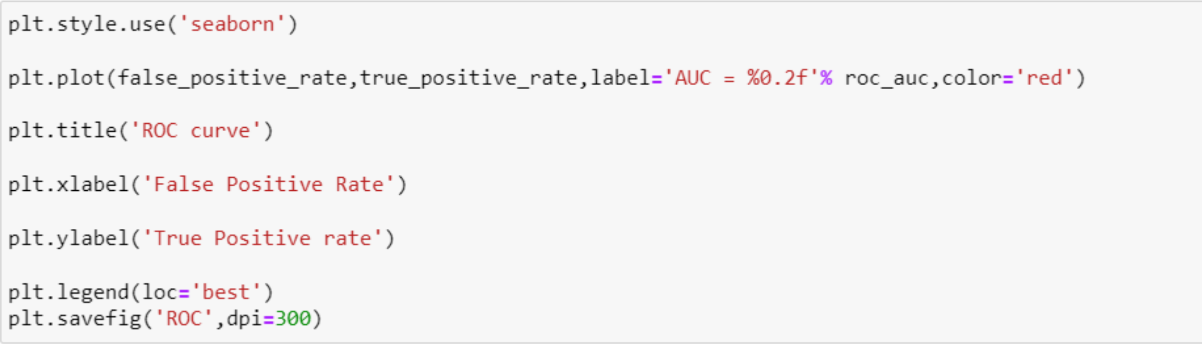
```
print(" Accuracy score :",accuracy_score(y_test,y_pred1),"\n","="*80,"\n Cross_validation_Score :",
      cross_val_score(LR,x,y,cv=3).mean(),"\n","="*80,"\n Classification report :\n",classification_report(y_test,y_pred1),
     "="*80,"\n Confusion matrix :\n",confusion_matrix(y_test,y_pred1))
```

```
 Accuracy score : 0.9550259024064172
 ================================================================================
 Cross_validation_Score : 0.954315005813238
 ================================================================================
 Classification report :
               precision    recall  f1-score   support

           0       0.96      1.00      0.98     43004
           1       0.94      0.60      0.73      4868

    accuracy                           0.96     47872
   macro avg       0.95      0.80      0.85     47872
weighted avg       0.95      0.96      0.95     47872
 ================================================================================
 Confusion matrix :
 [[42813   191]
 [ 1962  2906]]
```

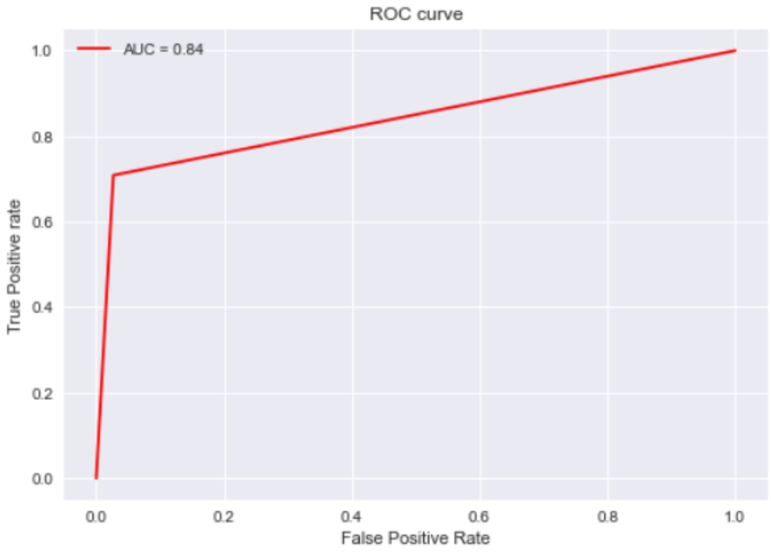


```
log_loss(y_test,y_pred1)
```

```
1.553353490586395
```



```
plt.style.use('seaborn')

plt.plot(false_positive_rate,true_positive_rate,label='AUC = %0.2f'% roc_auc,color='red')

plt.title('ROC curve')

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive rate')

plt.legend(loc='best')
plt.savefig('ROC',dpi=300)
```



CONCLUSION

- **Key Findings and Conclusions of the Study:**
  - I used various classification methods and out of all machine learning algorithm used, Logistic Regression yields the best results.
  - These malignant comments classification can be used by social media companies to filter and classify some keywords as highly malignant and set their own policy going forward for the customers and other shareholders.
  - For this project I have used wordcloud method that represents the visualization of most frequent words that are highly malignant in nature available in the dataset.

- **Learning Outcomes of the Study in respect of Data Science**:

  As per as learning outcomes is concerned, I have learnt the following things in this project:

  - Algorithm need to be used by understanding the dataset for the classification model.
  - From describe method we can get some knowledge related to outliers present in the particular columns (large difference between 75th percentile and maximum percentile)
  - I also understand the visualization of related features and importance related to dataset.
  - I have also used NLTK library to clean the text/comments and find out the actual length of the comments that can be used for further evaluation.

- **Challenges**:

➢ It was difficult to load the dataset in notebook as it took some time.

➢ Running each line code was a bit slow in notebook, possibly due to low CPU configuration.

- **Limitations of this work and Scope for Future Work:**
  - ➢ Since I have only used a sample dataset, hence sometimes it is difficult to understand the overall impact of this project while filter out the toxic comments in public forum.

Thank you