FLIP ROBO

MALIGNANT COMMENTS CLASSIFICATION PROJECT.

Submitted by:

DISHANT DOSHI

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. 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 carefully.

- Conceptual Background of the Domain Problem

Online platforms and social media become the place where people share the thoughts freely without any partiality and overcoming all the race people share their thoughts and ideas among the crowd. Social media is a computer-based technology that facilitates the sharing of ideas, thoughts, and information through the building of virtual networks and communities. By design, social media is Internetbased and gives users quick electronic communication of content. Content includes personal information, documents, videos, and photos. Users engage with social media via a computer, tablet, or smartphone via web-based software or applications. While social media is ubiquitous in America and Europe, Asian countries like India lead the list of social media usage. More than 3.8 billion people use social media. In this huge online platform or an online community there are some people or some motivated mob wilfully bully others to make them not to share their thought in rightful way. They bully others in a foul language which among the civilized society is seen as ignominy. And when innocent individuals are being bullied by these mob these individuals are going silent without speaking anything. So, ideally the motive of this disgraceful mob is achieved. To solve this problem, we are now building a model that identifies all the foul language and foul words, using which the online platforms like social media principally stops these mob using the foul language in an online community or even block them or block them fom using this foul language.

- Review of Literature
  1. Identify the foul words or foul statements that are being used.
  2. 2. Stop the people from using these foul languages in online public forum.

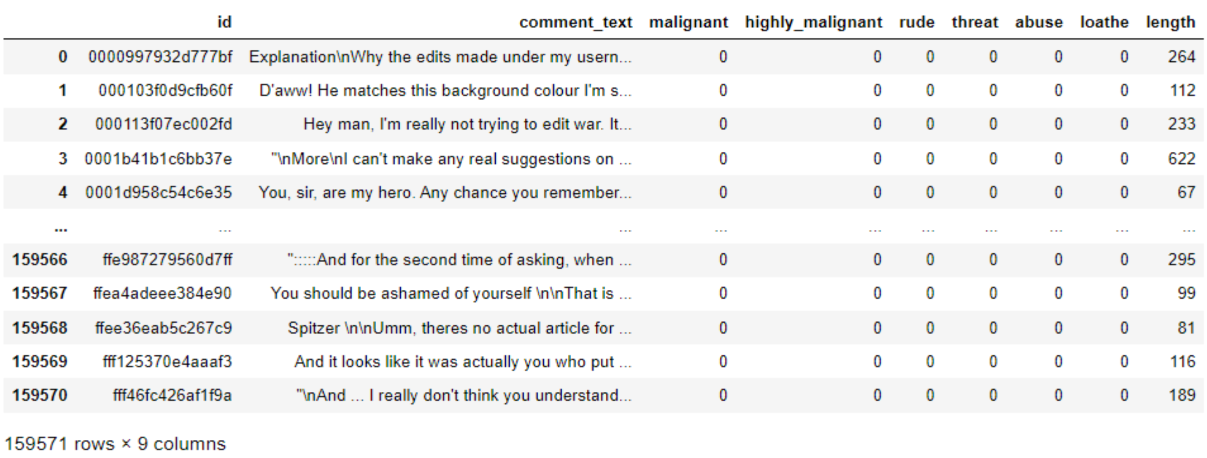
To solve this problem, we are now building a model using our machine language technique that identifies all the foul language and foul words, using which the online platforms like social media principally stops these mob using the foul language in an online community or even block them or block them from using this foul language. I have used 5 different Classification algorithms and shortlisted the best on basis on the metrics of performance and I have chosen one algorithm and build a model in that algorithm.

Motivation for the Problem Undertaken

One of the first lessons we learn as children is that the louder you scream and the bigger of a tantrum you throw, you more you get your way. Part of growing up and maturing into an adult and functioning member of society is learning how to use language and reasoning skills to communicate our beliefs and respectfully disagree with others, using evidence and persuasiveness to try and bring them over to our way of thinking. Social media is reverting us back to those animalistic tantrums, schoolyard taunts and unfettered bullying that define youth, creating a dystopia where even renowned academics and dispassionate journalists transform from Dr. Jekyll into raving Mr. Hyde's, raising the critical question of whether social media should simply enact a blanket ban on profanity and name calling? Actually, ban should be implemented on these profanities and taking that as a motivation I have started this project to identify the malignant comments in social media or in online public forms.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem



	id	comment_text	malignant	highly_malignant	rude	threat	abuse	loathe	length
0	0000997932d777bf	Explanation\nWhy the edits made under my usern...	0	0	0	0	0	0	264
1	000103f0d9cfb60f	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0	112
2	000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0	233
3	0001b41b1c6bb37e	"\nMore\nI can't make any real suggestions on ...	0	0	0	0	0	0	622
4	0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0	67
...	...	...	...	...	...	...	...	...	...
159566	ffe987279560d7ff	":::::And for the second time of asking, when ...	0	0	0	0	0	0	295
159567	ffea4adeee384e90	You should be ashamed of yourself \n\nThat is ...	0	0	0	0	0	0	99
159568	ffee36eab5c267c9	Spitzer \n\nUmm, theres no actual article for ...	0	0	0	0	0	0	81
159569	fff125370e4aaaf3	And it looks like it was actually you who put ...	0	0	0	0	0	0	116
159570	fff46fc426af1f9a	"\nAnd ... I really don't think you understand...	0	0	0	0	0	0	189

159571 rows × 9 columns

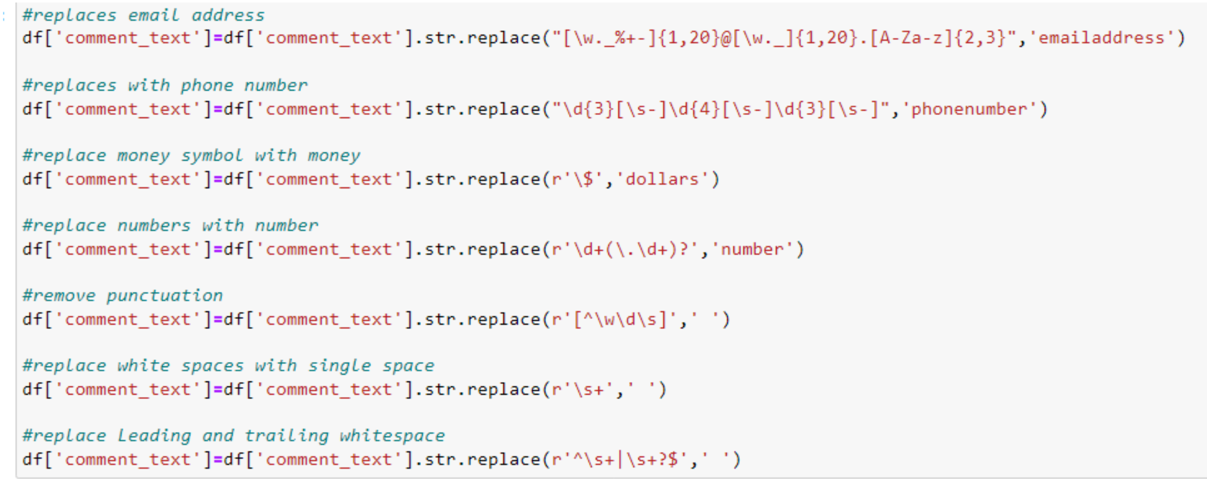
From this we can see that comment text contains too much un necessary words.so we will reduce this unnecessary words.so that model will learn good thing.

We will use lemmatization and TF-IDF for feature extraction

Data Sources and their formats

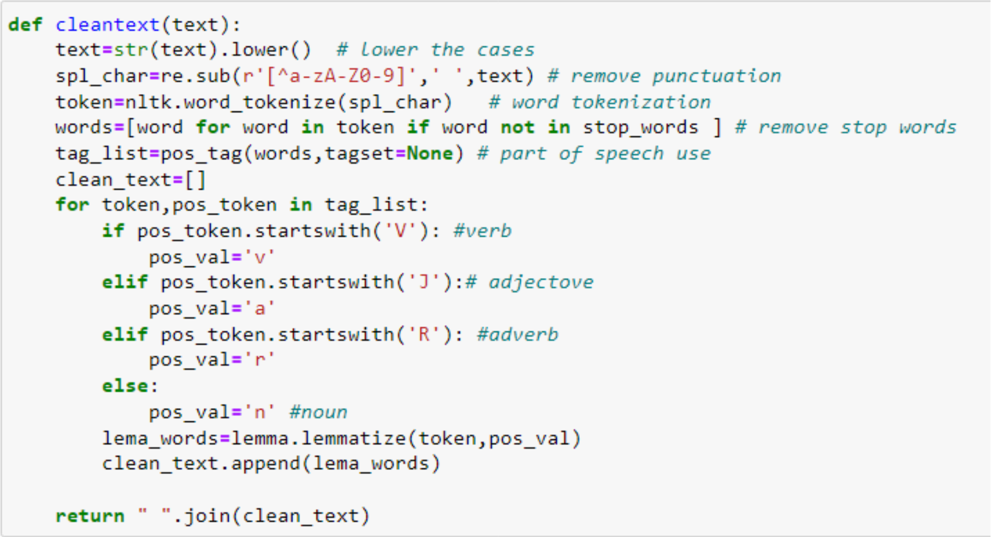
We have 159571 rows as text in comment text and its divide into 6 different categories.

Data Preprocessing Done



```
#replaces email address
df['comment_text']=df['comment_text'].str.replace("[\w._%+-]{1,20}@[\w._]{1,20}.[A-Za-z]{2,3}",'emailaddress')

#replaces with phone number
df['comment_text']=df['comment_text'].str.replace("\d{3}[\s-]\d{4}[\s-]\d{3}[\s-]",'phonenumber')

#replace money symbol with money
df['comment_text']=df['comment_text'].str.replace(r'\$','dollars')

#replace numbers with number
df['comment_text']=df['comment_text'].str.replace(r'\d+(\.\d+)?','number')

#remove punctuation
df['comment_text']=df['comment_text'].str.replace(r'[^\w\d\s]',' ')

#replace white spaces with single space
df['comment_text']=df['comment_text'].str.replace(r'\s+',' ')

#replace Leading and trailing whitespace
df['comment_text']=df['comment_text'].str.replace(r'^\s+|\s+?$',' ')
```

As per above Image we use regurization for replace words



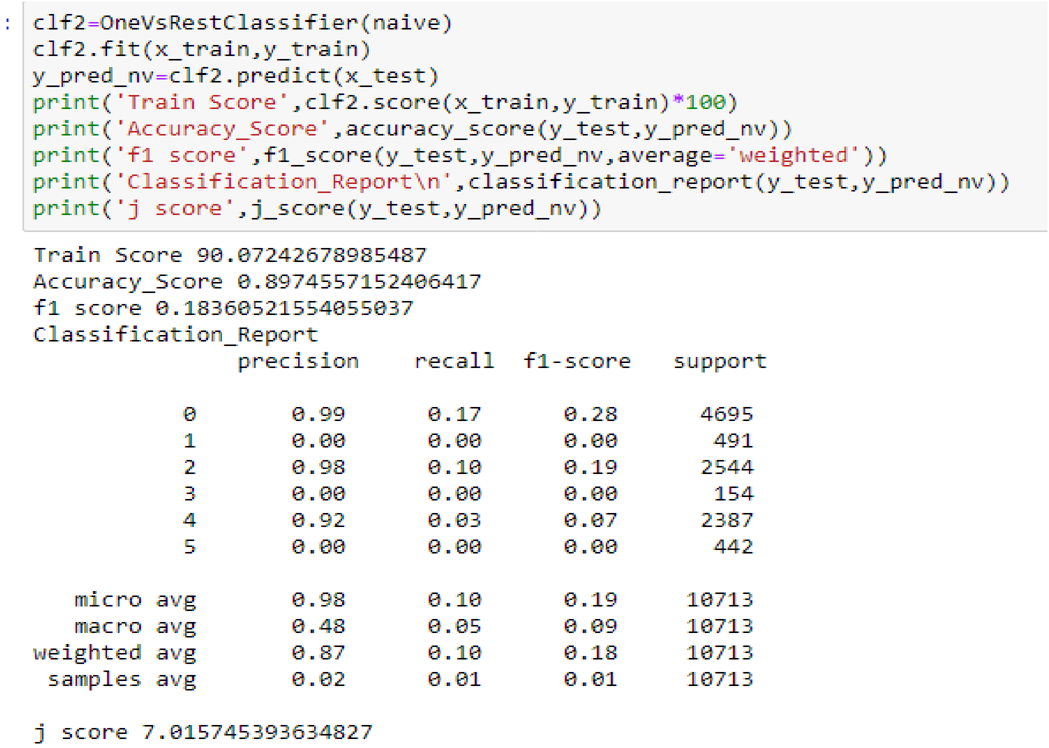
```
def cleantext(text):
    text=str(text).lower()  # lower the cases
    spl_char=re.sub(r'[^a-zA-Z0-9]',' ',text) # remove punctuation
    token=nltk.word_tokenize(spl_char)   # word tokenization
    words=[word for word in token if word not in stop_words ] # remove stop words
    tag_list=pos_tag(words,tagset=None) # part of speech use
    clean_text=[]
    for token,pos_token in tag_list:
        if pos_token.startswith('V'): #verb
            pos_val='v'
        elif pos_token.startswith('J'):# adjectove
            pos_val='a'
        elif pos_token.startswith('R'): #adverb
            pos_val='r'
        else:
            pos_val='n' #noun
        lema_words=lemma.lemmatize(token,pos_val)
        clean_text.append(lema_words)

    return " ".join(clean_text)
```

We use above function for Data Cleaning Purpose.

- Hardware and Software Requirements and Tools Used
- Pandas
- NumPy
- NLTK
- RE
- Matplotlib
- Seaborn

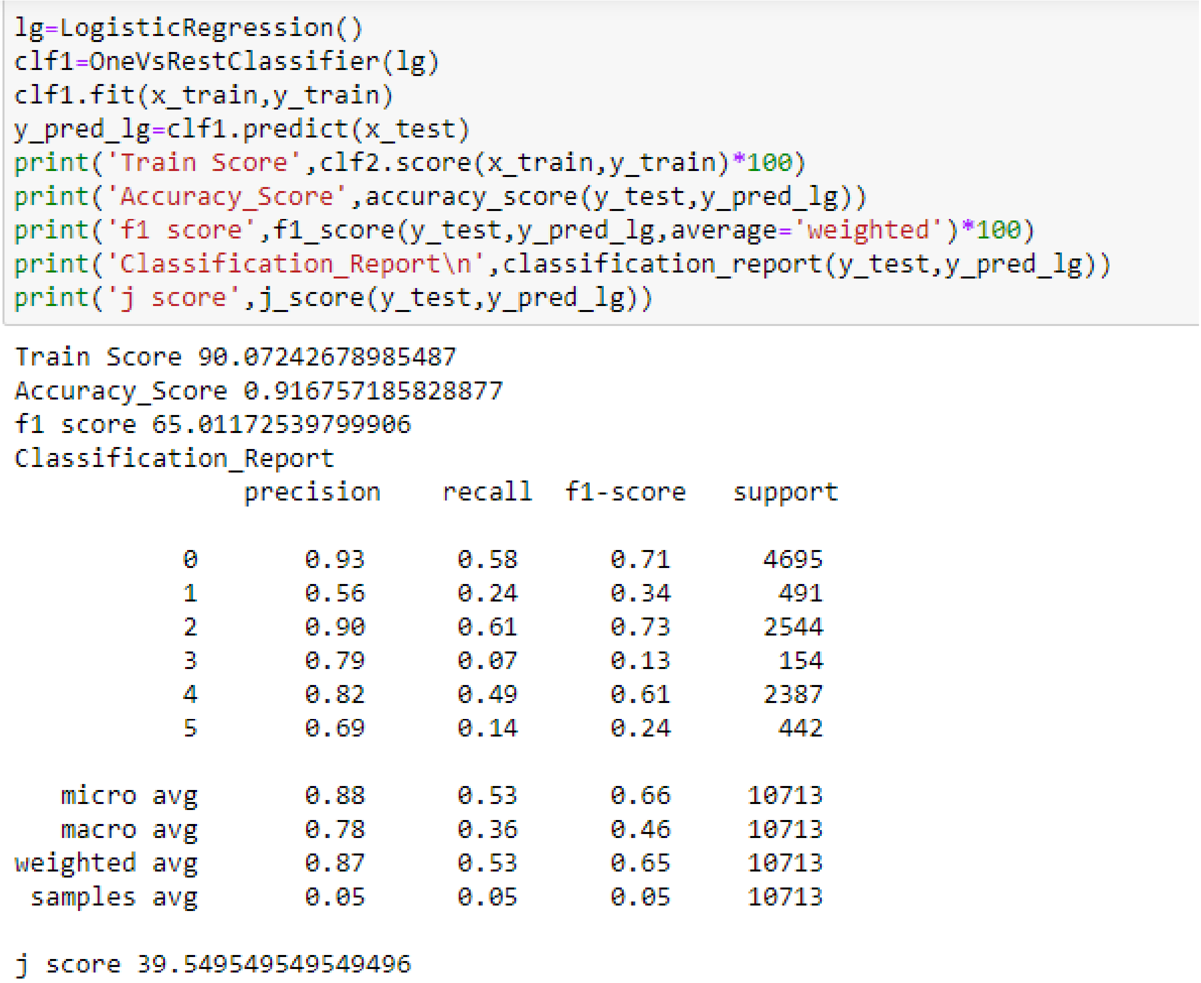
Model/s Development and Evaluation

- Testing of Identified Approaches (Algorithms)



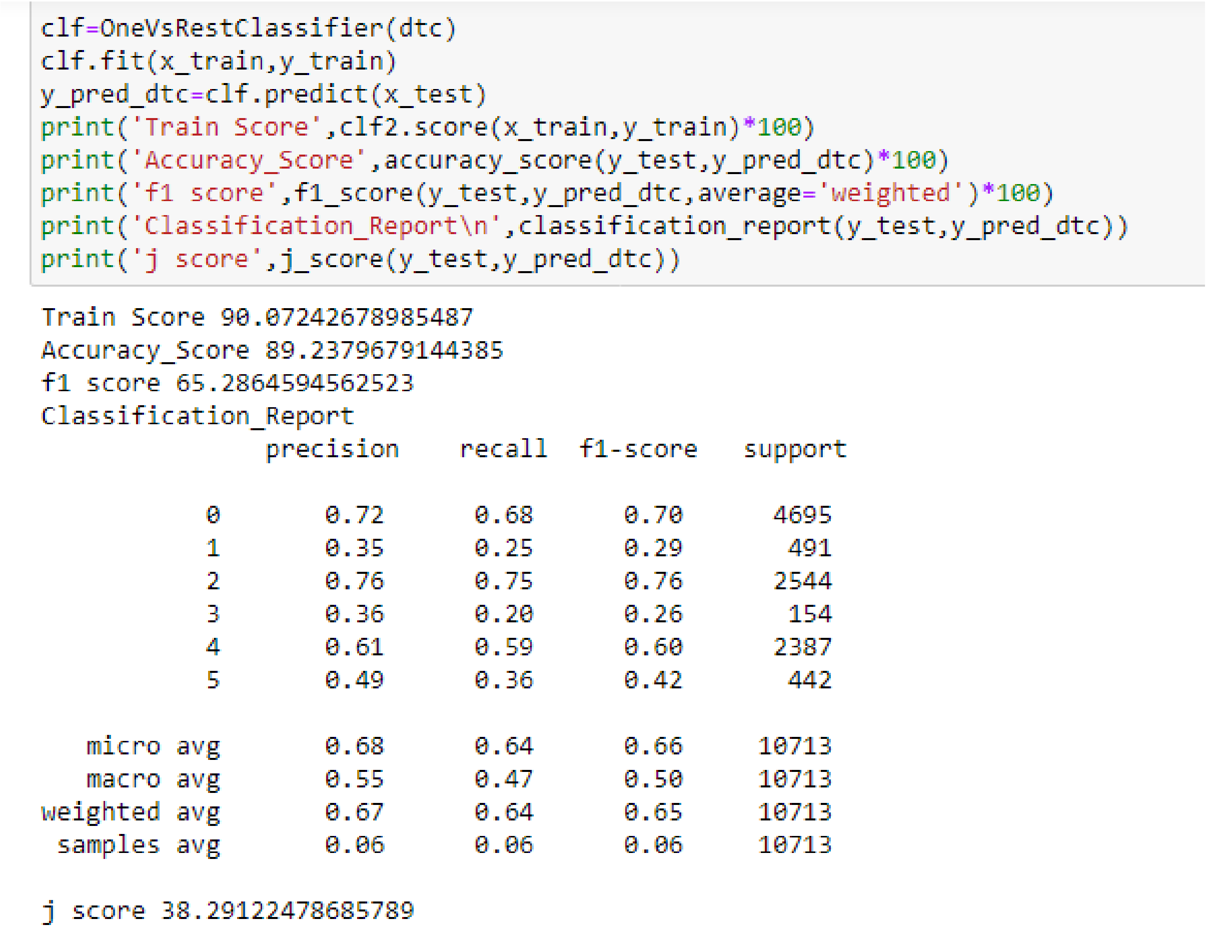
```
clf2=OneVsRestClassifier(naive)
clf2.fit(x_train,y_train)
y_pred_nv=clf2.predict(x_test)
print('Train Score',clf2.score(x_train,y_train)*100)
print('Accuracy_Score',accuracy_score(y_test,y_pred_nv))
print('f1 score',f1_score(y_test,y_pred_nv,average='weighted'))
print('Classification_Report\n',classification_report(y_test,y_pred_nv))
print('j score',j_score(y_test,y_pred_nv))
```

```
Train Score 90.07242678985487
Accuracy_Score 0.8974557152406417
f1 score 0.18360521554055037
Classification_Report
               precision    recall  f1-score   support

           0       0.99      0.17      0.28      4695
           1       0.00      0.00      0.00       491
           2       0.98      0.10      0.19      2544
           3       0.00      0.00      0.00       154
           4       0.92      0.03      0.07      2387
           5       0.00      0.00      0.00       442

   micro avg       0.98      0.10      0.19     10713
   macro avg       0.48      0.05      0.09     10713
weighted avg       0.87      0.10      0.18     10713
 samples avg       0.02      0.01      0.01     10713

j score 7.015745393634827
```



```
lg=LogisticRegression()
clf1=OneVsRestClassifier(lg)
clf1.fit(x_train,y_train)
y_pred_lg=clf1.predict(x_test)
print('Train Score',clf2.score(x_train,y_train)*100)
print('Accuracy_Score',accuracy_score(y_test,y_pred_lg))
print('f1 score',f1_score(y_test,y_pred_lg,average='weighted')*100)
print('Classification_Report\n',classification_report(y_test,y_pred_lg))
print('j score',j_score(y_test,y_pred_lg))
```

```
Train Score 90.07242678985487
Accuracy_Score 0.916757185828877
f1 score 65.01172539799906
Classification_Report
               precision    recall  f1-score   support

           0       0.93      0.58      0.71      4695
           1       0.56      0.24      0.34       491
           2       0.90      0.61      0.73      2544
           3       0.79      0.07      0.13       154
           4       0.82      0.49      0.61      2387
           5       0.69      0.14      0.24       442

   micro avg       0.88      0.53      0.66     10713
   macro avg       0.78      0.36      0.46     10713
weighted avg       0.87      0.53      0.65     10713
 samples avg       0.05      0.05      0.05     10713

j score 39.549549549549496
```



```
clf=OneVsRestClassifier(dtc)
clf.fit(x_train,y_train)
y_pred_dtc=clf.predict(x_test)
print('Train Score',clf2.score(x_train,y_train)*100)
print('Accuracy_Score',accuracy_score(y_test,y_pred_dtc)*100)
print('f1 score',f1_score(y_test,y_pred_dtc,average='weighted')*100)
print('Classification_Report\n',classification_report(y_test,y_pred_dtc))
print('j score',j_score(y_test,y_pred_dtc))
```

```
Train Score 90.07242678985487
Accuracy_Score 89.2379679144385
f1 score 65.2864594562523
Classification_Report
               precision    recall  f1-score   support

           0       0.72      0.68      0.70      4695
           1       0.35      0.25      0.29       491
           2       0.76      0.75      0.76      2544
           3       0.36      0.20      0.26       154
           4       0.61      0.59      0.60      2387
           5       0.49      0.36      0.42       442

   micro avg       0.68      0.64      0.66     10713
   macro avg       0.55      0.47      0.50     10713
weighted avg       0.67      0.64      0.65     10713
 samples avg       0.06      0.06      0.06     10713

j score 38.29122478685789
```

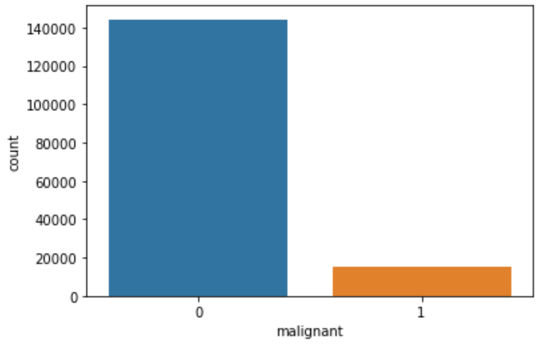
- Key Metrics for success in solving problem under consideration

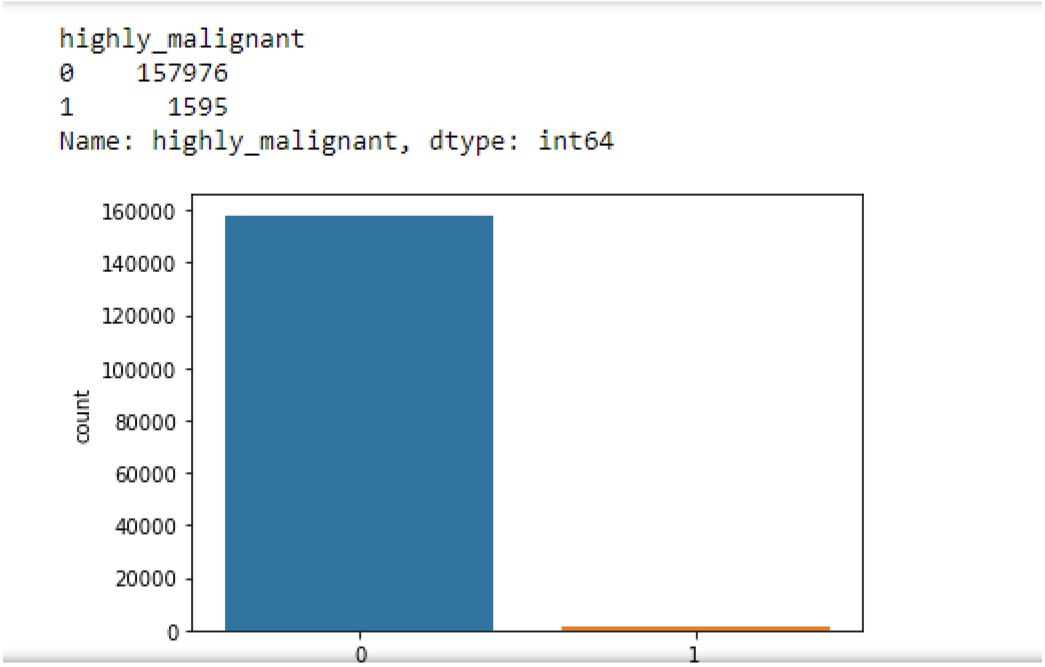
  F1-score

  j-score

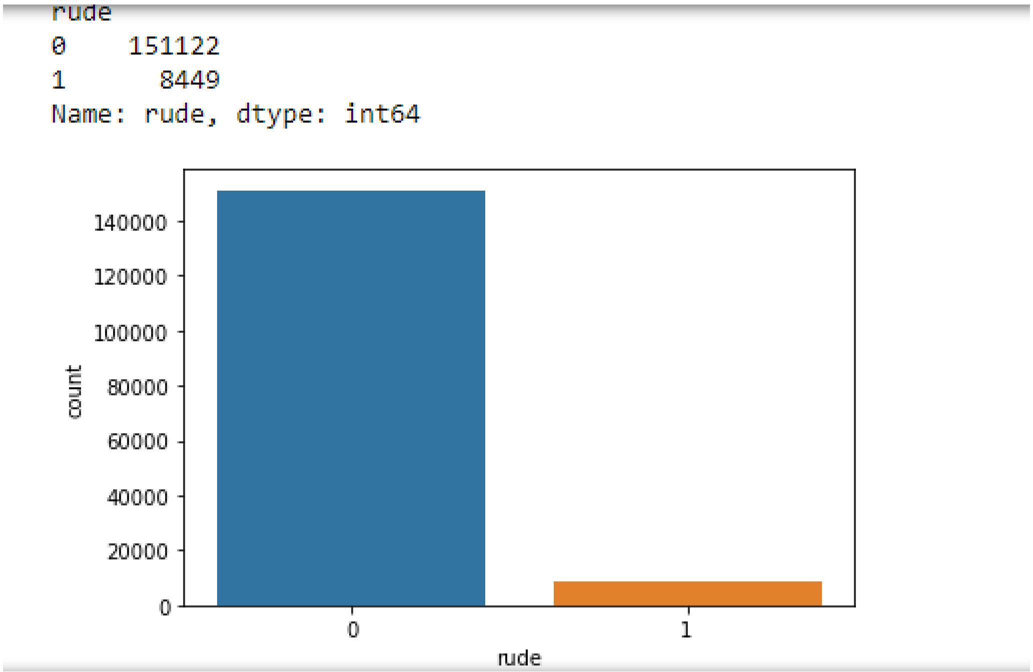
  Accuracy Score

- Visualizations

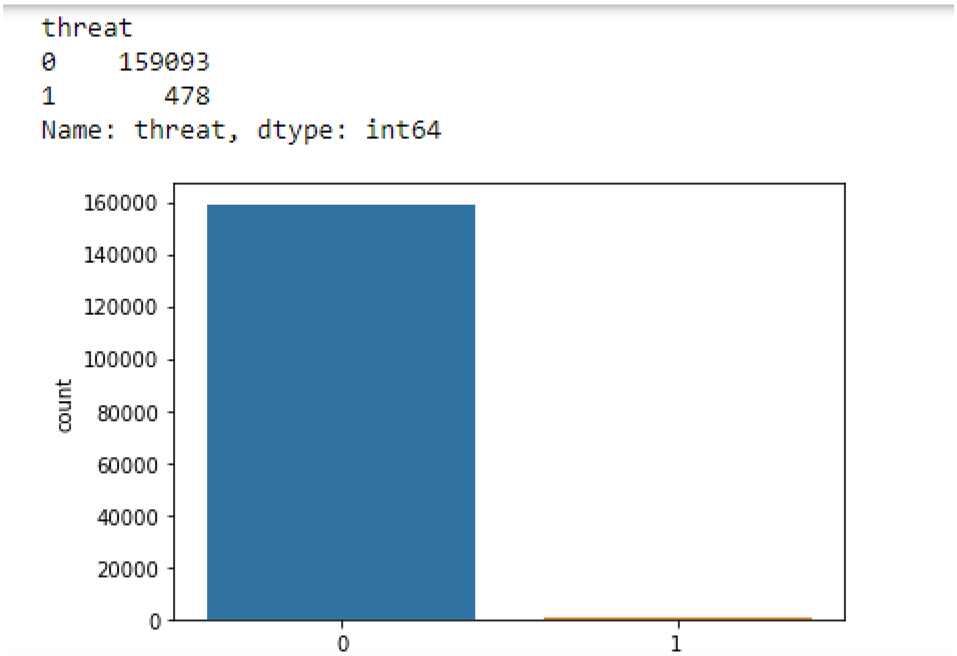
- 



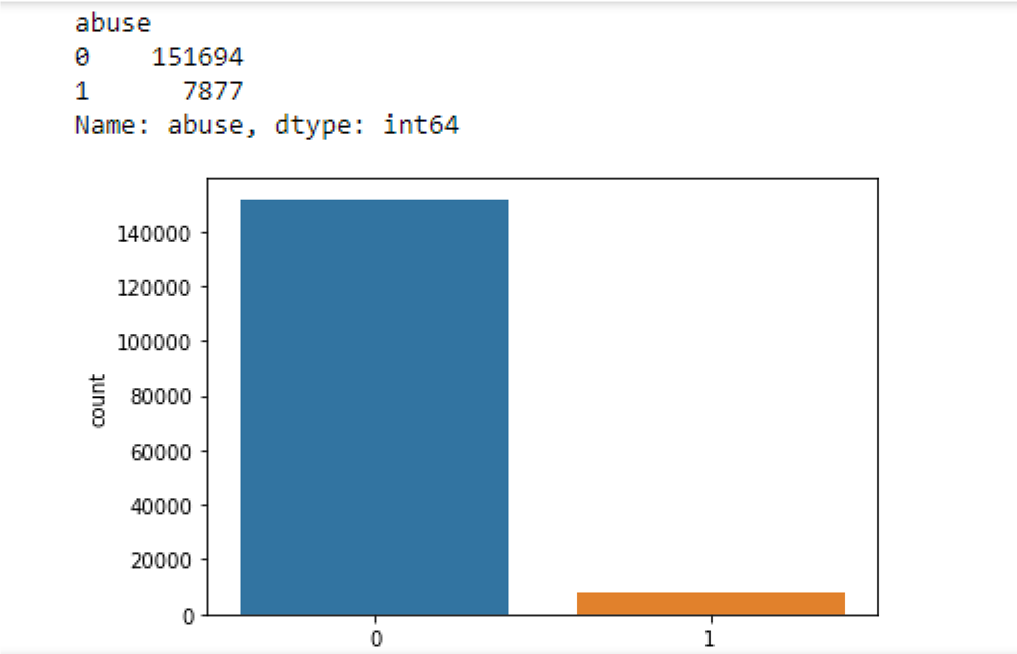
```
highly_malignant
0    157976
1      1595
Name: highly_malignant, dtype: int64
```

- 

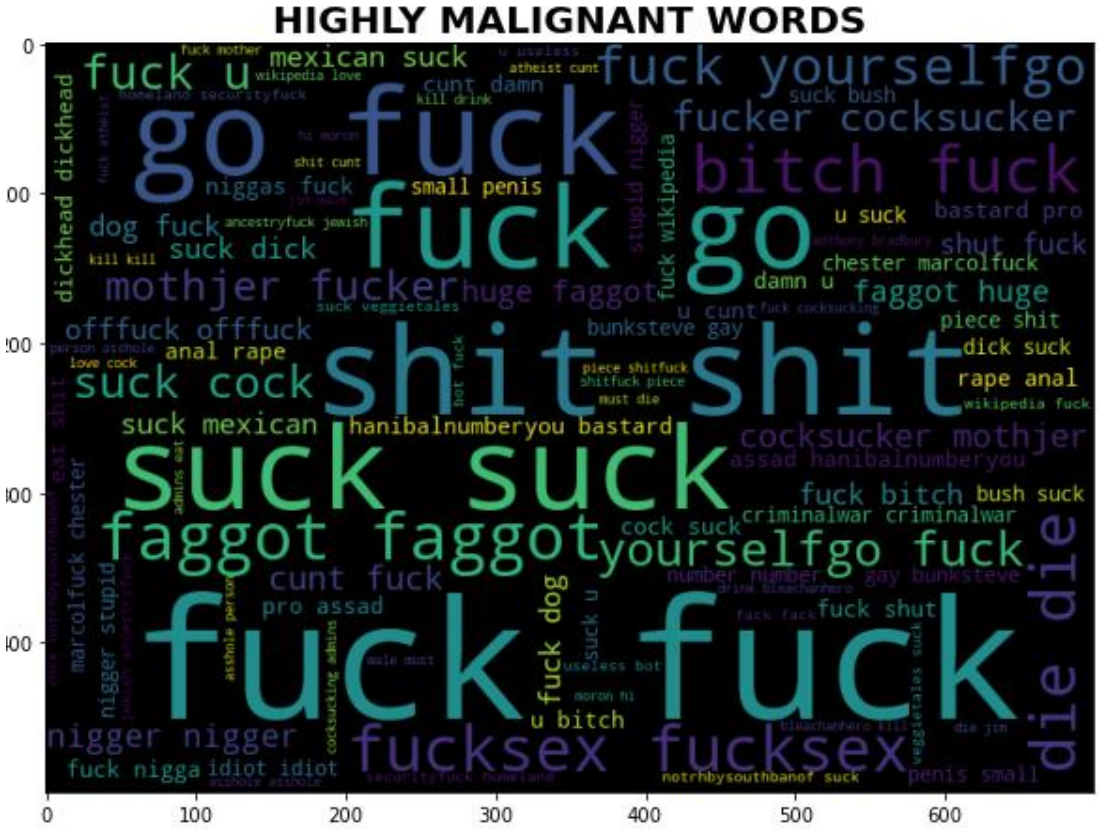
```
rude
0    151122
1      8449
Name: rude, dtype: int64
```



```
threat
0    159093
1       478
Name: threat, dtype: int64
```



```
abuse
0    151694
1      7877
Name: abuse, dtype: int64
```



CONCLUSION

- Key Findings and Conclusions of the Study

The finding of the study is that only few users over online use unparliamentary language. And most of these sentences have more stop words, and are being long. As discussed before few motivated disrespectful crowds uses these foul languages in the online forum to bully the people around and to stop them from doing the things that they are supposed to do. Our Study helps the online forms and social media to induce a ban to profanity or usage of profanity over these forms.

- Learning Outcomes of the Study in respect of Data Science

The use of social media is the most common trend among the activities of today’s people. Social networking sites offer today’s teenagers a platform for communication and entertainment. They use social media to collect more information from their friends and followers. The vastness of social media sites ensures that not all of them provide a decent environment for children. In such cases, the impact of the negative influences of social media on teenage users increases with an increase in the use of offensive language in social conversations. This increase could lead to frustration, depression and a large change in their behaviour. Hence, I propose a novel approach to classify bad language usage in text conversations. I have considered the English medium for textual conversation. I have developed our system based on a foul language classification approach; it is based on an improved version of a Random Forest Classification Algorithm that detects offensive language usage in a conversation. As per our evaluation, we found that lesser number of users conversation is not decent all the time. We trained 190000 observations for eight context categories using a Random Forest algorithm for context detection. Then, the system classifies the use of foul language in one of the trained contexts in the text conversation. In our testbed, we observed 10% of participants used foul language during their text conversation. Hence, our proposed approach can identify the impact of foul language in text conversations using a classification technique and emotion detection to identify the foul language usage.

Limitations of this work and Scope for Future Work

The limitation of the study is that we have a imbalanced data so our model learnt more about the non-abusive sentence more than the abusive sentence. Which makes our model act like a overfit model when tested with live data. And also, model tend to not identify a foul or a sarcastically foul language.