FlipRobo Technologies

Project Name:
Malignant Comments Classifier Report

- Submitted by Smriti Mathur

Acknowledgement

I would like to express my deep sense of gratitude to my SME (Subject Matter Expert) Khushboo Garg as well as Flip Robo Technologies who gave me the golden opportunity to do this data analysis project on Malignant Comments Classifier, which also helped me in doing lots of research and I came to know about so many new things.

I have put in my all efforts while doing this project. A huge thanks to my academic team “DataTrained” who is the reason behind what I am today. Last but not least, my parents have been my backbone in every step of my life. And also thank you to many other persons who have helped me directly or indirectly to complete the project.

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 un-offensive, but “u are an idiot” is offensive.

Our goal is to build a prototype of online hate and abuse comment classifier which can be used to classify hate and offensive comments so that they can be controlled and restricted from spreading hatred and cyberbullying.

Conceptual Background of the Domain Problem:
Online platforms and social media become the place where people share their thoughts freely without any partiality and overcome all the races 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 Internet-based 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 share their thought rightfully. They

bully others with foul language which among the civilised society is seen as ignominy. And when innocent individuals are being bullied by this 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 stop these mobs from using the foul language in an online community or even block them or block them from using this foul language.

Literature Review:

The purpose of the literature review is to:
#1 Identify the foul words or foul statements that are being used.
#2 Stop the people from using these foul languages in an 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 9 different Classification algorithms and shortlisted the best on basis of the metrics of performance and I have chosen one algorithm and built a model in that algorithm.

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. Our goal is to build a prototype of online hate and abuse comment classifier which can be used to classify hate and offensive comments so that they can be controlled and restricted from spreading hatred and cyberbullying.

Motivation for the Problem Undertaken:

Social media is reverting us to those animalistic tantrums, schoolyard taunts, and unfettered bullying that define youth, creating a dystopia. With the widespread usage of online social networks and their popularity, social networking platforms have given us incalculable opportunities than ever before, and

their benefits are undeniable. Despite benefits, people may be humiliated, insulted, bullied, and harassed by anonymous users, strangers, or peers. In this study, we have proposed a cyberbullying detection framework to generate features from online content by leveraging a pointwise mutual information technique.

Based on these features, we developed a supervised machine learning solution for cyberbullying detection and multi-class categorization of its severity. Results from experiments with our proposed framework in a multi-class setting are promising both concerning classifier accuracy and f-measure metrics. These results indicate that our

proposed framework provides a feasible solution to detect cyberbullying behaviour and its severity in online social networks.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem:
In this project, we have been provided with two datasets namely train and test CSV files. We will build a machine learning model by using NLP using a training dataset. And using this model we will make predictions for our test dataset.

We will need to build multiple classification machines and learning models. Before model building, we will need to perform all data pre-processing steps involving NLP. After trying different classification models with different hyperparameters than will select the best model out of them. We will need to follow the complete life cycle of data science that includes steps like -

1. Data Cleaning
2. Exploratory Data Analysis
3. Data Pre-processing
4. Model Building
5. Model Evaluation
6. Selecting the best model

Finally, we compared the results of proposed and baseline features with other machine learning algorithms. The findings of the comparison indicate the significance of the proposed features in cyberbullying detection.

Data Source/ Problems:
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 include ‘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 an indication of the comments that are giving any threat to someone.
- Abuse: It is for abusive comments.

- Loathe: It describes the hateful comments 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.

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 a good amount of data exploration and derive some interesting features using the comments text column available. We need to build a model that can differentiate between comments and their categories.

Importing Libraries:

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import re
import string
import nltk
from nltk.corpus import  stopwords
import sklearn
from sklearn import preprocessing
from sklearn.feature_selection import SelectFromModel
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, f1_score, precision_score, recall_score,
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import MultinomialNB, BernoulliNB
from sklearn.svm import LinearSVC
from sklearn.ensemble import RandomForestClassifier

import warnings
warnings.filterwarnings('ignore')
```

Data Pre-Processing:

#1 Loading dataset.
#2 Checking and removing null values.
#3 Drop column ‘ID’.
#4 Convert comment text to lower case and replace '\n' with a single space.
#5 Keep only text data ie. ‘a-z' and remove other data from comment text.
#6 Remove stop words and punctuations.
#7 Apply Stemming using SnowballStemmer.
#8 Convert text to vectors using TfidfVectorizer
#9 Load saved or serialised model.
#10 Predict values for multi-class labels.

Data Inputs- Logic- Output Relationships

We have analysed the input-output logic with a word cloud and have word clouded the sentences that are classified as foul language in every category. A tag/word cloud is a novelty visual representation of text data, typically used to depict keyword metadata on websites or to visualise free-form text. It’s an image composed of words used in a particular text or subject, in which the size of each word indicates its frequency or importance.

```
In [25]:
#importing required libraries
from wordcloud import WordCloud
```

```
In [26]:
cols = 3
rows = len(output_labels)//cols
if len(output_labels) % cols != 0:
    rows += 1

fig = plt.figure(figsize=(16,rows*cols*1.8))
fig.subplots_adjust(top=0.8, hspace=0.3)

p=1
for i in output_labels:
    word_cloud = WordCloud(height=650,width=800,background_color="white",max_words=80).generate(' '.join(df.comment_text[df[i
    ax = fig.add_subplot(rows,cols,p)
    ax.imshow(word_cloud)
    ax.set_title(f"WordCloud for: [{i}]",fontsize=14)
    for spine in ax.spines.values():
        spine.set_edgecolor('r')

    ax.set_xticks([])
    ax.set_yticks([])
    p += 1

fig.suptitle("WordCloud: Representation of Loud words in BAD COMMENTS",fontsize=16)
fig.tight_layout(pad=2)
plt.show()
```

Output is as follows:

These are the comments that belong to different types so which the help of word cloud we can see if there is an abuse comment and which type of words it contains and similar to other comments as well.

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

Cyberbullying has become a growing problem in countries around the world. Essentially, cyberbullying doesn't differ much from the type of bullying that many children have unfortunately grown accustomed to in school. The only difference is that it takes place online.

Cyberbullying is a very serious issue affecting not just the young victims, but also the victims' families, the bully, and those who witness instances of cyberbullying. However, the effect of cyberbullying can be most detrimental to the victim, of course, as they may

experience several emotional issues that affect their social and academic performance as well as their overall mental health.

Model/s Development and Evaluation:

Identification of possible problem-solving approaches (methods) - We checked through the entire training dataset for any kind of missing values information and all these pre-processing steps were repeated on the testing dataset as well. Then we went ahead and took a look at the dataset information. Using the info method, we can confirm the non-null count details as well as the datatype information. We have a total of 8 columns out of which 2 columns have object datatype while the remaining 6 columns are of integer datatype.

Then we went ahead and performed multiple data cleaning and data transformation steps. We have added a column to store the original length of our comment_text column. Since there was no use for the "id" column, we dropped it and converted all the text data in our comment text column into a lowercase format for easier interpretation.

Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or the roots of words known as a lemma. Stemming is important in natural language understanding (NLU) and natural language processing (NLP).

Testing of Identified Approaches (Algorithms):

The list of all the algorithms used for the training and testing classification model are listed below:
1) Gaussian Naïve Bayes
2) Multinomial Naïve Bayes

Run and Evaluate Selected Models:

We created a classification function that included the evaluation metrics details for the generation of our Classification Machine Learning models.

In [34]:

```
#Function to train and test model
def build_models(models,x,y,test_size=0.33,random_state=42):
    #splitting train test data using train_test_split
    x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=test_size,random_state=random_state)

    #training models using BinaryRelevance of problem transform
    for i in tqdm.tqdm(models,desc="Building Models"):
        start_time = timeit.default_timer()

        sys.stdout.write("\n=================================================================================================\n")
        sys.stdout.write(f"Current Model in Progress: {i} ")
        sys.stdout.write("\n=================================================================================================\n")

        br_clf = BinaryRelevance(classifier=models[i]["name"],require_dense=[True,True])
        print("Training: ",br_clf)
        br_clf.fit(x_train,y_train)

        print("Testing: ")
        predict_y = br_clf.predict(x_test)

        ham_loss = hamming_loss(y_test,predict_y)
        sys.stdout.write(f"\n\tHamming Loss  : {ham_loss}")

        ac_score = accuracy_score(y_test,predict_y)
        sys.stdout.write(f"\n\tAccuracy Score: {ac_score}")

        cl_report = classification_report(y_test,predict_y)
        sys.stdout.write(f"\n{cl_report}")

        end_time = timeit.default_timer()
        sys.stdout.write(f"Completed in [{end_time-start_time} sec.]")

        models[i]["trained"] = br_clf
        models[i]["hamming_loss"] = ham_loss
        models[i]["accuracy_score"] = ac_score
        models[i]["classification_report"] = cl_report
        models[i]["predict_y"] = predict_y
        models[i]["time_taken"] = end_time - start_time

        sys.stdout.write("\n=================================================================================================\n\n\n")

    models["x_train"] = x_train
    models["y_train"] = y_train
    models["x_test"] = x_test
    models["y_test"] = y_test

    return models
```

Code:

In [35]:

```
#### preparing list of models
models = {
    "GaussianNB": {
        "name":GaussianNB(),
    },
    "MultinomialNB":{
        "name":MultinomialNB(),
    },
}


#taking the one forth of the data for training and testig
half = len(df)//4
trained_models = build_models(models,X[:half,:],Y[:half,:])
```

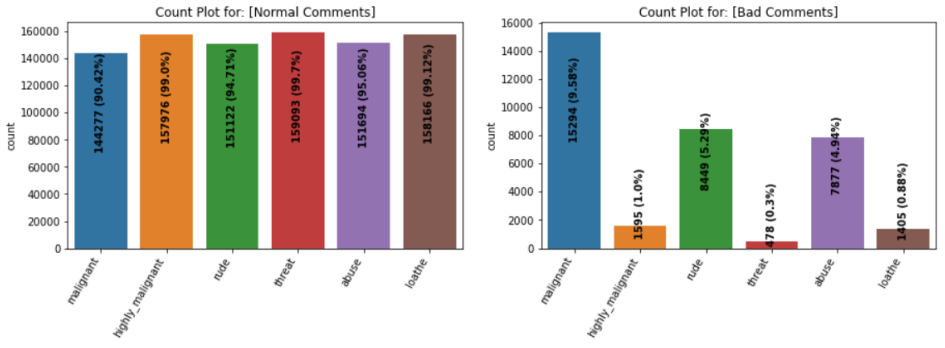
Output:

```
Building Models: 100% 2/2 [01:45<00:00, 46.86s/it]

================================================================================
Current Model in Progress: GaussianNB
================================================================================
Training:  BinaryRelevance(classifier=GaussianNB(), require_dense=[True, True])
Testing:

        Hamming Loss  : 0.21560957083175086
        Accuracy Score: 0.4729965818458033
              precision    recall  f1-score   support

           0       0.16      0.79      0.26      1281
           1       0.08      0.46      0.13       150
           2       0.11      0.71      0.19       724
           3       0.02      0.25      0.03        44
           4       0.10      0.65      0.17       650
           5       0.04      0.46      0.07       109

   micro avg       0.11      0.70      0.20      2958
   macro avg       0.08      0.55      0.14      2958
weighted avg       0.12      0.70      0.21      2958
 samples avg       0.05      0.07      0.05      2958
Completed in [83.41879670000003 sec.]
================================================================================


================================================================================
Current Model in Progress: MultinomialNB
================================================================================
Training:  BinaryRelevance(classifier=MultinomialNB(), require_dense=[True, True])
Testing:

        Hamming Loss  : 0.024091657171793898
        Accuracy Score: 0.9074060007595898
              precision    recall  f1-score   support

           0       0.94      0.48      0.63      1281
           1       1.00      0.01      0.01       150
           2       0.93      0.45      0.60       724
           3       0.00      0.00      0.00        44
           4       0.84      0.35      0.49       650
           5       0.00      0.00      0.00       109

   micro avg       0.91      0.39      0.55      2958
   macro avg       0.62      0.21      0.29      2958
weighted avg       0.87      0.39      0.53      2958
 samples avg       0.04      0.03      0.04      2958
Completed in [20.29213630000004 sec.]
================================================================================
```

Observation:From the above model comparison, it is clear that MultinomialNB performs better with Accuracy Score: 90.74% and Hamming Loss: 2.4% than other models. Therefore, we will use MultinomialNB.

Data Visualization

Univariate analysis is the simplest form of data analysis where the data being analyzed contains only one variable. In this project, distribution plot, count plot, box plot, and bar plot have been used.



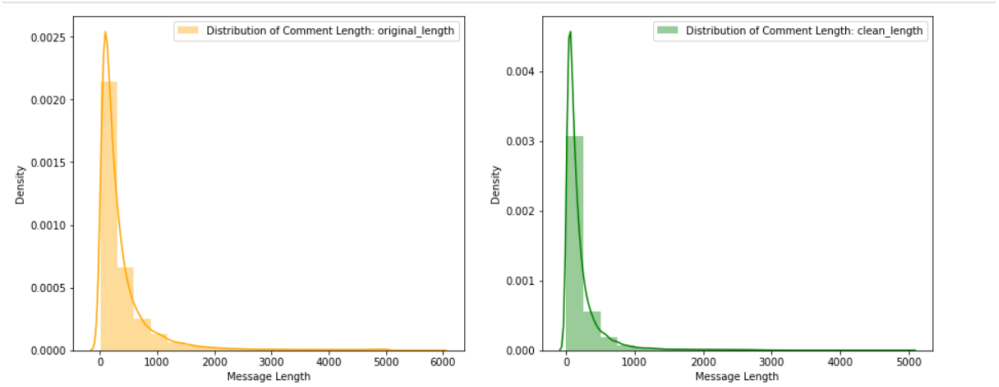
Observations:

• Dataset consists of a higher number of Normal Comments than Bad or Malignant Comments. Therefore, it is clear that the dataset is imbalanced and needs to be handled accordingly.

• Most of the bad comments are of type malignant while the least number of type threat is present in the dataset.

• Majority of bad comments are of type malignant, rude, and abusive.

Distribution Plot:



Observations: Before cleaning comment text, most of the comment's length lies between 0 to 1100 while after cleaning, it lies between 0 to 900.

Displaying with WordCloud:

Observations:

- From the word cloud of malignant comments, it is clear that it mostly consists of words like fuck, nigger, moron, hate, suck, etc.
- From the word cloud of highly_malignant comments, it is clear that it mostly consists of words like ass, fuck, bitch, shit, die, suck, faggot, etc.

- From the word cloud of rude comments, it is clear that it mostly consists of words like nigger, ass, fuck, suck, bullshit, bitch, etc.
- From the word cloud of threatening comments, it is clear that it mostly consists of words like die, must die, kill, murder, etc.
- From the word cloud of abuse comments, it is clear that it mostly consists of words like a moron, nigger, fat, jew, bitch, etc.
- From the word cloud of loathing comments, it is clear that it mostly consists of words like nigga, stupid, nigger, die, gay cunt, etc.

Interpretation of Results:
Starting with univariate analysis, with the help of a count plot it was found that the dataset is imbalanced with having a higher number of records for normal comments than bad comments (including malignant, highly malignant, rude, threat, abuse and loathe). Also, with the help of a distribution plot for comments length, it was found that after cleaning most of the length of the comment decreased from a range of 0-1100 to 0-900.

Moving further with word cloud it was found that malignant comments consist of words like fuck, nigger, moron, hate, suck, etc. highly_malignant comments consist of words like ass, fuck, bitch, shit, die, suck, faggot, etc. rude comments consists of words like nigger, ass, fuck, suck, bullshit, bitch, etc. threat comments consists of words like die, must die, kill, murder, etc. abuse comments consists of words like a moron, nigger, fat, jew, bitch, etc. and loathe comments consists of words like nigga, stupid, nigger, die, gay, cunt, etc.

Conclusion

1. Key Findings and Conclusions of the Study

The finding of the study is that only a few users over online use unparliamentary language. And most of these sentences have more stop words and are quite long. As discussed before few motivated disrespectful crowds use these foul languages in the online forum to bully the people around and to stop them from doing these things that they are not supposed to do. Our study helps the online forums and social media to induce a ban on profanity or usage of profanity over these forums.

2. Learning Outcomes of the Study in respect of Data Science

Through this project we were able to learn various Natural language processing techniques like lemmatization, stemming, and removal of stopwords. We were also able to learn to convert strings into vectors through a hash vectorizer. In this project, we applied different evaluation metrics like log loss and hamming loss besides accuracy.

My point of view from the project is that we need to use proper words which are respectful and also avoid using abusive, vulgar, and worst words on social media. It can cause many

problems which could affect our lives. Try to be polite, calm, and composed while handling stress and negativity and one of the best solutions is to avoid it and overcoming positively.

3. Limitations of this work and Scope for Future Work

- Problems faced while working on this project:
- More computational power was required as it took more than 2 hours
- Imbalanced dataset and bad comment texts Areas of improvement:
- Could be provided with a good dataset that does not take more time.
- Less time complexity
- Providing a properly balanced dataset with fewer errors.