



Malignant Comment Classifier



Submitted By -: IMRAN AHMAD

ACKNOWLEDGMENT

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

- I would like to thank FlipRobo Technologies for providing me this opportunity and guidance throughout the project and all the steps that are implemented.
- I have primarily referred to various articles scattered across various websites for the purpose of getting an idea on “*Malignant Comment Classifier*” project.
- I would like to thank the technical support team also for helping me out and reaching out to me on clearing all my doubts as early as possible.
- I would like to thank my project SME Mohd.Kashif for providing the flexibility in time and also for giving us guidance in creating the project.

I have referred to various articles in Towards Data Science and Kaggle sites.

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.

Conceptual Background of the Domain Problem: 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 clearly offensive.

Review of Literature: 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.

Motivation for the Problem Undertaken: These comments which here are referred as malignant here represents the human behaviour as well as tendency of the individual in the society. As a data scientist, the aim is to filter out these words from the social media from flashing of and blocking the user who does these types of act on the regular basis.

Analytical Problem Framing

Mathematical/Analytical Modelling of the Problem: Here we are dealing with one main text columns which held some importance of the data and others shows the multiple types of behaviour inferred from the text. I prefer to select on focus more on the words which has great value of importance in the context. Countvector is the NLP terms I am going to apply on text columns. This converts the important words a proper vectors with some weights.

Data Sources and their formats:

I received the dataset for this project through my mentor on this project and is in the .csv file. Some of the info other than malignant features are seems unnecessary with respect to the problem. So I drop it before any processing. Here is the snap of the data.

Data Pre-processing: Removing the stop-words, punctuations and special characters from the text from both feature is done here.

Data Inputs- Logic- Output Relationships The input data for the processing and getting the output is converted in to the numerical forms or it data words, in to the vector form. It feed to the model in the form of series (one by one) which analyses by the model providing the certain score through the medium of performance metrics.

Hardware and Software Requirements and Tools Used:

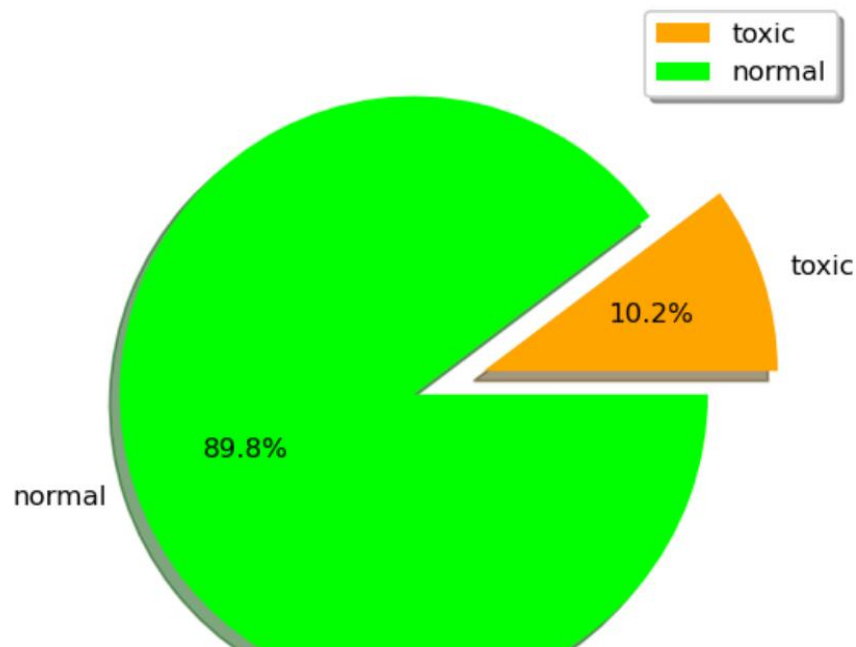
- Hardware And Software required for this project:
- Laptop with I3 processor and 8 GB Ram
- Jupyter Notebook
- Python Pandas for processing • Scikit learn library

```
1 import pandas as pd
2 import numpy as np
3 import seaborn as sns
4 import matplotlib.pyplot as plt
5 %matplotlib inline
6
7
8
9 import warnings
10 warnings.filterwarnings('ignore')
11
12 import re
13 from collections import Counter
14 import string
15 from string import digits as d, punctuation as p
```

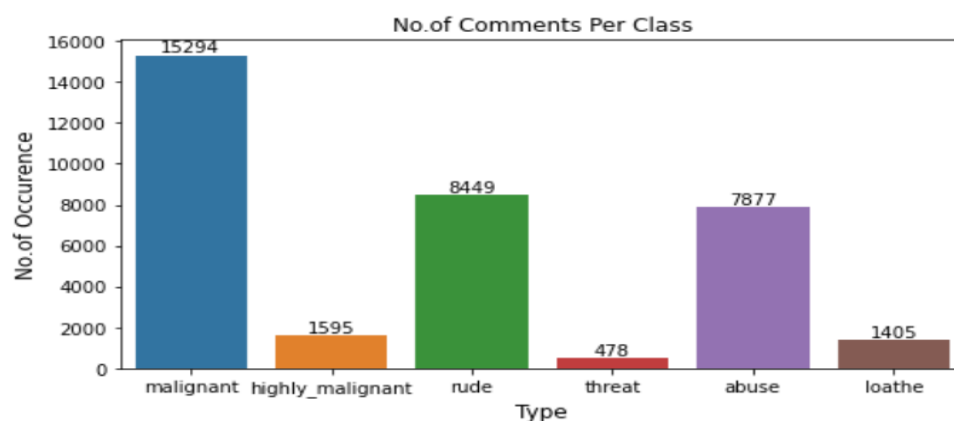
Models Development and Evaluation

Identification of possible problem-solving approaches (methods) These are the following approaches I have used here:

```
7 plt.legend(shadow = True, fancybox=True)  
8 plt.show()
```



- Importing and displaying insights of the data using pandas Dataframe.



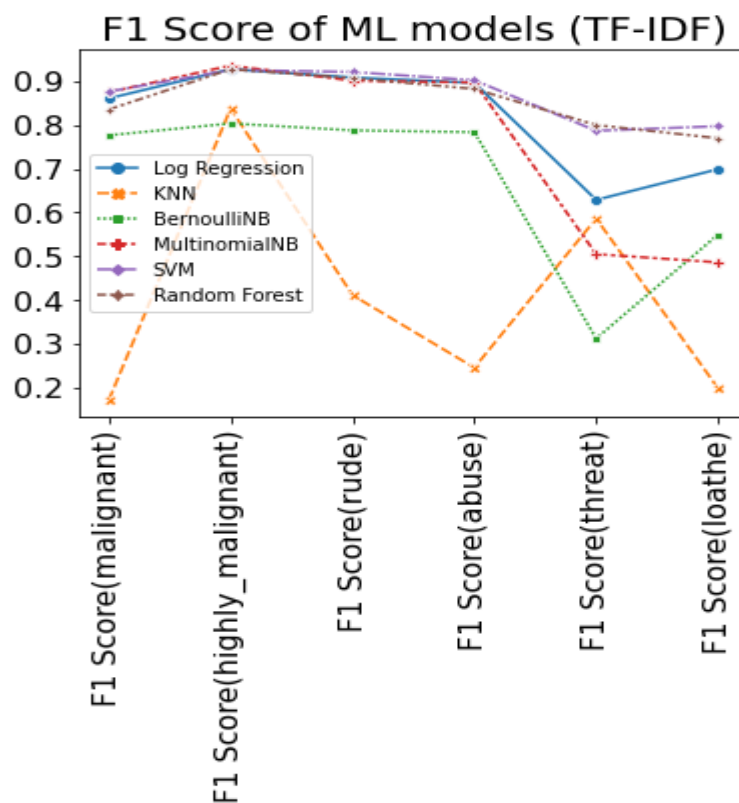
- Analysing the data and use proper pre-processing techniques to extract out important words.
- Applying data operations such as countvector to convert text to vectorized form.
- Applying algorithms to check the performance using evaluation metrics.

Testing of Identified Approaches (Algorithms)

Following are the algorithms that applied on this dataset. Random search is used for hyper parameter tuning the best model.

```
[70]: 1 sns.lineplot(data=final_all_trp, markers=True)
      2 plt.xticks(rotation='90', fontsize=16)
      3 plt.yticks(fontsize=16)
      4 plt.legend(loc='best')
      5 plt.title('F1 Score of ML models (TF-IDF)', fontsize=20)
```

```
[70]: Text(0.5, 1.0, 'F1 Score of ML models (TF-IDF)')
```



- LogisticRegression
- KNeighborsClassifier
- MultinomialNB, BernoulliNB
- LinearSVC
- RandomForestClassifier

Metrics for success in solving problem under consideration:

Key metrics are F1-score, Precision, Recall and Roc-Auc Score.

Interpretation of the Results

- LinearSVM and Random Forest models perform best (purple and brown lines seem to be the highest).

CONCLUSION

Key Findings and Conclusions of the Study Some of the findings from data explorations are as follows:

- From displaying the data, it seems there is lot of special characters present in the data. So, it is better to proceed by filter it out.
- As the above data is in text, so presence of special characters and stopwords is always there.
- After proper cleaning and processing, decision tree classifier gives the highest accuracy as well as roc score.

Limitations of this work and Scope for Future Work Some of the extention techniques that give the best results after optimization are:

- Multinomial naïve bayes and Random Forest using hyperparameter tuning.
- After analysing for each behaviour separately by creating models.
- Support Vector Machine also performs well in text data but its hyperparameter tuning is very complex and takes much more time.

Thank You