MALIGNANT COMMENTS CLASSIFICATION

Problem Statement

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.

Data Set Description

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.

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.

You need to build a model that can differentiate between comments and its categories.

Refer to the data set file provided along with this.

1)DATA CLEANING: we must check the data to be cleaned..as it contains some of the null values.We must remove the null values using the dropna()function.there are no duplicates found..we are good to go.

2)EXPLORATORY_ANALYSIS:We must check the distributions of our wanted columns or variables.we only require the ratings and reviews columns.we drop the rest of the columns.

3)DATA PREPROCESSING: since the reviews column is the string datatype,we must use the NLP preprocessing techniques.

First we convert the whole text into lower case letters,then we remove punctuations,tabs..etc

Then we tokenize the text,Remove the stopwords and Lemmatize the text.

Then we map the ratings to 0,1.

Next we use TFIDF technique to convert into machine readable texts

So our data is ready to train

4)MODEL BUILDING: we split the data into train and test.we train the data into different models such as multinomial NB,XGBclassifier,Randomforestclassifier,DecisionTreeClassifier and GradientBoostingClassifier.

5)MODEL EVOLUTION:we check the each model performance using the test data and we keep track of the model which performed well with good accuracy_score,classification_report.In our case RandomForestClassifier worked better when compared to other model interms of accuracy.

6)Selection of the best model after HyperParamter tuning..in our case randomforest classifier works better..