

**MALIGNANT COMMENTS CLASSIFICATION**

Submitted by:

MITESH VERMA

**ACKNOWLEDGMENT**

I would like to acknowledge Mr. Shwetank Mishra (FlipRobo) for his vital cooperation and help in ensuring the successful completion of my assignment. He deserves the utmost credit for the assignment’s outcome.

Finally, I would want to convey my sincere thanks Datatrained Academy and their guidance without them, the task would not have been accomplished.

The website that I referred are:

<https://learning.datatrained.com>

<https://www.w3schools.com>

https://medium.com/coders-camp

<https://github.com>

<https://www.geeksforgeeks.org>

https://www.kaggle.com

**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.

* Conceptual Background of the Domain Problem

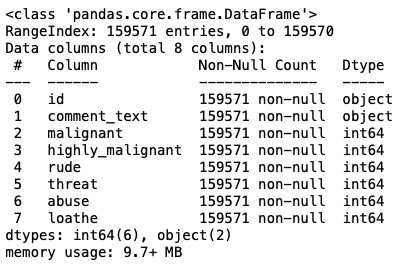
Our goal is to build a prototype of online hate and abuse comment classifier which can used to classify hate and offensive comments so that it can be controlled and restricted from spreading hatred and cyberbullying.

* Review of Literature
* **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.
* Motivation for the Problem Undertaken

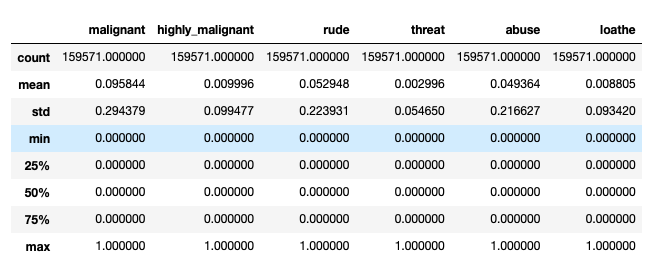
This project is on the NLP, data science and machine learning model, build the model to prototype of online hate and abuse comment classifier which can used to classify hate and offensive comments so that it can be controlled and restricted from spreading hatred and cyberbullying.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem
  + **Information of the dataset:**



* + **Description of the dataset:**



* Data Sources and their formats

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
  + 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.

* Data Pre-processing Done

**In data pre-processing, I have done the various steps to clean the dataset, as the dataset contains the comment that are in object datatype, which cannot be read by the model, so before giving the features to the model I had to convert that object datatype to meaningful data and that can be understand by the model, so for this I have used the NLP (Natural Processing Language).**

“Natural language processing (NLP) refers to the branch of computer science and more specifically, the branch of artificial intelligence (AI) concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.”

* Data Inputs- Logic- Output Relationships

Used TF-IDF Vectorizer to encode the comments section.

“TfidfVectorizer is the base building block of many NLP pipelines. It is a simple technique to vectorize text documents i.e. transform sentences into arrays of numbers and use them in subsequent tasks.”

* Hardware and Software Requirements and Tools Used

Anaconda-navigator

jupyter notebook

matplotlib-inline==0.1.6

numpy==1.23.2

packaging==21.3

pickleshare==0.7.5

platformdirs==2.5.2

prompt-toolkit==3.0.30

pyparsing==3.0.9

python-dateutil==2.8.2

scikit-learn==1.1.2

scipy==1.9.0

sklearn==0.05

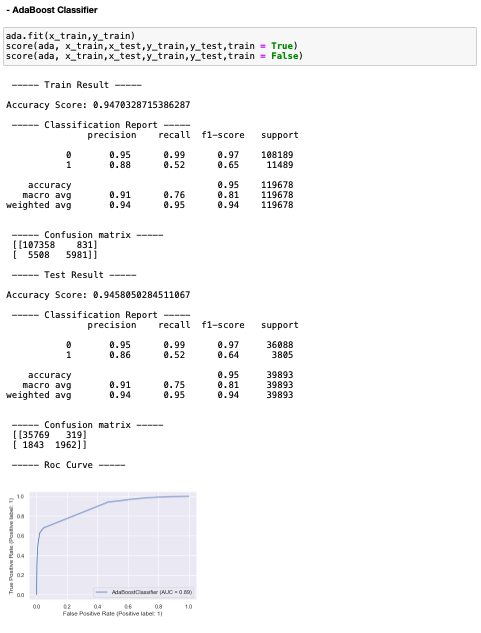
NLP==GPT3

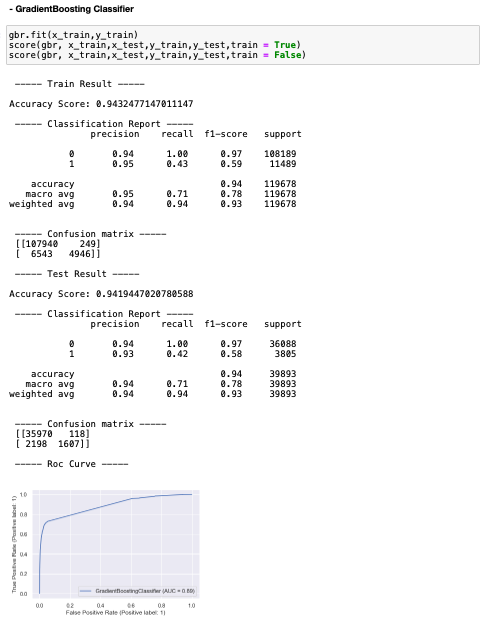
**Model/s Development and Evaluation**

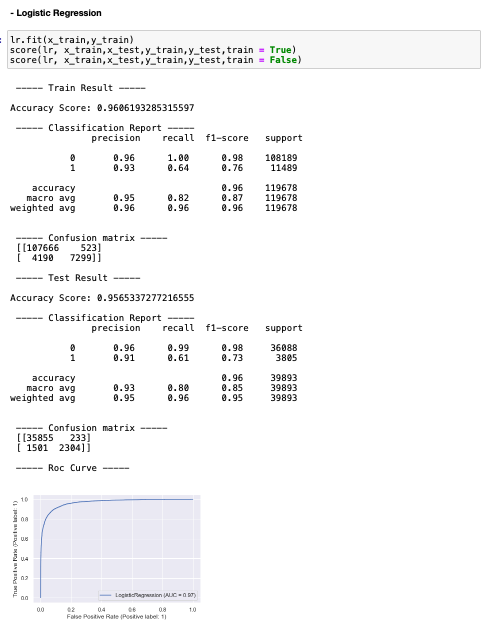
* Identification of possible problem-solving approaches (methods)
* **EDA**
* **Description**
* **Visualization**
* **Data cleaning**
* **Data Pre-processing (NLP)**
* **Word Cloud**
* **Encoding**
* **Model Building**
* **Select the best model**
* **Cross-Validation**
* **Hyperparameter tuning**
* Testing of Identified Approaches (Algorithms)

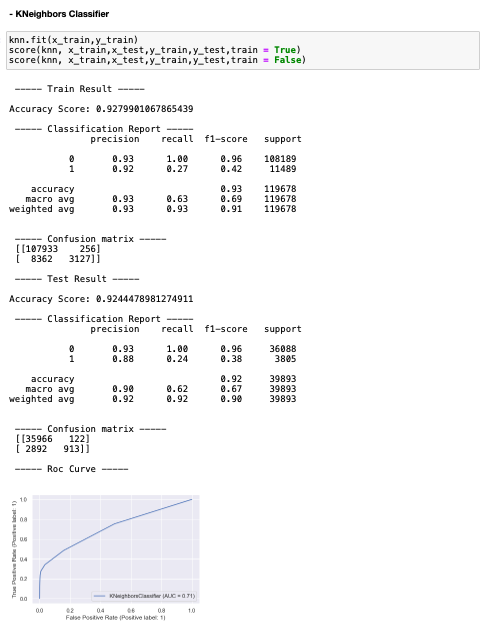
Algorithms used for the training and testing:

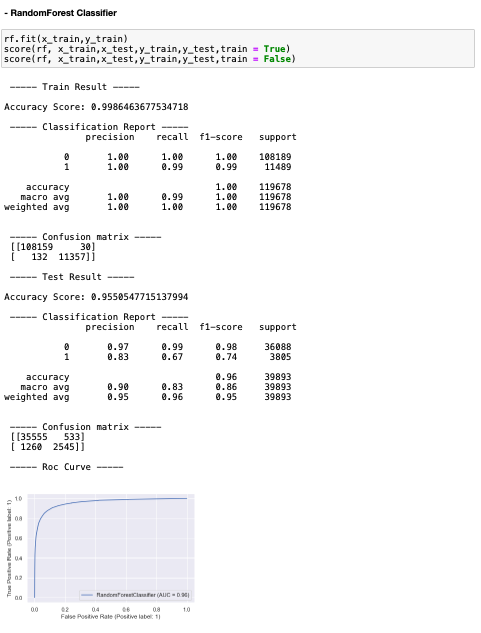
* + - AdaBoost Classifier
    - GradientBoosting Classifier
    - KNeighbors Classifier
    - RandomForest Classifier
    - Logistic Regression
* Run and Evaluate selected models











* Interpretation of the Results

RandomForest Classifier is giving the best score.