

MALIGNANT COMMENT CLASSIFIER

Submitted by:

NATASHA PODDAR

.

**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 cyberbullying.

* Conceptual Background of the Domain Problem

Basic knowledge on comments and types are needed

* Motivation for the Problem Undertaken

This issue is very realistic and common in today’s world and one should know to deal with such situations in the future

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

Count & Unique values were checked

Correlation with all independent variables and wrt target were checked

Skewness was checked and tools were applied to control them and scale the data

Models were applied to train and test the model

* Data Sources and their formats

Detailed data was procured from Malignant Comment Classifier Case Study ,

* Data Preprocessing Done

What were the steps followed for the cleaning of the data? What were the assumptions done and what were the next actions steps over that?

1. Duplicate values check
2. Unique & Count of all columns were checked
3. Skewness removal through Power Transform and scaling of the data
4. Correlation check
5. Graphical Univariate & Bivariate Analysis

* Data Inputs- Logic- Output Relationships

Mostly all the columns were low – moderately correlated with the target column, both positive and negative in nature

* Hardware and Software Requirements and Tools Used

1. Pandas – For Data Reading and understanding
2. Duplicate- To check for duplicate Values
3. CORR-To check Correlation
4. Numpy- For mathematical operations
5. LOGITSIC REGRESSION (SKLEARN) – Training & Testing the model
6. KNN CLASSIFIER (SKLEARN) – Training & Testing the model
7. DECISION TREE CLASSIFIER (SKLEARN) – Training & Testing the model
8. GAUSSIAN NB (SKLEARN) – Training & Testing the model
9. CROSS VAL SCORE – Regularizing the model
10. GRID SEARCH CV- Hyper Tuning the Model for higher accuracy
11. SEABORN- VISUALIZATION LIBRARY – HISTPLOTS, DISTPLOTS, SCATTERPLOTS, COUNTPLOTS, BOXPLOTS and other graphs
12. MATPLOTLIB.PY PLOT -Visualization tool

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

Correlation with all independent variables and wrt target were checked

Skewness was checked and tools were applied to control them and scale the data

Models were applied to train and test the model

* Testing of Identified Approaches (Algorithms)

1. LOGISTIC REGRESSION
2. KNN CLASSIFIER
3. DECISION TREE CLASSIFIER
4. MULTINOMIAL NB CLASSIFIER

* Key Metrics for success in solving problem under consideration

1. ACCURACY SCORE
2. CONFUSION MATRIX
3. CLASSIFICATION REPORT
4. F1 SCORE
5. PRECISION
6. RECALL SCORE
7. AUC-ROC SCORE

* Visualizations

Seaborn Library was used along with matplotlib Library for visualizations

Histplots, bar plots, count plots, swarmplots, boxplots etc were made and analysed

* Interpretation of the Results

All the models predicted an accuracy in the range of 89-91 where as DTC had the least difference between CV MEAN SCORE & MODEL ACCURACY SCORE hence we had hyper tuned the said model and saved the same

**CONCLUSION**

* Key Findings and Conclusions of the Study

DTC had the least difference between CV MEAN SCORE & MODEL ACCURACY SCORE hence we had hyper tuned the said model and saved the same

* Learning Outcomes of the Study in respect of Data Science

DTC had the least difference between CV MEAN SCORE % MODEL ACCURACY SCORE hence we had hyper tuned the said model and saved the same

With unique feature we realized the type of data all the columns had, The various visualization tools helped in understanding the different relationships between the variables .Cross Val score helped in regularizing the model