



Malignant Comments Classification Project

Submitted by:
Milind Nerpagar

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.

INTRODUCTION

- **Business Problem Framing**

- 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 it can be controlled and restricted from spreading hatred and cyber bullying.

- **Conceptual Background of the Domain Problem**

1. Firstly we will check whether the problem is supervised or not, mostly we will have a supervised one where there will be target variable.
2. After that we will check whether the project is a regression type or a classification type.
3. We will also check whether our dataset is balanced or imbalanced. If it is an imbalanced one, we will apply sampling techniques to balance the dataset.
4. Then we will do model building and check its accuracy.
5. Our main motto is to build a model with good accuracy and for that we will also go for hyperparameter tuning.

- **Review of Literature**

I am summarizing my research done on the topic.

- I have imported important libraries for my project.
- I have created the dataframe for both train and test dataset.
- I have analysed my data by checking the shape of the train and test dataset, information of the train and test dataset, presence of null values in train dataset if any and also the statistical summary of train and test dataset.
- Then I have done some data cleaning steps, e.g checking correlation between the dependant and independent variables using heatmap, visualizing data using count plots, removing stopwords, converting text into vectors using TF-IDF and splitting the data into independent and dependant variables.

▪ I have used Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Ada Boost Classifier and K-Neighbours Classifier for model building and Random Forest Classifier model is showing the best accuracy where Training accuracy is 0.9988898736783678, Test accuracy is 0.954879679144385 and cross validation score is 95.66211845827077.

- **Motivation for the Problem Undertaken**
- The objective behind to make this project 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 cyber bullying.

Analytical Problem Framing

- **Mathematical/ Analytical Modeling of the Problem**

If you look at data science, we are actually using mathematical models to model (and hopefully through the model to explain some of the things that we have seen) business circumstances, environment etc and through these model, we can get more insights such as the outcomes of our decision undertaken, what should we do next or how shall we do it to improve the odds. So mathematical models are important, selecting the right one to answer the business question can bring tremendous value to the organization.

Here I am using 5 algorithms for model building ,out of which Random Forest Classifier is giving us the best accuracy

- **.Data Sources and their formats**

Data Source: The read_csv function of the pandas library is used to read the content of a CSV file into the python environment as a pandas DataFrame. The function can read the files from the OS by using proper path to the file.

Data description: Pandas describe() is used to view some basic statistical details like percentile, mean, std etc. of a data frame or a series of numeric values. When this method is applied to a series of string, it returns a different output which is shown below.

• Data Preprocessing Done

- I have checked the correlation between the target variables.
- I have studied the statistical summary of the train and test dataset.
- I have done some visualization using count plots.
- Then I have converted the comments in train data into lowercase.
- Then I have removed punctuations , stopwords to get a clean length.
- Then I have converted text into vectors using TF-IDF so that the data gets ready for model building.
- I have splitted the independent and dependant variables into x and y.

• Data Inputs- Logic- Output Relationships

Hardware requirements:

Processor: Intel(R) Celeron(R) CPU N3050 @1.60 GHz 1.60 GHz

RAM: 3.92 GB

System type: 64-bit operating system, x64-based processor

Software requirements:

Python: One of the most used programming languages

Tools used:

Jupyter notebook: Jupyter is a free, open-source, interactive web tool known as a computational notebook where I have written my python codes.

NumPy: NumPy is an open-source numerical Python library.

NumPy contains a multi-dimensional array and matrix data structures.

Pandas: Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays.

Matplotlib: It is a Python library used for plotting graphs with the help of other libraries like Numpy and Pandas. It is a powerful tool for visualizing data in Python.

Seaborn: It is also a Python library used for plotting graphs with the help of Matplotlib, Pandas, and Numpy.

Scikit-learn: It is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction.

Scipy.stats: This module contains a large number of probability

distributions as well as a growing library of statistical functions. Natural Language Toolkit: NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

Describe the approaches you followed, both statistical and analytical, for solving of this problem.

- Testing of Identified Approaches (Algorithms)

Listing down all the algorithms used for the training and testing.

- Run and Evaluate selected models

Describe all the algorithms used along with the snapshot of their code and what were the results observed over different evaluation metrics.

- Key Metrics for success in solving problem under consideration

What were the key metrics used along with justification for using it? You may also include statistical metrics used if any.

- Visualizations

Mention all the plots made along with their pictures and what were the inferences and observations obtained from those. Describe them in detail.

If different platforms were used, mention that as well.

- **Interpretation of the Results**

Give a summary of what results were interpreted from the visualizations, preprocessing and modelling.

CONCLUSION

- **Key Findings and Conclusions of the Study**

The key findings are we have to study the data very clearly so that we are able to decide which data are relevant for our findings.

The conclusion of our study is we have to achieve a model with good accuracy and f1-score.

- **Learning Outcomes of the Study in respect of Data Science**

We will develop relevant programming abilities. We will demonstrate proficiency with statistical analysis of data. We will develop the ability to build and assess data-based models. We will execute statistical analyses with professional statistical software. The best algorithm for this project according to my work is Random Forest Classifier because the accuracy and f1 score that I have achieved is quite satisfactory than the other model.